Computer Science & Information Technology 159

Natural Language Processing

David C. Wyld,
Dhinaharan Nagamalai (Eds)

Computer Science & Information Technology

- 10th International Conference on Natural Language Processing (NLP 2021), December 23 ~ 24, 2021, Sydney, Australia
- 2nd International Conference on Machine Learning Techniques (MLTEC 2021)
- 2nd International Conference on Cloud and Big Data (CLBD 2021)
- 10th International Conference on Software Engineering and Applications (SEAPP 2021)
- 2nd International Conference on Networks & IOT (NeTIOT 2021)
- 2nd International Conference on VLSI & Embedded Systems (VLSIE 2021)
- 10th International Conference on Information Technology Convergence and Services (ITCS 2021)
- 8th International Conference on Artificial Intelligence & Applications (ARIA 2021)
- 12th International conference on Database Management Systems (DMS 2021)

Published By

AIRCC Publishing Corporation

Volume Editors

David C. Wyld,
Southeastern Louisiana University, USA
E-mail: David.Wyld@selu.edu

Dhinaharan Nagamalai (Eds),
Wireilla Net Solutions, Australia
E-mail: dhinthia@yahoo.com

ISSN: 2231 - 5403
ISBN: 978-1-925953-57-2
DOI: 10.5121/csit.2021.112301- 10.5121/csit.2021.112331

Typesetting: Camera-ready by author, data conversion by NnN Net Solutions Private Ltd., Chennai, India

Preface

10th International Conference on Natural Language Processing (NLP 2021), December 23 ~ 24, 2021, Sydney, Australia, 2nd International Conference on Machine Learning Techniques (MLTEC 2021), 2nd International Conference on Cloud and Big Data (CLBD 2021), 10th International Conference on Software Engineering and Applications (SEAPP 2021), 2nd International Conference on Networks & IOT (NeTIOT 2021), 2nd International Conference on VLSI & Embedded Systems (VLSIE 2021), 10th International Conference on Information Technology Convergence and Services (ITCS 2021), 8th International Conference on Artificial Intelligence & Applications (ARIA 2021) and 12th International conference on Database Management Systems (DMS 2021) was collocated with 10th International Conference on Natural Language Processing (NLP 2021). The conferences attracted many local and international delegates, presenting a balanced mixture of intellect from the East and from the West.

The goal of this conference series is to bring together researchers and practitioners from academia and industry to focus on understanding computer science and information technology and to establish new collaborations in these areas. Authors are invited to contribute to the conference by submitting articles that illustrate research results, projects, survey work and industrial experiences describing significant advances in all areas of computer science and information technology.

The NLP 2021, MLTEC 2021, CLBD 2021, SEAPP 2021, NeTIoT 2021, VLSIE 2021, ITCS 2021, ARIA 2021 and DMS 2021 Committees rigorously invited submissions for many months from researchers, scientists, engineers, students and practitioners related to the relevant themes and tracks of the workshop. This effort guaranteed submissions from an unparalleled number of internationally recognized top-level researchers. All the submissions underwent a strenuous peer review process which comprised expert reviewers. These reviewers were selected from a talented pool of Technical Committee members and external reviewers on the basis of their expertise. The papers were then reviewed based on their contributions, technical content, originality and clarity. The entire process, which includes the submission, review and acceptance processes, was done electronically.

In closing, NLP 2021, MLTEC 2021, CLBD 2021, SEAPP 2021, NeTIoT 2021, VLSIE 2021, ITCS 2021, ARIA 2021 and DMS 2021 brought together researchers, scientists, engineers, students and practitioners to exchange and share their experiences, new ideas and research results in all aspects of the main workshop themes and tracks, and to discuss the practical challenges encountered and the solutions adopted. The book is organized as a collection of papers from the NLP 2021, MLTEC 2021, CLBD 2021, SEAPP 2021, NeTIoT 2021, VLSIE 2021, ITCS 2021, ARIA 2021 and DMS 2021.

We would like to thank the General and Program Chairs, organization staff, the members of the Technical Program Committees and external reviewers for their excellent and tireless work. We sincerely wish that all attendees benefited scientifically from the conference and wish them every success in their research. It is the humble wish of the conference organizers that the professional dialogue among the researchers, scientists, engineers, students and educators continues beyond the event and that the friendships and collaborations forged will linger and prosper for many years to come.

David C. Wyld,
Dhinaharan Nagamalai (Eds)

General Chair	Organization
David C. Wyld,	Southeastern Louisiana University, USA
Dhinaharan Nagamalai (Eds)	Wireilla Net Solutions, Australia

Program Committee Members

Abd El-Aziz Ahmed,	Cairo University, Egypt
Abdel-Badeeh M. Salem,	Ain Shams University, Egypt
Abdelhadi Assir,	Hassan 1st University, Morocco
Abdelhak Merizig,	Mohamed Khider University, Algeria
Abderrahmane Ez-zahout,	Mohammed V University, Morocco
Abdessamad Belangour,	University of Hassan II Casablanca, Morocco
Abdul Khalique Shaikh,	Sultan Qaboos University, Sultanate of Oman
Abdullah,	Adigrat University, Ethiopia-Africa
Abhishek Das,	Aliah University, India
Addisson Salazar,	Universitat Politècnica de València, Spain
Ahmad A. Saifan,	Yarmouk University, Jordan
Ahmed Farouk AbdelGawad,	Zagazig University, Egypt
Ahmed Kadhim Hussein,	Babylon university babylon city, Iraq
Ahmet Çifci,	Burdur Mehmet Akif Ersoy University, Turkey
Aishwarya Asesh,	Adobe, USA
Ajay Anil Gurjar,	Sipna College of Engineering & Technology, India
Ajay B Gadicha,	Amravati University, India
Ajit Kumar Singh,	Patna Women's College, India
Akhil Gupta,	Lovely Professional University, India
Alexander Gelbukh,	Instituto Politécnico Nacional, Mexico
Ali A. Amer,	Taiz University, Yemen
Alireza Valipour Baboli,	Technical and Vocational University, Iran
Allel Hadjali,	Lias/Ensma, France
Amel Ourici,	Badji Mokhtar University of Annaba, Algeria
Amir abbas baradaran,	Shahid beheshti University of Iran, Tehran
Amiya Kumar Tripathy,	Don Bosco Institute of Technology, India
Amizah Malip,	University of Malaya, Malaysia
Ammar Khader Almasri,	Al-Balqa Applied University, Jordan
Ana Luísa Varani Leal,	University of Macau, Macau
Anand Nayyar,	Duy Tan University, Vietnam
Andras Markus,	University of Szeged, Hungary
Andy Rachman,	Institut Teknologi Adhi Tama Surabaya, Indonesia
Ankush Ghosh,	The Neotia University, India
Anouar Abtoy,	Abdelmalek Essaadi University, Morocco
António Abreu,	ISEL - Polytechnic Institute of Lisbon, Portugal
Anwar Basha,	SRM Institute of Science and Technology, India
Aoud Sahar,	Ibn Zohr University, Morocco
Aridj Mohamed,	Hassiba Benbouali University Chlef, Algeria
Arjav Bavarva,	Marwadi University, India
Arthur,	Universidade Federal de Santa Catarina, Brazil
Assia Djenouhat,	University Badji Mokhtar Annaba, Algeria
Attila Kertesz,	University of Szeged, Hungary
Atul Garg,	Chitkara University, India
B D C N Prasad,	V R Siddhartha Engineering college, India

B. K. Tripathy,	VIT, India
Balagadde,	Kampala International University, Uganda
Basant Verma,	G H Raisoni College of Engineering, India
Beshair Alsiddiq,	Prince Sultan University, Saudi Arabia
Bilal Muhammad Atif,	Jilin University, China
Bin Hu,	Changsha Normal University, China
Bin Zhao,	JD.com Silicon Valley R&D Center, USA
Boukari Nassim,	Skikda University, Algeria
Brahami Menaouer,	National Polytechnic School of Oran, Algeria
BrahimLejdel,	University of El-Oued, Algeria
Chang-Yong Lee,	Kongju National University, South Korea
Cheng Siong Chin,	Newcastle University, Singapore
Chetan J. Shelke,	Alliance University, India
Chin-Ling Chen,	Chaoyang University of Technology, Taiwan
Christian Mancas,	DATASIS ProSoft srl, Bucharest, Romania
Dadmehr Rahbari,	Tallinn University of Technology, Iran
Daniel Hunyadi,	Lucian Blaga University of Sibiu, Romania
Dário Ferreira,	University of Beira Interior, Portugal
Debjani Chakraborty,	Indian Institute of Technology, India
Denis Reilly,	Liverpool John Moores University, UK
Dhamyaa Saad Khudhur,	Mustansiriyah University, Iraq
Dibya Mukhopadhyay,	University of Alabama, USA
Dinesh Reddy Vemula,	SRM University, India
Dinesh Reddy.V,	SRM Institute of Science and Technology, India
Domenico Calcaterra,	University of Catania, Italy
Domenico Rotondi,	Fincons SpA, Italy
Ekbal Rashid,	RTC Institute of Technology, India
El Murabet Amina,	Abdelmalek Essaadi University, Morocco
Elżbieta Macioszek,	Silesian University of Technology, Poland
Endre Pap,	Singidunum University, Serbia
F. Abbasi,	Islamic Azad University, Iran
Faouzia Benabbou,	University Hassan II of Casablanca, Morocco
Fatiha Merazka,	University of Science and Technology, Algeria
Fazlollah Abbasi,	Islamic Azad University, Iran
Felix Jesus Garcia Clemente,	University of Murcia, Spain
Filiz Cele,	Istanbul Aydin UNniversity, Turkey
Francesco Zirilli,	Sapienza Universita Roma, Italy
Friday Zinzendoff Okwonu,	Universiti Utara Malaysia, Malaysia
Fu Jen,	Catholic University, Taiwan
Fulvia Pennoni,	University of Milano-Bicocca, Italy
G. Rajkumar,	N.M.S.S.Vellaichamy Nadar College, India
Gajendra Sharma,	Kathmandu University, Nepal
Gang Liu,	Harbin Engineering University, China
Gang Wang,	University of Connecticut, USA
Geeta R. Bharamagoudar,	KLE Institute of technology, India
Gniewko Niedbała,	Poznań University of Life Sciences, Poland
Grigorios N. Beligiannis,	University of Patras, Greece
Grzegorz Sierpiński,	Silesian University of Technology, Poland
Gururaj H L,	Vidyavardhaka College of Engineering, India
HamedTaherdoost,	University West Canada, Canada
Hamid Ali Abed Al-Asadi,	Iraq University College, Iraq
Hamidreza Rokhsati,	Sapienza University of Rome, Italy

Hao-En Chueh,	Chung Yuan Christian University, Taiwan
Haruna Abdu,	Federal University Lokoja, Nigeria
Heba AL.Marwai,	Sana'a University, Yeman
Hedayat Omidvar,	Research & Technology Dept, Iran
Hlaing Htake Khaung Tin,	University of Computer Studies, Myanmar
Hunyadi Ioan Daniel,	Lucian Blaga University of Sibiu, Romania
Hyunsung Kim,	Kyungil University, Korea
Ibrahim Abu El-Khair,	Minia University, Egypt
Ilango Velchamy,	CMR Institute of Technology, India
Ilham Huseyinov,	Istanbul Aydin University/Istanbul, Turkey
Israa Shaker Tawfic,	Ministry of Science and Technology, Iraq
Iyad Alazzam,	Yarmouk University, Jordan
J K Mantri,	MSCB University, India
J.Naren,	Rathinam College of Arts and Science, India
Jahnavi Patel,	Paderborn University, Germany
Janusz Kacprzyk,	Polish Academy of Sciences, Poland
Janusz Wielki,	Opole University of Technology, Poland
Jasmin Cosic,	DB Netz AG, Research & Development, Germany
Jawad K. Ali,	University of Technology, Iraq
Jiajun Sun,	Huaiyin Normal University, China
Joan Lu,	University of Huddersfield, UK
Jose Alfredo F. Costa,	Federal University, Brazil
Jyoti, Jaypee	University, India
Kamel Hussein Rahouma,	Nahda University, Egypt
Kamel Jemai,	University of Gabes, Tunisia
Ke-Lin Du,	Concordia University, Canada
Khosrow Shafiei Motlagh,	Islamic Azad University, Iran
Ki-Il Kim,	Chungnam Natiuonal University, Korea
Kirtikumar Patel,	I&E Engineer, USA
Labraoui Nabila,	University of Tlemcen, Algeria
Larbi Guezouli,	University of Batna 2, Algeria
Loc Nguyen,	Independent Scholar, Vietnam
Luisa Maria Arvide Cambra,	University of Almeria, Spain
Lutz Schubert,	University of Ulm, Germany
M V Ramana Murthy,	Osmania University, India
M. Akhil Jabbar,	Vardhaman College of Engineering, India
Mabroukah Amarif,	Sebha University, Libya
Madhanmohan,	Annamalai University, India
Mahdi Sabri,	Islamic Azad University Urmia Branch, Iran
Mai Zaki,	American University of Sharjah, UAE
Mamoun Alazab,	Charles Darwin University, Australia
Manal Mostafa,	Al Azhar University, Egypt
Manish Kumar Mishra,	University of Gondar, Ethiopia
Manuel J. Cobo,	University of Cádiz, Spain
Marcin Paprzycki,	Adam Mickiewicz University, Poland
Marco Javier Suarez Baron,	University in Tunja, Colombia
Marichelvam,	Mepco Schlenk Engineering College, India
Mario Versaci,	DICEAM - Univ. Mediterranea, Italy
Marius Cioca,	University of Sibiu, Romania
Maryam Rastgarpour,	Azad University, Iran
Masoomeh Mirrashid,	Semnan University, Iran
M-Chun Su,	National Central University, Taiwan

Md. Maniruzzaman,	Khulna University, Bangladesh
Mehdi Nezhadnaderi,	Islamic Azad University, Iran
Mehdi Soltani,	Qazvin Islamic Azad University, Iran
MERIAH Sidi Mohammed,	University of Tlemcen,Algeria
Metais Elisabeth,	Le Cnam, France
Michail Kalogiannakis,	University of Crete, Greece
Mitali Chugh,	UPES, India
Moceheb Lazam Shuwandy,	Tikrit University, Iraq
Mohamed Abdelaziz Hassan Eleiwa,	University of Hail, Egypt
Mohamed Anis Bach Tobji,	University of Manouba, Tunisia
Mohamed Arezki Mellal,	M'Hamed Bougara University, Algeria
Mohamed Hamlich,	ENSAM, UH2C, Morocco
Mohamed Sbai,	Université de Tunis El Manar, Tunisia
Mohammad A. Alodat,	Sur University College, Oman
Mohammad Hajjar,	Lebanese University, Lebanon
Mohammad Shameem,	KL University, India
Morris Riedel,	University of Iceland, Iceland
Mounir Zrigui,	University of Monastir, Tunisia
Mousse Ange Mikael,	Université de Parakou, Benin
Mueen Uddin,	Universiti Brunei Darussalam, Malaysia
Muhammad Sarfraz,	Kuwait University, Kuwait
Mu-Song Chen,	Da-Yeh University, Taiwan
Mu-Yen Chen,	National Cheng Kung University, Taiwan
Nadia Abd-Alsabour,	Cairo University, Egypt
Nahlah Shatnawi,	Yarmouk University, Jordan
Namrata Dhanda,	Amity University, India
Narinder Singh,	Punjabi University, Punjab, India
Neha Pattan,	Carnegie Mellon University, Pennsylvania
Ngoc Hong Tran,	Vietnamese-German University, Vietnam
Nihar Athreyas,	Spero Devices Inc, USA
Nikola Ivković,	University of Zagreb, Croatia
Noor Mowafeq Al layla,	Mosul University, Iraq
Noraziah Ahmad,	University Malaysia Pahang, Malaysia
Nour El Houda Golea,	Batna 2 University, Algeria
Noureddin Amaigarou,	Abdelmalek Essaid University, Morocco
Odedoyin Abiodun Omolara,	INTECU, Nigeria
Oleksii K. Tyshchenko,	University of Ostrava, Czech Republic
P. S. Hiremath,	KLE Technological University, India
P. Susheelkumar S,	University of Mumbai, India
P.S. Hiremath,	KLE Technological University, India
P.V.Siva Kumar,	VNR VJIET, India
Paulo Trigo,	ISEL/GuIAA, Portugal
Pavel Loskot,	ZJU-UIUC Institute, China
Petra Perner,	FutureLab Artificial Intelligences IBaI-2, Germany
Prasan Kumar Sahoo,	Chang Gung University, Taiwan
Preeti Garg,	Shobhit University, India
Qi Zhang,	Shandong University, China
Raed Ibraheem Hamed,	University of Anbar, Iraq
Rahul M Mulajkar,	Jaihind College of Engineering, India
Rajeev Kanth,	University of Turku, Finland
Rajkumar,	N.M.S.S.Vellaichamy Nadar College, India
Rakesh Kumar Mahendran,	Anna University, India

Ramadan Elaiess,	University Of Benghazi, Libya
Ramgopal Kashyap,	Amity University Chhattisgarh, India
Rami Raba,	Al_Azhar University- Gaza, Palestine
Ricardo Branco,	University Of Coimbra, Portugal
Richa Purohit,	DY Patil International University, India
Ruhaidah Samsudin,	Universiti Teknologi, Malaysia
S.Sibi Chakkaravarthy,	Vellore Institute of Technology, India
S.Taruna,	JK Lakshmipat University, India
Saad Al- Janabi,	Al- hikma college university, Iraq
Sabina Rossi,	Università Ca' Foscari Venezia, Italy
Sadique Shaikh,	AIMSR, India
Sami Bedra,	University of Khenchela, Algeria
Samira Hazmoune,	University of Skikda, Algeria
Sandeep Chaurasia,	Manipal University, India
Satish Gajawada,	IIT Roorkee Alumnus. India
Sebastian Floerecke,	University of Passau, Germany
Sebastian Fritsch,	IT and CS enthusiast, Germany
Seppo Sirkemaa,	University of Turku, Finland
Seyed Mahmood Hashemi,	KAR University, Iran
Shahid Ali,	AGI Education Ltd, New Zealand
Shahram Babaie,	Islamic Azad University, Iran
Shaima Wikars,	University of Vaasa, Finland
Sharathyh Kumar,	Mit Mysore, India
Shashikant Patil,	NMIMS Deemed-to-be-University, India
Sherri Harms,	University of Nebraska, USA
Shervan Fekri-Ershad,	Islamic Azad University, Iran
Shi Dong,	Zhoukou Normal University, China
Shing-Tai Pan,	National University of Kaohsiung, Taiwan
Siarry Patrick,	Universite Paris-Est Creteil, France
Siddhartha Bhattacharyya,	Rajnagar Mahavidyalaya, India
Sidi Mohammed Meriah,	University of Tlemcen, Algeria
Sikandar Ali,	China University of Petroleum, China
Smain Femmam,	UHA University, France
Sobiawasan,	Nanjing University, China
SofianeSofiane,	University Abbes Laghrour Khenchela, Algeria
Somayeh Mohamadi,	Islamic Azad University, Iran
Sonia Martin Gomez,	Universidad San Pablo - CEU, Spain
Souad Taleb,	Oran 2 University, Algeria
Sridharan D,	Anna University, India
Stefano Michieletto,	University of Padova, Italy
Subhendu kumar pani,	BPUT, India
Suhad Faisal Behadili,	University of Baghdad, Iraq
Sunil Karamchandani,	University of Mumbai, India
Swati Nikam,	Savitribai Phule Pune University, India
Tadonki,	MINES ParisTech - PSL, France
Taleb zouggar souad,	Oran 2 University, Algeria
Tanzila Saba,	Prince Sultan University, Saudi Arabia
Taskeen zaidi,	Shri Ramswaroop Memorial University, India
Ubhendu Kumar Pani,	Krupajal Computer Academy, India
Umesh kumar singh,	Vikram University, India
Vahideh Hayyolalam,	Koc University, Turkey
Vanlin Sathya,	University of Chicago, USA

Venkata Duvvuri,	Purdue University, USA
Vilem Novak,	University of Ostrava, Czech Republic
Vinita Verma,	University of Delhi, India
Virgínia Araújo,	Atlântica University Institute, Portugal
Virupakshappa,	Sharnbasva University Kalaburagi, India
Vuda Sreenivasarao,	Bahir Dar University, Ethipoia
Waleed Bin Owais,	Qatar University, Qatar
Wei Cai,	Qualcomm Tech, USA
Wei-Chiang Hong,	Jiangsu Normal University, China
Yang Cao,	Southeast University, China
Yanrong Lu,	Civil Aviation University of China, China
Yew Kee Wong,	HuangHuai University, China
Yousef Farhaoui,	Moulay Ismail University, Morocco
Yu-Chen Hu,	Providence Universiy, Taiwan
Yugen Yi,	Jiangxi Normal University, China
Zakaria Kurdi,	University of Lynchburg, Virginia, USA
Zhihao Wu,	Shanghai Jiao Tong University, China
Zhihong Tian,	Guangzhou University, China
Zhou Quan,	Guangzhou University, China
Zoran Bojkovic,	University of Belgrade, Serbia

Technically Sponsored by

Computer Science & Information Technology Community (CSITC)

Artificial Intelligence Community (AIC)

Soft Computing Community (SCC)

Digital Signal & Image Processing Community (DSIPC)

10th International Conference on Natural Language Processing (NLP 2021)

2nd International Conference on Machine Learning Techniques (MLTEC 2021)

2nd International Conference on Cloud and Big Data (CLBD 2021)

10th International Conference on Software Engineering and Applications (SEAPP 2021)

2nd International Conference on Networks & IOT (NeTIOT 2021)

2nd International Conference on VLSI & Embedded Systems (VLSIE 2021)

10th International Conference on Information Technology Convergence and Services (ITCS 2021)

8th International Conference on Artificial Intelligence & Applications (ARIA 2021)

12th International conference on Database Management Systems (DMS 2021)

An Enhanced Machine Learning Topic Classification Methodology for Cybersecurity

Elijah Pelofske[1], Lorie M. Liebrock[1], and Vincent Urias[2]

[1]Cybersecurity Centers, New Mexico Institute of Mining and Technology, Socorro, New Mexico, USA
[2]Sandia National Laboratories, Albuquerque, New Mexico, USA

Abstract. In this research, we use user defined labels from three internet text sources (Reddit, Stackexchange, Arxiv) to train 21 different machine learning models for the topic classification task of detecting cybersecurity discussions in natural text. We analyze the false positive and false negative rates of each of the 21 model's in a cross validation experiment. Then we present a Cybersecurity Topic Classification (CTC) tool, which takes the majority vote of the 21 trained machine learning models as the decision mechanism for detecting cybersecurity related text. We also show that the majority vote mechanism of the CTC tool provides lower false negative and false positive rates on average than any of the 21 individual models. We show that the CTC tool is scalable to the hundreds of thousands of documents with a wall clock time on the order of hours.

Keywords: cybersecurity, topic modeling, text classification, machine learning, neural networks, natural language processing, Stackexchange, Reddit, Arxiv, social media

1 Introduction

Identifying cybersecurity discussions in open forums at scale is a topic of great interest for the purpose of mitigating and understanding modern cyber threats [11, 18, 20]. The challenge is that often these discussions are quite noisy (i.e., they contain community known synonyms or acronyms) and difficult to get labelled data in order to train resilient NLP (natural language processing) topic classifiers. Additionally, it is important that a tool which detects cybersecurity discussions in internet text sources is *scalable* and offers *low errors rates* (in particular, both low false negative rates and low false positive rates).

In order to address the challenges of finding relevant cybersecurity labelled data, we use a technique of gathering posts or articles from different internet sources which have user defined *topic labels*. We then collect and label the training text as being cybersecurity related or not based on the subset of labels that text source offers. Thus, the labelled training data we gather is not manually labelled by researchers; instead it is labelled inherently by the system we gather the text from. This provides the additional benefit that for cybersecurity related discussions, it might be difficult for a manual labelling process to catch all of the noise (e.g., unknown synonyms), but this labelling method uses the user defined labels (which removes the need for the labelling process to identify all known cybersecurity

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 01-22, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112301

terms used in online discussion forums). Lastly, this method of gathering labelled data is *highly* scalable. The reason is that the platforms we used have publicly available data and systems to retrieve that data in very large amounts. We used three specific sources of text: Reddit, Stackexchange, and Arxiv.

Using the topic classification labelled data, we train a total of 21 different machine learning models using several different algorithms and the three different text sources. We then show the validation accuracy of the models, as well as the cross validation (i.e., validating a model on a text source upon which it was not trained) accuracy of each of the models. We next present the Cybersecurity Topic Classification (CTC) tool, which uses the majority vote consensus of all 21 trained models in order to evaluate whether a novel document is cybersecurity related or not. We show that the CTC tool has both scalability to hundreds of thousands of documents per hour and low error rates. Lastly, we provide all of the labelled data we used to train and validate the models in an open source Github repository [22].

This article is structured as follows. After a brief literature review in Section 1.1, we define our methods of gathering labelled data and then training and validating the machine learning models in Section 2. Section 3 describes the experiments. The investigation of how the minimum token length of the training data changes false negative and false positive rates is presented in Section 3.1. After training each of the models using the specified parameters, in Section 3.2 we show the validation accuracy rates for each of the 21 trained models. In Section 3.3 we present the CTC tool and show it's scalability and high accuracy. Section 4 discusses conclusions and future work.

1.1 Previous work

There is significant interest surrounding the goal of being able to automate cybersecurity threat detection on social media [18, 17, 14, 10, 11, 25, 13, 2]. Twitter, Reddit, and Stackexchange are popular forums from which several previous studies have gathered cybersecurity related documents [18, 10, 13, 14, 2, 19] for the purpose of training machine learning detection systems and classifiers. In particular, [18] used tags (or other community defined mechanisms) as a document labelling method for cybersecurity topic classification related text. There is also some interest in investigating vulnerability discussions on developer sites such as Stackoverflow [17, 19].

There are several different approaches taken with which topic modelling task to use as a signal to detect cybersecurity discussions. Typically the topic classification task is related to training directly on labelled text and then perhaps developing an idea of the more relevant keywords in these discussions [18, 11]. Other researchers use sentiment analysis in conjunction with other machine learning models [25, 10].

There are also interesting approaches that use social media as a signal to detect specific cyber-attacks (e.g., DDoS attacks) or vulnerabilities [14, 17, 18, 2].

Table 1: Number of documents from each text source and the labelling method used for each source

Source	cybersecurity	non cybersecurity	labelling method
Reddit	164750	4184184	sub-reddits
Stackexchange	41162	4842461	post topic labels
Arxiv	12132	28996	keyword search + topic restriction

For a review of text classification with deep learning models, see [20], and for a survey of gathering social media data, see [6].

2 Methods

This section describes the methods used to gather labelled text, preprocess and vectorize that text, train several machine learning models using the labelled data, and lastly evaluate the accuracy of these machine learning models.

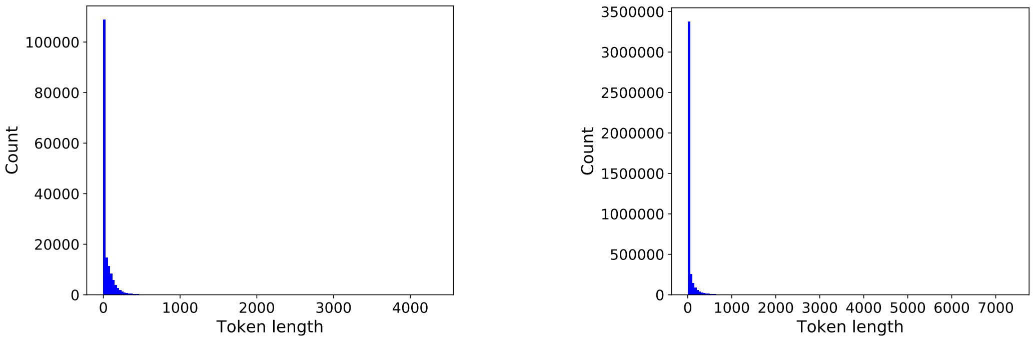
All figures in this article were created using Matplotlib and Python3.7 [12].

2.1 Text sources

For this research, we focus on gathering large amounts of text from the three sources Reddit, Stackexchange, and Arxiv. For gathering Reddit text we used the python modules *praw* and *psaw* [23, 1]. In order to query data from StackExchange, we used the python module *StackAPI* (which is a python wrapper for the StackExchange API [26]). For gathering Arxiv documents, we used the python module *arxiv* [5]. As a summary of the raw data collected, Table 1 shows the number of cybersecurity and non cybersecurity documents gathered from each source, along with the document labelling method for each source. Next we define the precise methodology for gathering and labelling the documents from each source.

Reddit Reddit [24] is an internet discussion website that allows registered users to submit different types of content (e.g., text posts, images, links, videos) to the site. Each of these posts then get voted on (upvoted or downvoted) by other users. The entire site is logically organized into sub-reddits. Each sub-reddit has a specific scope and topic of discussion. The document labeling strategy we use is to label documents according to which sub-reddit they originate from.

For gathering cybersecurity related text, we first defined 40 cybersecurity related sub-reddits. Next, we queried each of those sub-reddits with a maximum number of posts



(a) Cybersecurity documents.

(b) Non cybersecurity documents.

Fig. 1: Reddit token data histogram

returned of $1,000,000$ (This does not mean we get $1,000,000$ posts for each sub-reddit). The default sorting method for the posts returned is a Reddit defined post metric called *Hot*; for the purpose of querying with the API, this can not be changed.

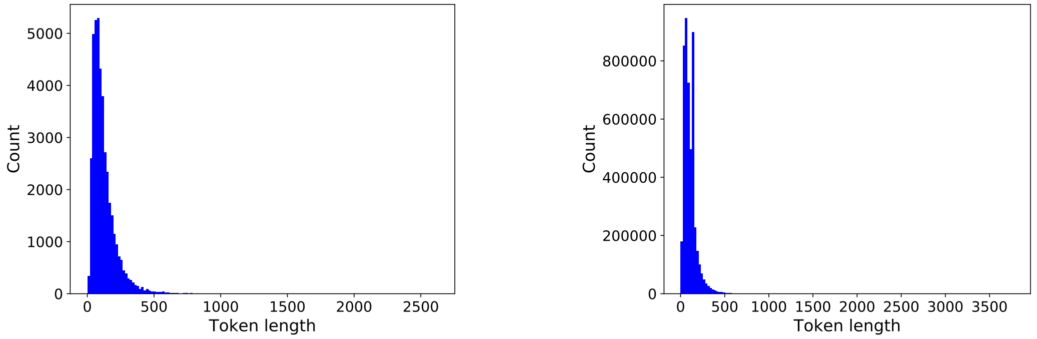
For gathering non cybersecurity related text, we search the top 100 most popular sub-reddits at the time of searching (this list can also be found at [22]) using the same method described above. None of the 40 cybersecurity topic focused sub-reddits are in the top 100 most popular sub-reddits.

For gathering posts in general, we perform some filtering of the data before actually labelling and storing each document. In particular, we remove all posts which are marked as **deleted** or **removed**, since those posts do not contain any post text anymore (these posts were either removed by the user who posted it or a Reddit administrator).

For each post, the title and main post are treated separately in the API. In order to construct a document out of each post, we treated the title as the first sentence and the post content as the remainder of the document (i.e., we merged the two pieces of text with a period and space in between to logically separate them for future parsing).

Figure 1 shows the distribution of the collected tagged Reddit text in terms of token (i.e., usable words) length.

Stackexchange Stackexchange [27] is a group of Q&A websites whose topics of discussion are wide ranging; but the most popular sites are developer and programming sites such as Stackoverflow. The sites are self moderating in that registered user's can upvote and downvote posts. Each site also allows the users who post to identify posts using topic tags.



(a) Cybersecurity documents.

(b) Non cybersecurity documents.

Fig. 2: Stackexchange token data histogram.

We used multiple Stackexchange sites as text sources (see [22] for the full list). For each of these Stackexchange sites, we gathered the top (defined by most upvoted) 10,000 posts for each month since the inception of the given Stackexchange site.

Next, we defined a list of cybersecurity related topic tags across different security and technology related Stackexchange sites (this list of cybersecurity terms, we labelled each post as being cybersecurity related if it used any of the tags in our list and otherwise it was labelled as not cybersecurity related.

As with Reddit, for each post we queried, we merged the title and main post text into a single document.

Figure 2 shows the distribution of the collected tagged Stackexchange text in terms of token (i.e., usable parsed word) length.

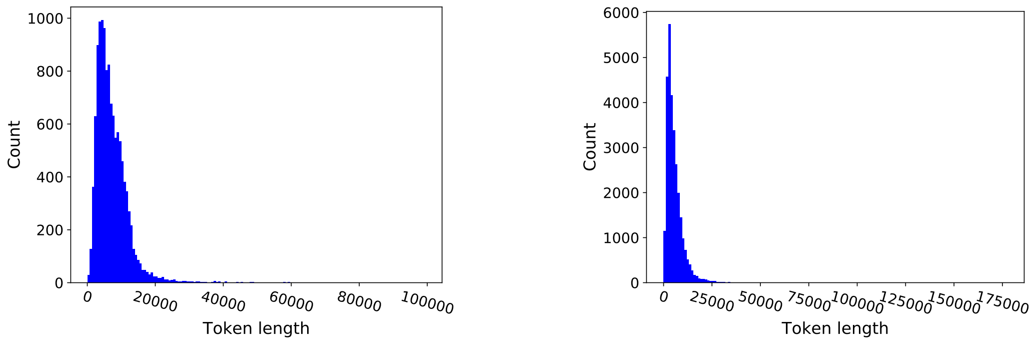
Arxiv Arxiv [4] is an open access repository of e-prints (including papers before peer-review and after peer-review). The repository includes scientific papers on wide ranging topics including computer science, mathematics, physics, statistics, and economics. Each paper comes with one more topic labels and can be downloaded in the form of a PDF.

The methodology for gathering cybersecurity labelled text is as follows. We used a seed list of cybersecurity terms and topics in order to search Arxiv (the word list is provided in [22]). Of the resulting papers returned in the search, if any of the tags are **cs.CR** (which broadly is defined as computer science regarding cryptography and cybersecurity), then we download that pdf and tag the document as cybersecurity related.

For gathering non cybersecurity related documents from Arxiv, we searched all of the remaining non **cs.CR** categories and chose the top 100 (i.e., most relevant) papers from each of those categories; any of these papers with a **cs.CR** were not downloaded, since those documents would be cybersecurity related.

The last step involves some text cleaning, which is specific to Arxiv, since all of the documents are PDFs. First, we remove all non-English documents (some of the downloaded technical documents were in a variety of other languages). Non-English documents were found using **langdetect** [16]. With non-English documents, the majority of the text was non-English, therefore the full document was not used. In future work, we may instead translate the non-English documents and use them as well. Next, we use **tika** to parse the PDF's into raw text. In some cases, **tika** is unable to parse the PDF's (in which case we can not use those documents). Figure 3 shows the distribution of the collected tagged Arxiv text in terms of token (i.e., usable word) length.

In Figure 3, we see that the average document length for Arxiv is in the thousands of words. In Figure 1, we saw that the Reddit average post length is less than 100 words, in contrast to Arxiv. In Figure 2, we observe that the average StackExchange post length is approximately 100. Across all three text sources, the difference in average document length between cybersecurity and non cybersecurity documents is marginal. The most significant difference between the cybersecurity and non cybersecurity documents is that there are many more non cybersecurity documents than cybersecurity documents.



(a) Cybersecurity documents.

(b) Non cybersecurity documents.

Fig. 3: Arxiv token data histogram.

2.2 Text preprocessing

In order to use each document in the various classification algorithms, it is necessary to vectorize each document. In order to standardize all documents so that this vectorization process is consistent, we preprocess each document in a variety of ways. Specifically we

Table 2: Pre-processed text showing the number of cleaned and tokenized documents that are not empty (here by empty we mean having no detectable English words).

Source	cybersecurity	non cybersecurity
Reddit	163,739	4,129,605
Stackexchange	41,162	4,842,459
Arxiv	12,132	289,969

remove URL's, non ascii characters, code tags, all HTML tags, and excessive whitespace. Table 2 shows the total number of usable documents after we have cleaned the text.

2.3 Vectorizer

For this research, we used a term frequency - inverse document frequency (TF-IDF) vectorizer found in *scikit-learn* [21]. TF-IDF vectorization is used to show how common (or important) a given word is in the input document. In particular, it weights words more heavily that occur at a greater frequency. The first step in creating a consistent vectorization method across all documents was compiling the more relevant and popular English words to use for this vectorizer. To this end, we used two English word lists and a custom list of cybersecurity terms. We used *google20kwords* from [9] and *30k.txt* from [8].

We merge all of these lists, remove all duplicates, and remove all entries that are not English words. To determine whether to remove or keep a word, if any of the three methods *nltk* (see [7]) words, nltk wordnet, or the python module *enchant* consider a word to be English, we include it in the dictionary. Pronouns and nouns, such as names and company names, are not considered English words. The resulting English dictionary has a length of 24,538 (the word list is provided in [22]). Non English words were not vectorized because the vast majority of the text we gathered was English and using this fixed English dictionary means that non English words are automatically removed. Future work may include handling foreign language words.

We then fit a TF-IDF vectorizer to this dictionary, save it with 32 bit precision (in order to reduce memory costs), and use this vectorizer for computing the inputs to all machine learning classifiers. The TF-IDF vectorizer can be used to vectorize sentences, paragraphs, or entire documents. In our case, we vectorized each document using TF-IDF. We do not remove any stop words during this vectorization because in simple grid search experiments, removing stop words increased accuracy in some cases, while in other cases it decreased accuracy (both training and testing accuracy). Therefore, since removing stop words or not was not clearly motivated, we did not perform this extra step.

2.4 Machine learning classifiers

We test a total of 6 different types of machine learning classifiers. For all of these classifiers, we use a fixed TF-IDF vectorizer that has been fitted to the English dictionary described in the previous section. Specifically, we use *Decision Tree*, *Random Forest*, *Logistic*, *LinearSVC*, and *Multi-layer Perceptron* (MLP). Each of these five models were provided in the python module **scikit-learn** (version 0.22.0) [21]. We also build a Deep Neural Network (DNN) model using **tensorflow** (version 2.3.0).

For the Decision Tree classifier, we set the maximum depth to be 100, to reduce the real time computational cost. For the DNN model, we specify all of the parameters and hyper-parameters used in Section 2.4. For all other models, we used default parameters.

Deep Neural Network We build a simple multi-layer neural network with one hidden layer using tensor flow; we use a sequential model with 3 layers. The first layer has an input equal to the length of English dictionary we used (24,538 words), with 10,000 nodes. The second layer has 1,000 nodes and the last layer has 100 nodes. The output layer has 2 nodes. The output layer uses an activation function *softmax* and every other layer uses *relu*.

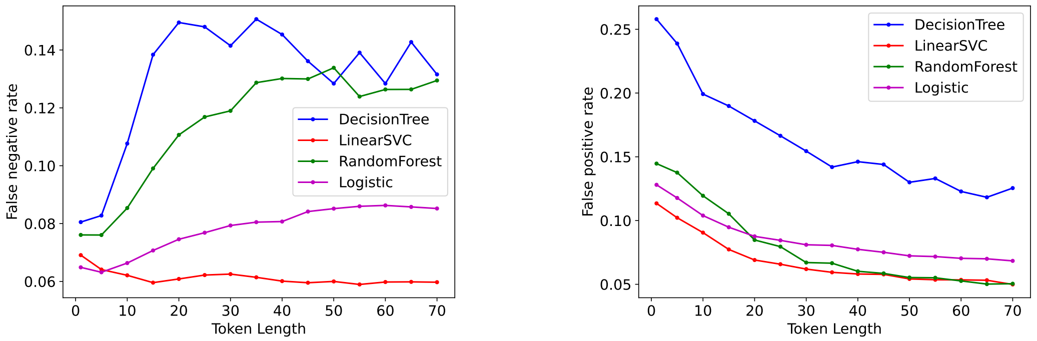
The number of epochs we use is not fixed when training the model. Instead, we train the model until a certain threshold training accuracy has been reached. In our experiments we try two different accuracy thresholds 0.95 and 0.99. The number of samples per iteration was fixed to 4,000. We use *sparse-categorical-crossentropy* for the model loss function, the model metric is *accuracy*, and *Adam* [15] as the model optimizer. To speed up the training and validation time, we also set the flags *workers* to 400 and *use-multiprocessing* to True.

Since we try two different accuracy thresholds (0.95 and 0.99), we actually train 2 different deep neural network models. Thus, in total we train 7 different models in the following section.

3 Experiments

In this section we investigate some of the relevant parameters, both of the text and the machine learning models, that influence the training and testing accuracy.

First, we perform some simple grid-search optimizations of some of the model hyperparameters. Although not shown here for the sake of space, we found that these hyperparameters generally converge to reasonable performance after some number of iterations and / or model size. Second, we investigate how the minimum word length of the data changes both the training and the testing accuracy. In these first two steps, we generally had to run many instances of each algorithm and therefore typically we would select a subset



(a) False negative rate as a function of token length

(b) False positive rate as a function of token length

Fig. 4: False negative and false positive rates as a function of token length (i.e. usable word length) for the Reddit labelled text

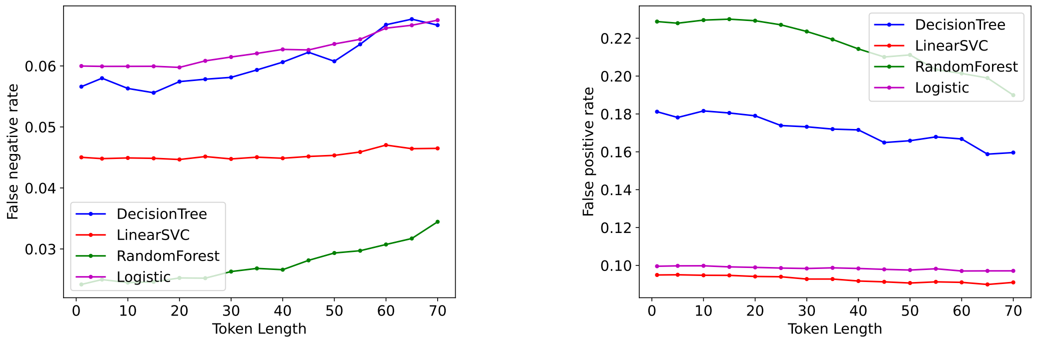
of the data to train and validate on. In the third step, we train on half of each of the text source's datasets and validate the models on the other half (in some cases, due to RAM limitations, we trained on 3/8 of the data and validated on 5/8). Next, we validate how each of these trained models performs on disparate text sources that they were not trained on (e.g., validating a model trained on Reddit text using text from Stackexchange). Finally, we combine all of these models into a unified Python tool called CTC. We show that CTC classifies large numbers of text documents with relatively low error rates.

As shown in Table 1, the labelled data we have gathered is quite skewed towards being made of mostly not cybersecurity documents. This is not unexpected, but it means that training a machine learning model on that dataset will result in unequal weighting of the importance of the classification tasks. In an attempt to correct this towards an evenly balanced dataset, for each machine learning model we use the following class weighting rule. If we have n_c cybersecurity documents and n_{nc} not cybersecurity documents, then the class weight for not cybersecurity is 1 and the class weight for cybersecurity is $\frac{n_{nc}}{c}$.

In the remainder of the article we use FN to denote false negative rate and FP to denote false positive rate.

3.1 Token Length

In this section, we determine the behavior of several of the classifiers as a function of the minimum token length used on the training and validation datasets. Across all three text sources we select random subsets of the full training data to reduce the overall needed computation time. The procedure we used was to eliminate all documents with usable token lengths less than N (N is plotted on the x-axis of the plots shown in the remaining



(a) False negative rate as a function of token length

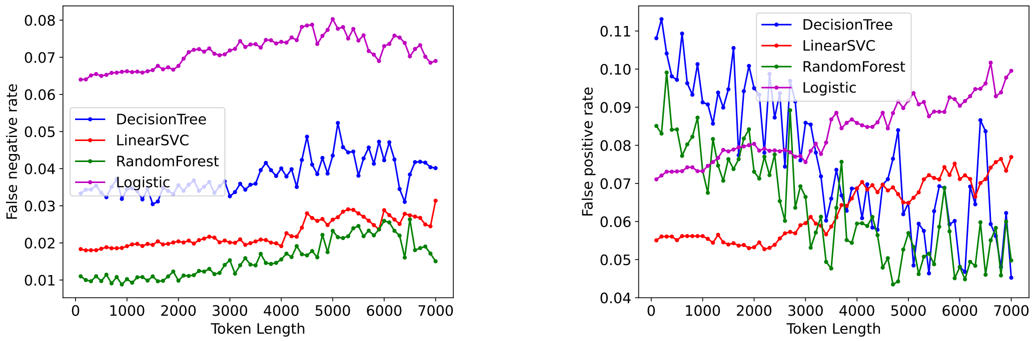
(b) False positive rate as a function of token length

Fig. 5: False negative and false positive rates as a function of token length for the Stackexchange labelled text

subsections), train the model on the training data, and then validate (and plot) the accuracy using the unseen validation data (both the validation data and training data had at least *N* usable tokens in each document). To save space, we show these results for 4 different machine learning classifiers, *Decision Tree*, *LinearSVC*, *Random Forest*, *Logistic*, on each of three text sources.

Reddit For Reddit, we set both the training and validation datasets to have $40,000$ Cybersecurity labelled documents and $50,000$ non Cybersecurity labelled documents. In Figure 4, we plot the false negative and false positive rate as a function of token length for four different classifiers. Figure 4 on the left shows an increase in the false negative rate for the Random Forest Decision Tree and Logistic classifiers as token length increases, while LinearSVC false negative rate remains relatively unchanged. Figure 4 on the right shows a decrease in the false positive rate as a function of token length across all 4 classifiers.

Stackexchange For Stackexchange, we set both the training and validation datasets to have $20,581$ Cybersecurity labelled documents and $50,000$ non Cybersecurity labelled documents. In Figure 5, we plot the false negative and false positive rate as a function of token length for four different classifiers. Figure 5 on the left shows that Random Forest, Logistic, and Decision Tree models increase in false negative rate as token length increases. Figure 5 on the right shows that the false positive rate decreases as a function of token length for DecisionTree and RandomForest, while LinearSVC and Logisitc classifiers remain relatively unchanged.



(a) False negative rate as a function of token length

(b) False positive rate as a function of token length

Fig. 6: False negative and false positive rates as a function of token length for the Arxiv labelled text

Arxiv For Stackexchange, we set both the training and validation datasets to have 1,000 cybersecurity labelled documents, and 3,000 not cybersecurity labelled documents. In Figure 5 we plot the false negative and false positive rate as a function of token length for 4 different classifiers. Figure 5 on the left shows an increase in false negative rates as a function of usable token length for each of the 4 classifiers. Figure 5 on the right shows that DecisionTree and RandomForest false positive rates decrease as a function of token length, while LinearSVC and Logistic classifiers increase in false positive rates as a function of token length. Since the average usable word length of the documents from Arxiv are very large, see Figure 3, the manner in which the error rates change as a function of token length for the Arxiv source is not as important as for Reddit and Stackexchange sources.

To train the machine learning models, we want to not include documents with too few tokens because the topic being discussed may be ambiguous or unclear. However, we want relatively balanced error rates between false negative and false positive. Most importantly, we want our models to be able to classify a wide range of internet text. For this reason, we would want to train on data with smaller token lengths. Given these factors, we conclude that a minimum usable token length of 10 is reasonable for training the models in the remaining experiments.

3.2 Validation

Next, we train all seven of the machine learning models described in Section 2.4 on each of the three labelled text sources (Reddit, Stackexchange, Arxiv), resulting in a total of 21 distinct trained models. Following the token length experiments of Section 3.1, we use only labelled data which has word length (token length) of at least 10 (this reduces the

Table 3: Cross validate DNN at 0.99

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.0505	0.1438
Reddit 1/2	Stackexchange		0.2903	0.1115
Reddit 1/2	Reddit 1/2	0.9932	0.0177	0.1801
Stackexchange 1/2	Arxiv		0.0007	0.7711
Stackexchange 1/2	Stackexchange 1/2	0.991	0.0083	0.3081
Stackexchange 1/2	Reddit		0.0666	0.6837
Arxiv 1/2	Arxiv 1/2	0.9933	0.0172	0.0236
Arxiv 1/2	Stackexchange		0.2353	0.2007
Arxiv 1/2	Reddit		0.1068	0.4041

Table 4: Cross validate DNN at 0.95

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.0867	0.0886
Reddit 1/2	Stackexchange		0.4485	0.0499
Reddit 1/2	Reddit 1/2	0.9573	0.0397	0.1043
Stackexchange 1/2	Arxiv		0.0013	0.6222
Stackexchange 1/2	Stackexchange 1/2	0.9644	0.0317	0.1415
Stackexchange 1/2	Reddit		0.0662	0.5342
Arxiv 1/2	Arxiv 1/2	0.9578	0.0182	0.0748
Arxiv 1/2	Stackexchange		0.2052	0.2393
Arxiv 1/2	Reddit		0.0714	0.4964

total amount of labelled data we use from the initial corpus shown in Table 2). For each model type and each text source, we train the model on one half of the labelled data, and then validate on the other half (in some cases, where training on one half of the data is too computationally costly, we train on 3/8 of the data).

Since we want these machine learning models to be applied to text that may not come form Arxiv or Stackexchange or Reddit, one way to evaluate each of these models is to predict the topic of text from the two sources the model was not trained on; for example predicting the topic of the labelled Arxiv and Stackexchange dataset given the model was trained on Reddit text. In this section we show the validation accuracy (i.e., testing accuracy) and cross validation accuracy for each of the 21 trained models. In particular, Tables 3, 4, 5, 6, 7, 8, 9 show the cross validation results. In these tables we show the training accuracy as well as the cross validation false negative and false positive rates. Under the training accuracy column, we only get one training accuracy entry for each of the three text sources (the other rows are validation accuracy results), which means that six of the entries in that column will be empty.

As a general summary of the cross validation results, we observe the trend that the false negative and false positive rates for the models trained on a source and then validated

Table 5: Cross validate Logistic

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.27	0.0101
Reddit 1/2	Stackexchange		0.5136	0.0282
Reddit 1/2	Reddit 1/2	0.9509	0.0496	0.0849
Stackexchange 3/8	Arxiv		0.0255	0.2211
Stackexchange 3/8	Stackexchange 5/8	0.9599	0.0403	0.1037
Stackexchange 3/8	Reddit		0.1162	0.4089
Arxiv 1/2	Arxiv 1/2	0.965	0.0296	0.0557
Arxiv 1/2	Stackexchange		0.0347	0.6919
Arxiv 1/2	Reddit		0.0035	0.8513

Table 6: Cross validate RandomForest

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.0037	0.7019
Reddit 1/2	Stackexchange		0.0271	0.7481
Reddit 1/2	Reddit 1/2	0.9998	0.0018	0.6921
Stackexchange 1/2	Arxiv		0.0001	0.9736
Stackexchange 1/2	Stackexchange 1/2	1.0	0	0.9811
Stackexchange 1/2	Reddit		0.0004	0.982
Arxiv 1/2	Arxiv 1/2	1.0	0.013	0.0374
Arxiv 1/2	Stackexchange		0.001	0.9705
Arxiv 1/2	Reddit		0.0001	0.9914

on the same text source (not the same data, just the same text source e.g., Reddit) are usually relatively low (less than ten percent). The exceptions to this are RandomForest and Decision Tree for Reddit and Stackexchange sources. We also observe varied error rates for the cross validation experiments. In particular, we usually see an asymmetry in the error rates - i.e., the false negative rate is very high and the false positive rate is very low or the reverse.

3.3 Cybersecurity Topic Classification (CTC) tool

Now we combine all of these trained models (the validation data for these models was shown in the previous section) into a single NLP tool for cybersecurity topic modeling. In particular, for a given set of documents, we vectorize using the TF-IDF vectorizer used in all of these experiments. Then, we run all 21 models on those documents and report the results. To come to a final decision as the output of the tool, we take the majority vote on the output of the 21 ML models. This is a reasonable approach because it is simply taking the consensus of all of the trained models. This means that no outlier can change the tool's decision making process, making the system robust against outlier predictions.

Table 7: Cross validate LinearSVC

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.1482	0.0463
Reddit 1/2	Stackexchange		0.4543	0.0397
Reddit 1/2	Reddit 1/2	0.9566	0.0463	0.0992
Stackexchange 3/8	Arxiv		0.0041	0.5392
Stackexchange 3/8	Stackexchange 5/8	0.9673	0.0335	0.1385
Stackexchange 3/8	Reddit		0.0928	0.4993
Arxiv 1/2	Arxiv 1/2	0.9882	0.0147	0.0302
Arxiv 1/2	Stackexchange		0.0205	0.6619
Arxiv 1/2	Reddit		0.0039	0.8224

Table 8: Cross validate DecisionTree

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.2338	0.2309
Reddit 1/2	Stackexchange		0.4181	0.3088
Reddit 1/2	Reddit 1/2	0.9749	0.0398	0.3233
Stackexchange 1/2	Arxiv		0.0242	0.2986
Stackexchange 1/2	Stackexchange 1/2	0.9854	0.0188	0.4127
Stackexchange 1/2	Reddit		0.0248	0.6518
Arxiv 1/2	Arxiv 1/2	1.0	0.0249	0.0641
Arxiv 1/2	Stackexchange		0.0186	0.7244
Arxiv 1/2	Reddit		0.0132	0.7857

CTC Error rates Here, we show that the CTC tool on average performs better than any of the individual 21 trained models.

We use the labelled dataset of [28] for this experiment. In particular, we use both the training and testing dataset from [28] in this validation test. The dataset has 4 different labels: 1 (World), 2 (Sports), 3 (Business), 4 (Sci/Tech). Since 4 can include cybersecurity content, we entirely remove 4, and then merge classes 1, 2, and 3 into non cybersecurity. Collectively, we call this data source *ag-news*. We also used [3] as a data set, which is definitely not cybersecurity related. We call this dataset *philosophy*. Both *ag-news* and *philosophy* are used as large datasets which provide an idea of the *false positive* rate.

Next, we want to develop a labelled dataset for cybersecurity related text to validate the performance of CTC. To this end, we pull a random subset of 100 documents from seven internet forums/discussion blogs and then hand label the topic's of each of these 700 documents. Note that several of these sources are heavily cybersecurity related, which we want to get a reasonable sample of cybersecurity text with which to validate the models. In several cases, the random documents we accessed did not have any English words or enough English words (i.e., one or two words), so those documents were discarded. In total, from the seven sources, we have 698 labelled documents.

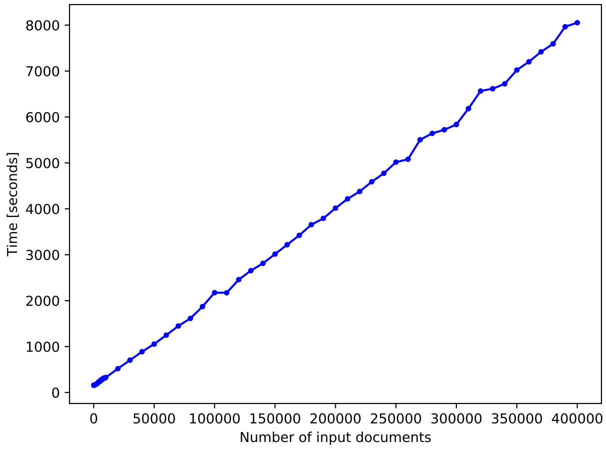


Fig. 7: Number of input documents vs Wall clock time for the CTC tool

Table 9: Cross validate Multi-Layer Perceptron

Training source	Validation source	Training accuracy	Validation FN	Validation FP
Reddit 1/2	Arxiv		0.0343	0.2406
Reddit 1/2	Stackexchange		0.1412	0.276
Reddit 1/2	Reddit 1/2	0.9996	0.0078	0.3183
Stackexchange 3/8	Arxiv		0.0004	0.8492
Stackexchange 3/8	Stackexchange 5/8	0.9995	0.0027	0.5325
Stackexchange 3/8	Reddit		0.0179	0.8125
Arxiv 1/2	Arxiv 1/2	0.9999	0.0147	0.0432
Arxiv 1/2	Stackexchange		0.3002	0.3592
Arxiv 1/2	Reddit		0.117	0.5524

Table 10 shows the number of incorrectly labelled documents when using the CTC tool (the majority vote of the 21 individual models) and when using each of the individual 21 models applied to the new validation data described above. On average, the CTC tool outperforms the individual models across different text sources. Figure 8 shows a more concise version of Table 10, where we aggregate the results across the *cybersecurity* and *non cybersecurity* labelled validation text. Figure 8 shows that while there are some individual models that have very low *false positive or* low *false negative* rates, the majority vote of the 21 models has the lowest overall *false positive and* low *false negative*. For example, we observe that *Arxiv-RandomForest* has a very low false positive rate, but then has a very high false negative rate. Thus, the majority vote mechanism used in the CTC tool is more robust compared to the individual models.

CTC Timing Lastly, we characterize how the CTC tool scales in terms of documents analyzed over time. We pull random subsets of some internet discussion posts of varying length and content and then measure the total wall clock time needed to classify that set of documents. We repeat this process for increasing numbers of input documents.

Figure 7 shows this scaling of documents analyzed over time. We observe that the wall clock time needed to classify *N* input documents has a consistently linear scaling. This scaling is largely due to using multiprocessing for the tensorflow DNN models. The limiting factor of how many documents CTC can ingest is the available RAM on the host computer. For the result shown in Figure 7, the device had 500 Gb of RAM, but we do not characterize exactly where the RAM limit is.

4 Conclusion and Future Work

This article proposed a methodology for gathering labelled English text for the purpose of topic modelling cybersecurity discussions. We then trained multiple machine learning models using this labelled data, and showed that combining these models into a consensus majority voting tool results in both low error rates and scalability in terms of documents classified per hour.

There remain many possible future research avenues:

1. Considering unsupervised clustering methods, which can determine how similar each source of training data is, and therefore also how similar a new input document is to each of these sources can yield lower error rates and lower computation times by using only a single machine learning classifier that has the highest accuracy for that type of document (based on the validation experiments done previously).
2. Using clustering methods to first cluster the corpus of labelled training data into a large number of clusters and then separately training machine learning models on each of those clusters could yield higher accuracy when using all of the models together.
3. Using other vectorization algorithms that make use of a larger dictionary of words, as well as the sequence in which words are used, may improve performance over using a bag of words model.
4. Gathering more labelled data from other online sources, for example from Twitter, Quroa, or Medium, would increase the size of the training set. These sites also offer methods of mining labelled natural language posts, for example [18] and [14] use data from Twitter.
5. Analyzing the explainability of the machine learning algorithms could generate new insight. For example, the decision tree algorithm might provide a more compact and explainable logical decision path for this particular topic modeling task.
6. Using other machine learning algorithms that may be better suited to topic classification tasks, for example Recurrent Neural Networks.

7. Exploring the use of subset's of the 21 machine learning models (with the majority voting mechanism) could improve results and reduce computational time. In particular, some subset may perform better than all 21 models and individual models.
8. Including non English posts / documents may provide additional valuable data once translated.

Acknowledgements

Sandia National Laboratories (SNL) is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. The New Mexico Cybersecurity Center of Excellence (NMCCoE) is a statewide Research and Public Service Project supported center for research, education, and economic development. The authors would like to thank both SNL and NMCCoE for funding and computing system access/support.

References

[1] 2021. URL: `https://pypi.org/project/psaw/` (visited on 06/11/2021).

[2] Rasim Alguliyev, Ramiz Aliguliyev, and Fargana Abdullayeva. "Deep Learning Method for Prediction of DDoS Attacks on Social Media". In: *Advances in Data Science and Adaptive Analysis* 11 (Feb. 2019). DOI: `10.1142/S2424922X19500025`.

[3] Kourosh Alizadeh. *Text-Based Ideological Classification*. `https://github.com/kcalizadeh/phil_nlp`. 2021.

[4] *Arxiv*. 2020. URL: `https://arxiv.org/` (visited on 06/11/2021).

[5] *Arxiv*. 2021. URL: `https://pypi.org/project/arxiv/` (visited on 06/11/2021).

[6] Bogdan Batrinca and Philip C. Treleaven. "Social media analytics: a survey of techniques, tools and platforms". In: *AI & SOCIETY* 30.1 (Feb. 2015), pp. 89–116. ISSN: 1435-5655. DOI: `10.1007/s00146-014-0549-4`. URL: `https://doi.org/10.1007/s00146-014-0549-4`.

[7] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[8] Derek Chuank. *high-frequency-vocabulary*. `https://github.com/derekchuank/high-frequency-vocabulary`. 2020.

[9] furiousapathy. *most-common-english-words*. `https://github.com/furiousapathy/most-common-english-words`. 2020.

[10] Aldo Hernandez-Suarez et al. "Social Sentiment Sensor in Twitter for Predicting Cyber-Attacks Using l1 Regularization". In: *Sensors* 18 (Apr. 2018), p. 1380. DOI: `10.3390/s18051380`.

[11] Jack Hughes et al. “Detecting Trending Terms in Cybersecurity Forum Discussions”. In: *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 107–115. DOI: `10.18653/v1/2020.wnut-1.15`. URL: `https://www.aclweb.org/anthology/2020.wnut-1.15`.

[12] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: `10.1109/MCSE.2007.55`.

[13] Ruth Ikwu and Panos Louvieris. “Monitoring “Cyber Related” Discussions in Online Social Platforms”. In: *International Journal on Cyber Situational Awareness* 4.1 (Dec. 2019), pp. 69–98. ISSN: 2633-495X. DOI: `10.22619/ijcsa.2019.100126`. URL: `http://dx.doi.org/10.22619/IJCSA.2019.100126`.

[14] Rupinder Paul Khandpur et al. “Crowdsourcing Cybersecurity: Cyber Attack Detection Using Social Media”. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. CIKM ’17. Singapore, Singapore: Association for Computing Machinery, 2017, pp. 1049–1057. ISBN: 9781450349185. DOI: `10.1145/3132847.3132866`. URL: `https://doi.org/10.1145/3132847.3132866`.

[15] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: `1412.6980 [cs.LG]`.

[16] *langdetect*. 2021. URL: `https://pypi.org/project/langdetect/` (visited on 06/30/2021).

[17] Triet Huynh Minh Le et al. “Demystifying the Mysteries of Security Vulnerability Discussions on Developer Q&A Sites”. In: *CoRR* abs/2008.04176 (2020). arXiv: `2008.04176`. URL: `https://arxiv.org/abs/2008.04176`.

[18] Richard P. Lippman et al. “Toward Finding Malicious Cyber Discussions in Social Media”. In: *AAAI Workshops*. 2017. URL: `http://aaai.org/ocs/index.php/WS/AAAIW17/paper/view/15201`.

[19] Tamara Lopez et al. “An Anatomy of Security Conversations in Stack Overflow”. In: *41st ACM/IEEE International Conference on Software Engineering*. Aug. 2019, pp. 31–40. URL: `http://oro.open.ac.uk/59243/`.

[20] Shervin Minaee et al. *Deep Learning Based Text Classification: A Comprehensive Review*. 2021. arXiv: `2004.03705 [cs.CL]`.

[21] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[22] Elijah Pelofske. *CTC*. `https://github.com/epelofske-student/CTC`. 2021.

[23] *PRAW*. 2021. URL: `https://praw.readthedocs.io/en/latest/` (visited on 06/11/2021).

[24] *Reddit*. 2020. URL: `https://www.reddit.com/` (visited on 06/11/2021).

[25] Kai Shu et al. “Understanding Cyber Attack Behaviors with Sentiment Information on Social Media”. In: *Social, Cultural, and Behavioral Modeling*. Ed. by Robert

Thomson et al. Cham: Springer International Publishing, 2018, pp. 377–388. ISBN: 978-3-319-93372-6.

[26] *StackAPI*. 2021. URL: `https://stackapi.readthedocs.io/en/latest/` (visited on 06/11/2021).

[27] *Stackexchange*. 2020. URL: `https://stackexchange.com/sites#` (visited on 06/11/2021).

[28] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. "Character-level Convolutional Networks for Text Classification". In: *NIPS*. 2015.

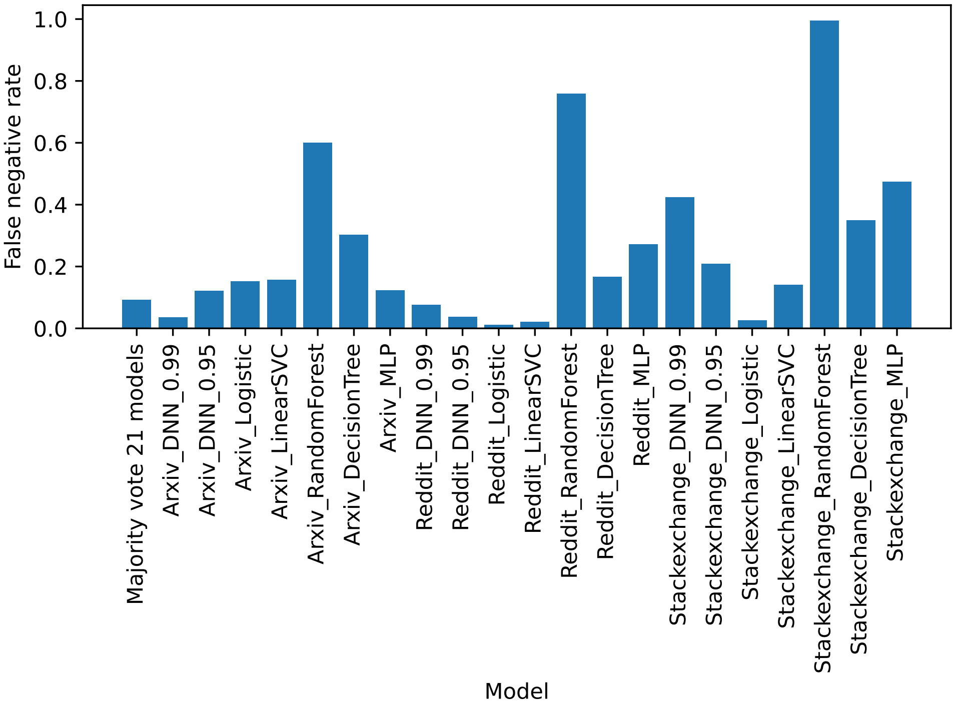
Authors

Elijah Pelofske is working on his B.S. in Computer Science and Mathematics at New Mexico Institute of Mining and Technology. He is interested in the fields of machine learning, quantum computing, cryptography, algorithm design, data science, tensor networks, and novel computing architectures. He has published several journal and conference papers in the fields of quantum computing and machine learning. He has written several novel quantum annealing open source python tools. He won the Los Alamos National Laboratory Distinguished Student award in 2019, and recently received a Best Conference Paper Award at LSSC'21 for joint work on using quantum annealers to create Boolean Hierarchical Tucker Networks.

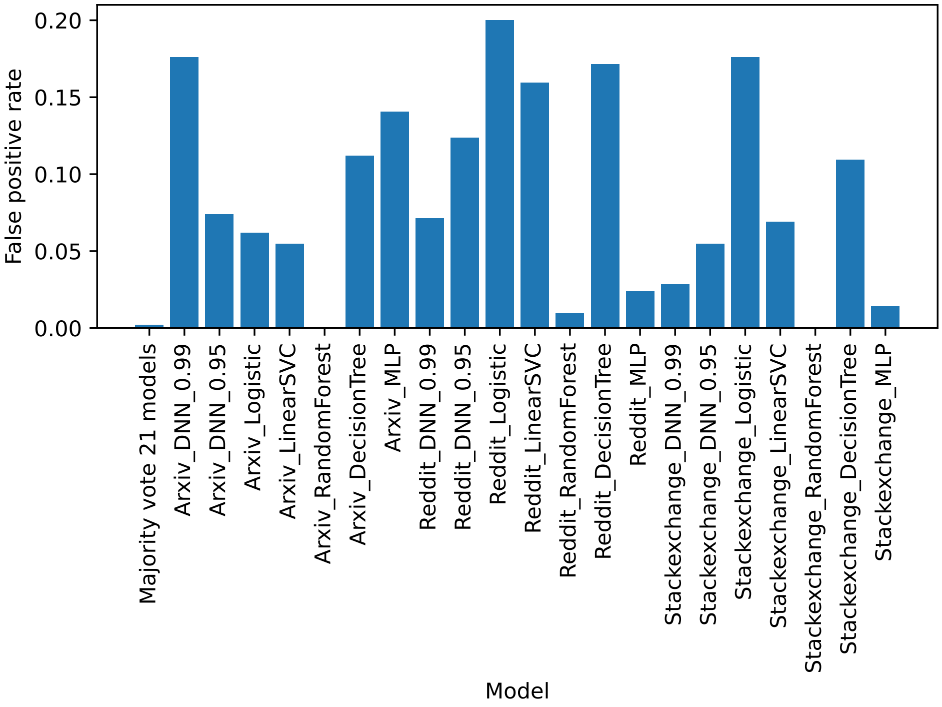
Dr. Lorie M. Liebrock is the Director of the New Mexico Cybersecurity Center of Excellence and the New Mexico Tech Cybersecurity Education Center. Her work with the Cybersecurity Centers leads projects including a Summer Institute with Sandia National Laboratories, an expansion project for Codebreaker Challenge, and an economic development project to help New Mexico companies prepare for Cybersecurity Maturity Model

Certification. In addition, she is the Chair for the Transdisciplinary Cybersecurity graduate programs (Master of Science, Professional Masters, and PhD). She is also a Professor of Computer Science and Engineering. Her background in graduate education comes from years of experience as NMT's Graduate Dean. Dr. Liebrock has extensive research and real-world experience in cybersecurity, as well as parallel and high performance computing. She has published twenty-six journal articles, forty conference and workshop papers, and holds two US patents. Her research includes enterprise-wide cybersecurity, foundations of computer science, information assurance, parallel processing, and visualization with a focus on complex problems that require the integration of many aspects of computer science. This provides many student research opportunities, as she integrates students in her research - from freshmen to Ph.D. candidates. Dr. Liebrock holds both M.S. and Ph.D. in Computer Science from Rice University, B.S. and M.S. in Computer Science from Michigan Technological University, and an Associates degree from Delta Community College.

Vincent Urias is a computer engineer, and Senior Member of Technical Staff in Sandia's Cyber Analysis Research Development Department continuing to make major contributions to Sandia's cyber defense programs, especially in the simulation of complex networks, in developing innovative cyber security methods, and in designing exercise scenarios that test the limits of current network security. This work is helping Sandia's customers anticipate current and emerging security threats and make critical decisions about their investments. Vince and his team use technologies to conduct cyber defense exercises in partnership with the U.S. Department of Defense, and to support national security in collaboration with colleagues at other U.S. Department of Energy national laboratories, Department of Defense national laboratories, and the U.S. military. Vince gives back to the community in a variety of ways, providing guidance and inspiration to college interns in the lab's Center for Cyber Defenders, he supports building computer labs for local organizations and is also helping to create an Urban Wildlife Refuge in Albuquerque's South Valley among other things. Vincent is currently pursuing his Ph.D. in computer science, at New Mexico Tech. He was honored by GMiS with a HENAAC Luminary Award in October of 2016.



(a) False negative rate comparison across models.



(b) False positive rate comparison across models.

Fig. 8: CTC error rate comparison to single ML models

Text source and label	Number of documents	21 models incorrectly labelled	individual models incorrectly labelled vector
Google security blog Cybersec	98	3	[0, 3, 5, 8, 63, 22, 4, 3, 0, 0, 0, 73, 11, 22, 47, 20, 3, 10, 98, 37, 54]
Darkreading Cybersec	98	4	[2, 9, 6, 8, 41, 23, 11, 6, 2, 0, 1, 83, 21, 29, 27, 17, 1, 11, 97, 30, 34]
Fireeye Cybersec	94	6	[2, 11, 5, 6, 56, 27, 3, 3, 3, 2, 2, 71, 10, 18, 51, 20, 3, 19, 94, 30, 52]
Ycombinator Cybersec	2	2	[1, 1, 2, 2, 2, 1, 1, 2, 2, 0, 2, 2, 1, 2, 2, 2, 1, 2, 2, 1, 2]
schneier Cybersec	35	5	[2, 7, 14, 12, 29, 7, 5, 4, 2, 1, 1, 20, 4, 7, 19, 13, 1, 7, 34, 14, 24]
toms-forum Cybersec	1	0	[0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1]
thehackernews Cybersec	92	19	[8, 20, 32, 30, 60, 47, 28, 13, 7, 2, 3, 69, 23, 35, 32, 16, 2, 10, 92, 34, 32]
Google security blog Not Cybersec	2	2	[2, 2, 2, 2, 0, 1, 2, 1, 2, 2, 2, 0, 1, 0, 0, 0, 2, 1, 0, 0, 1]
Darkreading Not Cybersec	1	0	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
Fireeye Not Cybersec	5	0	[2, 1, 0, 0, 0, 3, 1, 0, 0, 2, 1, 0, 2, 0, 1, 0, 3, 1, 0, 2, 0]
Ycombinator Not Cybersec	98	0	[22, 3, 2, 0, 0, 2, 14, 7, 14, 22, 15, 0, 19, 0, 1, 2, 22, 6, 0, 5, 0]
schneier Not Cybersec	65	15	[30, 19, 20, 21, 0, 41, 27, 11, 17, 31, 24, 4, 36, 6, 5, 15, 37, 19, 0, 38, 5]
toms-forum Not Cybersec	99	0	[14, 5, 0, 0, 0, 0, 12, 10, 16, 20, 19, 0, 12, 4, 5, 6, 6, 0, 0, 1, 0]
thehackernews Not Cybersec	8	0	[4, 1, 2, 0, 0, 0, 3, 1, 2, 6, 5, 0, 1, 0, 0, 0, 4, 2, 0, 0, 0]
ag-news Not Cybersec	95698	521	[13957, 8143, 1119, 1670, 0, 3832, 12603, 285, 862, 6435, 4565, 104, 4920, 345, 939, 1638, 14158, 9228, 112, 5929, 1319]
philosophy Not Cybersec	360808	434	[37654, 23525, 284, 454, 0, 1074, 35416, 34027, 52652, 13267, 23344, 59, 4831, 3913, 79093, 68829, 23065, 27708, 12, 2631, 4992]

Table 10: Number of incorrectly labelled documents across multiple text sources. The first column shows the labelled text source. Second column shows the total number of documents from that text source. Third column shows the total number of documents that the CTC tool (i.e. the majority vote of the 21 models) incorrectly labelled. Fourth column shows the number of incorrectly labelled documents by each of the 21 individual models.

REPRESENTATION LEARNING AND SIMILARITY OF LEGAL JUDGEMENTS USING CITATION NETWORKS

Harshit Jain and Naveen Pundir

Department of Computer Science and Engineering, IIT Kanpur, India

ABSTRACT

India and many other countries like UK, Australia, Canada follow the 'common law system' which gives substantial importance to prior related cases in determining the outcome of the current case. Better similarity methods can help in finding earlier similar cases, which can help lawyers searching for precedents.

Prior approaches in computing similarity of legal judgements use a basic representation which is either abag-of-words or dense embedding which is learned by only using the words present in the document. They, however, either neglect or do not emphasize the vital 'legal' information in the judgements, e.g. citations to prior cases, act and article numbers or names etc.

In this paper, we propose a novel approach to learn the embeddings of legal documents using the citationnetwork of documents. Experimental results demonstrate that the learned embedding is at par with the state-of-the-art methods for document similarity on a standard legal dataset.

KEYWORDS

Representation Learning, Similarity, Citation Network, Graph Embedding, Legal Judgements.

1. INTRODUCTION

It is hard to imagine modern human civilization without the 'rule of law'. Strangely, not enough effort has gone into using digital information processing and retrieval techniques in the legal domain. While digital archiving of legal information has indeed happened, the rich inter-relationships and dependencies between legal documents have not been adequately captured. The size of the information base precludes using manual methods to do this. It has to be done algorithmically.

Legal information can be put into two broad categories that are actually intimately connected. First, we have the laws that are codified in the form of articles of a constitution, acts, statutes, rules, procedures etc. passed or laid down by various bodies (like parliaments, legislatures, panchayats, local bodies, etc.) who are empowered to do so. And second, we have the vast body of judgements or decisions that interpret and apply the law or rules when two or more partiesare on opposing sides and approach an adjudicating body - typically a court. While laws, rules and procedures change relatively slowly the body of judgements grows much more rapidly.

Most countries like the Commonwealth countries (UK, Australia, Canada, India, etc.), USA and several others follow the *common law system*. In this system substantial importance is given to prior related cases to determine the outcome for the current case, a principle known as *stare*

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 23-30, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112302

decisis - literally ‘let the decision stand’. This means judgements in cases similar to the current case are a critical component for decision making. So, it is important to invent robust similarity metrics and ways to organize documents such that similar documents cluster together and are easily accessible.

In this paper, our main aim is to calculate the similarity between legal documents by learning *suitable embedding vector space*, which captures their semantic meaning and the relationships among them. We solve this problem by transforming each legal document to a vector R^d (d isthe dimension of a suitably chosen vector space) and then calculate the cosine similarity between the vectors.

2. CHALLENGES

Measuring the similarity of legal documents is non-trivial as legal documents are quite complex in nature. They contain different kinds of information like facts of the case, citations, mention and interpretations of laws/statutes, legal reasoning, opinions on social and economic matters all of this often in a single judgement. Clearly, different similarity metrics will apply depending on the kind of similarity that is sought by the end user. In this paper the similarity we are concernedwith is guided by the *common law system*. So, we look for earlier judgements that are pertinent (i.e. similar) to the current case.

Legal documents differ from other documents in the following ways:

- Legal documents are usually long and contain complex sentences and often the required information is embedded in a mass of text making it difficult to find.
- Legal documents contain legal jargon and citations to prior cases, laws, articles, acts, etc. which is important information that is external to the document itself.
- A legal document may pertain to more than one legal issue and thus may be useful in cases that are not similar on the surface.

Standard methods like: bag-of-words vectors, document vectors constructed using word embeddings in vector spaces, latent Dirichlet allocation (LDA) to assign topic words etc. ignores or does not give the importance to the specific legal information in the judgement (like acts, articles and citations to prior cases). So, we tried to capture this legal information in the form of a citation network (weighted graph) and learned the embedding of legal judgements using that graph.

3. RELATED WORK

KUMAR ET AL. [1] proposed multiple approaches to find similar legal judgements by using the techniques of information retrieval (IR) and search engines. They analyzed four approaches to find similarity between documents and found that *legal term cosine similarity* performed well. They analysed the following four approaches: **(1) All term cosine similarity**: In this approach, they represent each document as a vector using tf-idf and the similarity is computed using cosine similarity. The tf–idf is the product of two statistics, term frequency and inverse document frequency. **(2) Legal term cosine similarity**: They only consider ‘legal’ words in the documents - those that occur in a legal dictionary. Then represent each document as a vector using tf-idf and the legal dictionary and similarity is again computed using cosine similarity. **(3) Bibliographic coupling (BC) similarity**: They made a citation graph of the documents by finding the citations using regular expressions and the similarity is measured using Bibliographic coupling (BC). The bibliographic coupling similarity score between two legal documents is equal to the number of common out-citations. Out-citations for document *d* are the documents that are cited by document *d*. **(4) Co-citation (CC) similarity**: Co-citation (CC) similarity is the number of common in-citations in the citation graph. In-citations for document *d* are the documents that cite document *d*.

KUMAR ET AL. [2] presented a hybrid approach, based on the text and the citation network. They proposed an improved approach to find the similarity between the legal judgements using link-based similarity. Through the experiments, they observed that it is possible to find similar judgements by using citations. For this, they introduced the notion of paragraph-links to improve the efficiency of link-based similarity methods. To calculate paragraph-links, as each document constitutes many paragraphs and these paragraphs are considered as a separate entity. Now a similarity score across all pairs of paragraphs is measured. For measuring similarity between a pair of paragraphs, they first represented the paragraph into the tf-idf embedding and then computed the cosine similarity between them. A paragraph-link is inserted between judgements A and B if any similarity score between paragraphs is greater than a threshold.

THENMOZHI ET AL. [3] proposed an approach to retrieve precedent cases that are relevant to the current case. They used three methods to find the similarity score. **(1) With concepts and tf-idf scores**: In this approach, they extracted all forms of nouns (NN (singular noun), NNS (plural noun), and NNP (proper noun)) and considered only these nouns to get the tf-idf representation of the document. The similarity score was calculated using the cosine similarity between the tf-idf representation of the documents. **(2) With concepts, relations, and tf-idf scores**: In this approach, they extracted all forms of nouns (NN, NNS, and NNP) and verb (VB (verb, base), VBD (verb, past tense), VBZ (verb, 3rd person), VBN (verb, past participle)) as well to get the tf-idf representation of the documents. The similarity score was the cosine similarity between the tf-idf representation of documents. **(3) With concepts, relations, and Word2Vec**: In this approach, they extracted all forms of nouns (NN, NNS, and NNP) and verb (VB, VBD, VBZ, VBN) and used Word2Vec [4] to get a vector for each extracted word and represented each document as vector $\in$ R^{300} by taking the average of all the vectors of the vocabulary words present in the document.

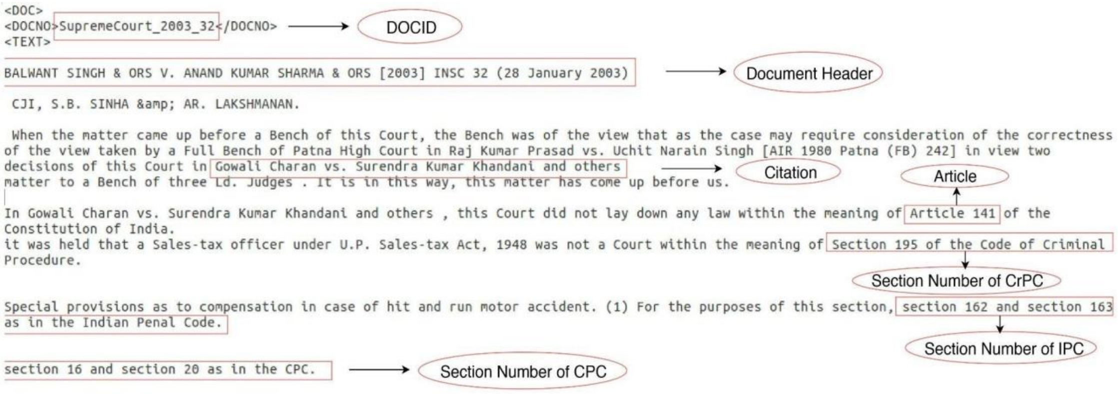
MANDAL ET AL. [5] performed extensive experiments on legal documents using tf-idf, similarity measures (such as topic modeling), and neural network models (such as the average of word embeddings Word2Vec and document embeddings Doc2Vec [6]. They have shown thatthe Doc2Vec based technique significantly out-performs baseline techniques which utilize both text-based (TF-IDF) and network-based similarity measures (Bibliographic coupling similarity).

4. PROPOSED METHOD

4.1. Dataset Preparation

For the experiments, we collected a set of 30,016 legal judgements of the Supreme Court of India over a period from 1950 to 2012. These documents were crawled from the Legal Information Institute of India[1] that provides free access to various databases related to the Indianlegal system. An example of a legal court case (document) is shown in Figure 4.1.

In addition, we crawled the dataset of Articles defined in the Constitution of India and important Acts like IPC (Indian Penal Code), CrPC(Code of Criminal Procedure), and CPC(Code of Civil Procedure)[2]. This dataset contains the information against each section's number of acts (IPC, CrPC, CPC) and articles.



```
<DOC>
<DOCNO>SupremeCourt_2003_32</DOCNO>
<TEXT>

BALWANT SINGH & ORS V. ANAND KUMAR SHARMA & ORS [2003] INSC 32 (28 January 2003)

 CJI, S.B. SINHA &amp; AR. LAKSHMANAN.

 When the matter came up before a Bench of this Court, the Bench was of the view that as the case may require consideration of the correctness
of the view taken by a Full Bench of Patna High Court in Raj Kumar Prasad vs. Uchit Narain Singh [AIR 1980 Patna (FB) 242] in view two
decisions of this Court in Gowali Charan vs. Surendra Kumar Khandani and others
matter to a Bench of three Ld. Judges . It is in this way, this matter has come up before us.

In Gowali Charan vs. Surendra Kumar Khandani and others , this Court did not lay down any law within the meaning of Article 141 of the
Constitution of India.
it was held that a Sales-tax officer under U.P. Sales-tax Act, 1948 was not a Court within the meaning of Section 195 of the Code of Criminal
Procedure.

Special provisions as to compensation in case of hit and run motor accident. (1) For the purposes of this section, section 162 and section 163
as in the Indian Penal Code.

section 16 and section 20 as in the CPC.
```

Figure 4.1. Example of a legal judgement marked with DOCID, Document Header, andCitation Information.

4.2. Our Approach

Our goal is to compute the similarity between legal documents. In order to find the similarity, we map each document in our corpus to a vector in R^d (where *d* is the dimension of the vector space) and compute their cosine similarity. The useful legal information is very sparse in the document and is completely dominated by details of the case and other kinds of information that are not useful in deciding whether or not this judgement can be used as a precedent. Since court judgements give us useful information in the form of citations, section numbers of acts and articles, rule numbers of procedures, etc, we exploit this information along with the important words of the document to learn the embeddings. These embeddings are learned using the following sub-steps: (1) Information Extraction (2) Graph Construction (3) Graph Embedding.

4.2.1. Information Extraction

The text of a judgement apart from the judgement itself contains other useful information like references to acts and article numbers and citations to similar prior cases. An example is shown in Figure 4.1. This information can be used to compute similarity among documents. Specifically, we find the article, section numbers, and citations (e.g. Oma Ram v. State Of Rajasthan and ors.) mentioned in the document using regular expressions. All this information isstored corresponding to each document in a table where every document is given a unique DOCID.

Each citation extracted from a document is mapped to the corresponding document title e.g. the extracted citation “Atma Ram vs. the State of Punjab” will be mapped with the document header “Atma Ram Kumar vs. State of Punjab and ors”. Since the citation and title cannot be matched directly, to solve this problem without manual intervention, we compute the score (given below) for each citation with respect to all the document titles. The score is calculated by

$$score = \frac{len(lcs(\text{cite, doc-header}))}{len(\text{doc-header})}$$

where *lcs* is the longest common subsequence and *len* is the number of characters in the string.

The values will be in the range [0-1].

For titles that have a score at least equal to a threshold, the citation is mapped to the document title with the highest score. If there is no such title then the citation is not mapped and is ignored. We don't want to include more false positives and also at the same time do not want to miss the true positives so we choose the threshold value as 0.6 based on our manual checks.

With this approach, we were able to map more than 55% of all the citations in all the documents.

4.2.2. Graph Construction

Citation information can be nicely captured if we convert our set of documents into an undirected weighted graph where nodes are the individual legal judgements and an edge defines the similarity between the two judgements. An example of a citation graph is shown in Figure

The resulting graph has 20,469 nodes and 109,892 edges. Some nodes are dropped since they contain no citations and also no judgement cites them. To further augment the graph, we added another type of edge called the tf-idf edge into our graph. The idea is to increase the number of edges and reduce sparsity. To add these edges, the following steps are performed sequentially:

1. Convert each document into a list of words using a tokenizer where each word is processed through a lemmatizer and excludes stopwords.
2. Since legal documents also contain some unnecessary information that is not useful in measuring the similarity, we need to remove this information. Intuitively, for any similar pair of documents, this unnecessary information will generally be specific to one document and will not be present in another document. Therefore to get the useful words in our corpus, we perform the following operation. For every pair of documents in which one document cites another, find the words that are common to both, called intersection words, then take the union of all intersection words for all citation pairs in our dataset. This final set of words are the catchwords and are found to be the most relevant words in our corpus.
3. Since our document is stored as a list of words, iterate over this list and exclude all words that are not present in the catchwords computed above.
4. Acts and articles present in a legal document are very important in capturing the similarity. To capture this information, we extended each document word list with the words that are mentioned in the definition of acts and articles which we collected in dataset preparation (section 4.1 above).
5. Now compute the tf-idf vector representation of each document.
6. Finally, compute cosine similarity (*sim*) for every possible pair of documents and if the *sim* is more than the threshold then this edge (tf-idf edge) is included in the graph with weight *sim* otherwise not. The value of threshold would affect the quality of the citation network, lower value of threshold would add more false (unrelated) edges while higher value would discard more good (related) edges. Through manual checks on a subset of document pairs, we used the value of threshold as 0.6.

Since any particular pair of documents in the graph can have both citation edge and tf-idf edge, in that case, we consider only the tf-idf edge. Finally, we get an undirected weighted graph of all the documents in our corpus. As new nodes are also included by introducing tf-idf edges. The final graph has 23,420 nodes and 355,426 edges.

4.2.3. Graph Embedding

Several graph embedding techniques have been proposed such as DeepWalk [7], LINE [8], BigGraph [9], Node2Vec [10] etc. In this paper, we used Node2Vec for feature learning. It takes as input an edge list with optional weights for edges and outputs a dense representation of the nodes in the network. It learns a feature representation that maximizes the likelihood of preserving network neighborhoods of nodes in a *d* dimensional feature space. We ran the Node2Vec algorithm on our graph obtained in the previous step. We chose the following parameter values: *dimensions*=300, *walk length*=16, *number of walks*= 300 and *p*, *q*= 1. After Node2Vec we have dense representations for all documents in our corpus.

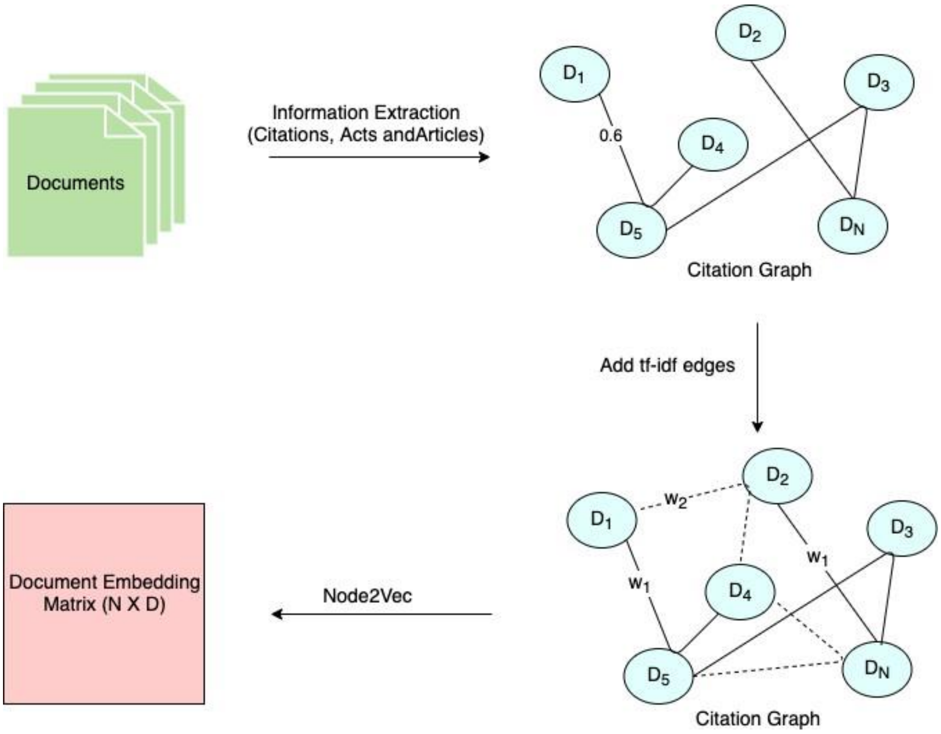


Figure 4.2. Complete system overview

5. Experiments and Results

As discussed above, our embeddings are derived from the citation graph, so the quality of learned embeddings depends on the quality of the citation graph. We conducted experiments to evaluate the quality of (i) Citation Graph (ii) Learned Embeddings using this citation graph.

Dataset: We used the dataset consisting of 47 pairs of documents from prior work by KUMAR ET AL. [1] and MANDAL ET AL. [5] . The original dataset were tagged by experts on a scale of 0 to 10 based on the similarity of the document pair.

Baseline: For baseline performance we consider the methods proposed in KUMAR ET AL. [1] and MANDAL ET AL. [5]. They studied different methods to learn document similarity. Specifically, [1] considered the co-citation and bibliographic coupling-based similarity methods to find similar judgements and [5] performed the experiments using (1) tf-idfrepresentation of the document (2) Word2Vec, where each document vector is computed by taking the weighted average of word vectors in the document with weights being the tf-idf of theword (3) Doc2Vec [6].

Results: To compare our approach with previous benchmarks, we used the same evaluation metric as in [5] i.e. the Pearson correlation coefficient between the similarity scores computed by different methods and the expert scores. The Pearson correlation coefficient of two variables is given by

$$\rho = \frac{cov(X, Y)}{\sigma_X . \sigma_Y}$$

where *cov(A, B)* is the covariance of variables *A* and *B*, and σ_A is the standard deviation of the variable *A*. The correlation coefficient has a value in the range [−1,1] and will indicate how similar are variables *A* and *B*.

To evaluate the Citation Graph, we computed the correlation coefficient between the expert's scores and similarity of documents (inverse of the distance between the nodes (representing the documents) in the graph). The distance between the nodes is the minimum path length between the nodes and documents that are more similar will have less distance in the graph.

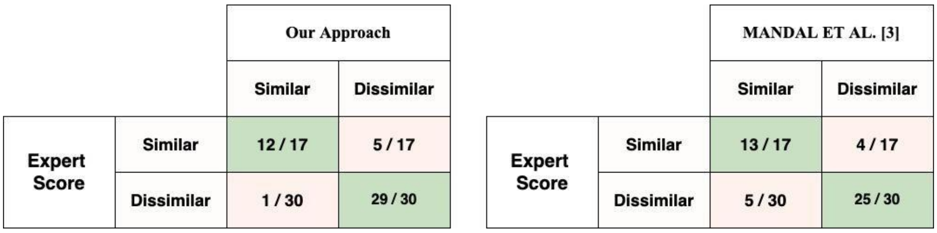
To evaluate the learned embeddings we computed the correlation coefficient between the expert's scores and similarity of documents (using cosine similarity of learned embeddings of the document-pair). A higher value of cosine similarity will signify document-pair is similar and dissimilar otherwise.

The correlation coefficients of the methods are given in Table 5.1. Our learned embeddings outperforms MANDAL ET AL. [5] and other document embedding techniques. Also, we obtained 0.64 correlation (ρ) for Citation Graph which is at par with other benchmarks. The Citation Graph can further be improved using better extraction techniques or manually extracted citation pairs and may further improve our learned embeddings.

Table 5.1. Correlation Coefficient comparisons using different document representations

	KUMAR ET AL. [1]	**Word2Vec**	**tf-idf**	**Our Approach (Citation Graph)**	**MANDALET AL. [5]**	**Our Approach (Learned Embeddings)**
ρ	0.33	0.60	0.62	**0.64**	0.69	**0.73**

We also performed a binary classification task by modifying the dataset using similar settings as done in [5]. The expert scores (in the range [0, 10]) are converted into two labels '0' and '1' i.e. scores in range [0, 5] are considered dissimilar pairs and given label '0' while scores in range [6, 10] are considered similar pairs and given label '1'. At the time of prediction, a document-pair is given label '1' if the cosine similarity between the embedding of the documents is greater than 0.5, otherwise given label '0'. The confusion matrix for our approach and [5] is shown in figure 5.1.



		Our Approach	
		Similar	**Dissimilar**
Expert Score	**Similar**	**12 / 17**	**5 / 17**
	Dissimilar	**1 / 30**	**29 / 30**

		MANDAL ET AL. [3]	
		Similar	**Dissimilar**
Expert Score	**Similar**	**13 / 17**	**4 / 17**
	Dissimilar	**5 / 30**	**25 / 30**

Figure 5.1. Confusion Matrix for evaluating the performance of our approach and [5]. Here, theproblem of identifying similar documents is modeled as a two-class classification problem, where document-pair is to be classified as either similar or not similar. Our Approach achieves better accuracies (87.23%) as compared to the baseline method [5] (80.85%)

6. CONCLUSION

In this paper, we have proposed a novel approach to algorithmically learn the representations of legal judgments using the citation graph. We explored the different types of information present in legal judgements. For example: facts of the case, citations to earlier cases, acts, articles, judicial reasoning, general observations. We started with the hypothesis that for common law systems the important information in the judgement is present in a) the citations of cases, articles, acts, etc. and b) the legally important words in the text. We represented this information in the form of a citation graph and learned the low dimensional representation of legal judgements on this graph. Our experiments and results show that the learned embeddings are capable of capturing the similarity between the legal judgements and outperform previous works on legal judgement similarity tasks. These learned representations of judgements can further be used in other downstream tasks like taxonomy construction, question-answering (document retrieval for user query), summarization, etc.

We believe that our results can be further improved by using better extraction methods for determining the cited judgements, acts, articles, and other important references or using the hand-curated dataset. This will help in constructing a dense citation graph which will result in learning better embeddings. Currently, we are not capturing amendments to existing laws that happen over time, so in future work, we can capture this information and may improve our results further.

REFERENCES

[1] Kumar, S., Reddy, P. K., Reddy, V. B., and Singh, A. (2011). Similarity analysis of legal judgments. In *Proceedings of the Fourth Annual ACM Bangalore Conference*, page 17. ACM

[2] Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Malti Suri. 2013. Finding Similar Legal Judgements under the Common Law System. 103–116

[3] Thenmozhi, D., Kannan, K., and Aravindan, C. (2017). A text similarity approach for precedence retrieval from legal documents. In *FIRE (Working Notes)*, pages 90–91

[4] Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013b). *Efficient estimation of word representations in vector space*. CoRR, abs/1301.3781

[5] Mandal, A., Chaki, R., Saha, S., Ghosh, K., Pal, A., and Ghosh, S. (2017). Measuring similarity among legal court case documents. In *Proceedings of the 10th Annual ACM India Compute Conference on ZZZ*, pages 1–9. ACM

[6] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196

[7] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena.2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM

[8] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, JunYan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077. International World WideWeb Conferences Steering Committee

[9] Adam Lerer, Ledell Wu, Jiajun Shen, TimotheeLacroix, Luca Wehrstedt, Abhijit Bose, and AlexPeysakhovich. 2019. PyTorch-BigGraph: A Large-scale Graph Embedding System. In *Proceedings of the 2nd SysML Conference, Palo Alto, CA, USA*

[10] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Con-ference on Knowledge Discovery and Data Mining*

OPTAGAN: ENTROPY-BASED FINETUNING ON TEXT VAE-GAN

Paolo Tirotta[1] and Stefano Lodi[2]

[1]Department of Statistics, University of Bologna, Italy
[2]Department of Computer Science, University of Bologna, Italy

ABSTRACT

Transfer learning through large pre-trained models has changed the landscape of current applications in natural language processing (NLP). Recently Optimus, a variational autoencoder (VAE) which combines two pre-trained models, BERT and GPT-2, has been released, and its combination with generative adversarial networks (GANs) has been shown to produce novel, yet very human-looking text. The Optimus and GANs combination avoids the troublesome application of GANs to the discrete domain of text, and prevents the exposure bias of standard maximum likelihood methods. We combine the training of GANs in the latent space, with the finetuning of the decoder of Optimus for single word generation. This approach lets us model both the high-level features of the sentences, and the low-level word-by-word generation. We finetune using reinforcement learning (RL) by exploiting the structure of GPT-2 and by adding entropy-based intrinsically motivated rewards to balance between quality and diversity. We benchmark the results of the VAE-GAN model, and show the improvements brought by our RL finetuning on three widely used datasets for text generation, with results that greatly surpass the current state-of-the-art for the quality of the generated texts.

KEYWORDS

Text Generation, Variational Autoencoders, Generative Adversarial Networks, Reinforcement Learning.

1. INTRODUCTION

Unsupervised text generation finds its use on a plethora of real-world application, ranging from machine translation [1], to summarization [2] and dialogue generation [3]. A general approach to modelling text sequences is to auto regressively generate the next token given the previous ones, and the most successful and widespread technique is to train a model using maximum likelihood estimation (MLE). This approach, however, is not without fault. At training time the model learns to generate a token given the ground truth, while at inference time it takes as input its own generated sequence of words. This dissimilarity leads to the so-called exposure bias [4], where the accumulation of errors during inference can produce poor outputs. Furthermore, the loss function of MLE is very strict. For each sequence, only the token accounted by the training sample is considered as correct [5], and the model learns precisely to mimic the given samples, often leading to quite dull and homogeneous outputs.

An alternative to MLE methods are generative adversarial networks (GANs) [6], where a generator learns to create outputs that can fool a discriminator into believing they are real. Thus, GANs do not have the strict loss function of MLE, and do not suffer from exposure bias, as they learn to sample during training. Nonetheless, the application of GANs to the text realm has been rather complicated. Due to the discreteness of text, the sampling of each token results in a non-

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 31-44, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112303

differentiable function, which does not allow to back propagate the loss of the discriminator. Countermeasures include the use of reinforcement learning (RL) [7][8][9][10], the use of the Gumbel-Softmax relaxation [11][12], or to avoid the discrete space altogether and work with continuous embeddings using autoencoders [13][14][15]. However, methods which utilize RL often rely on MLE pre-training, and usually do not improve over them [16]. Instead, for both the approaches using the Gumbel-Softmax distribution, and even more so for autoencoders, the discriminator considers a continuous representation of text, so it is not able to judge effectively the single word-by-word generation.

In the past few years, natural language processing (NLP) applications have found huge improvements with the introduction of the attention mechanism and the transformer architecture, with notable examples of BERT, GPT-2 and GPT-3 among others [17][18][19][20]. These kind of language models are large deep neural networks that are able to understand the dependencies between words thanks to attention and are trained over huge amounts of unannotated data. As such, pre-trained language models provide better language understanding over recurrent neural networks, can be very easily finetuned on a downstream task, including text generation, and reached state-of-the-art results in many areas. Recently Optimus, a text variational autoencoder (VAE), that is an autoencoder which maps sentences to a meaningful latent space, has been proposed [21]. It combines both BERT and GPT-2, as encoder and decoder respectively, and can be employed both as a generative model, and as a tool for language understanding tasks.

In this work, we aim to benchmark the results obtained from combining Optimus and GANs, similarly as indicated in the original paper. In doing so, we also investigate the GAN structure and compare the adaptive update strategy presented in [22] with the standard update strategy of GANs. Furthermore, we combine the training in the continuous space, with the finetuning of the decoder of Optimus in the discrete text space, in a similar fashion as done in ConCreteGAN [23]. However, differently from most approaches which use RL, we do not use REINFORCE, but add an additional value head to GPT-2, which outputs the intermediate rewards [24]. Moreover, we modify the reward function by considering the entropy of the model when generating tokens, and favour diversity in the output by adding an intrinsic reward.

Thus, our model OptAGAN[1] is able to model both the higher level sentence structure, and has more control over single word generation, in a way that favours both quality and diversity for the generated sentences. We measure such criteria using standard automatic metrics: BLEU for quality, Backwards-BLEU for diversity, and Fréchet distance, on which we also present further analysis. We consider the image caption dataset COCO, the Stanford Natural Language Inference (SNLI) dataset, and the EMNLP News 2017 dataset for unconditional text generation, and also provide results for the conditional review dataset YELP.

Results show that the base VAE-GAN model already improves over other GAN methods, especially with regards to quality. OptAGAN, further improves over these results, and manages to handle the quality-diversity trade-off very well. Moreover, we show a further experiment that helps understanding the strengths and weaknesses of our finetuning approach.

2. BACKGROUND

In this section we introduce the mathematical notation and briefly describe the main theoretical tools which are used in OptAGAN. We also present an overview of the other methods of text generation.

[1] Opt(imus) A(ugmented) GAN – Implementation can be found at https://github.com/Egojr/optagan

2.1. Variational Autoencoders

VAEs are generative models formed by two independent models, an encoder q_φ and a decoder p_θ. The encoder is tasked with mapping the input $\boldsymbol{x}$ to a latent space $\boldsymbol{z}$ that allows for interpolation. The decoder maps from $\boldsymbol{z}\rightarrow\tilde{\boldsymbol{x}}$, providing an approximation of the original input. Thanks to the introduction of a local variation from sampling the encoder output, it is possible to induce a smooth latent representation of the inputs, which differs from the rigid space of autoencoders.

Optimus Optimus combines the autoregressive nature of the GPT-2 text generation with the latent produced by the encoder, such that text generation is done as:

$$p_\theta(x|z) = \prod_{t=1}^{n} p_\theta(x_t|x_1, \dots, x_{t-1}, z) = \prod_{t=1}^{n} p_\theta(x_i|x_{<i}, z), \qquad (1)$$

where the probability of each token is estimated conditionally on the latent embedding and the previous tokens. The latent vector $\boldsymbol{z}$, which comes from the output of the BERT encoder, controls the high-level characteristics of the sentence, such as length, tense, style and topic, and allows for the guided generation of text.

2.2. Generative Adversarial Networks

GANs are also generative models formed by two models: a generator and a discriminator. Differently from VAEs, the generator G samples from a random variable to produce output that can fool the discriminator D into believing they are real, while the discriminator is constantly learning to distinguish between the real and generated data. The objective of the two models can formulated as:

$$\min_G \max_D \big(V(D,G)\big) = E_{x\sim p_D}\big[\log\big(D(x)\big)\big] + E_{\epsilon\sim p_\epsilon}[\log(1 - D(G(\epsilon)))], \qquad (2)$$

where p_D and p_ε are the distribution of the data and of the input noise of the generator, respectively. When combining GANs with autoregressive text generation, the operation of sampling the next tokens is non-differentiable, so the application of GANs relies on either the use of policy gradient algorithms, or the use of continuous approximations, such as the combination of GANs with autoencoders or VAEs.

2.3. Reinforcement Learning

Approaches that use policy gradient algorithms to allow the training of GANs consider the generator as the policy $\pi_{\boldsymbol{\theta}}$ to train, the sampling of the token as an action A from a state S, and the output of the discriminator as the reward R. As the discriminator only calculates the reward over finished sentences, the intermediate rewards are obtained through the REINFORCE algorithm and Monte-Carlo rollout. Optimization of the parameters is performed through gradient ascent:

$$\theta_{t+1} = \theta_t + \alpha\nabla_\theta J(\theta_t), \qquad (3)$$

$$\nabla_\theta J(\theta_t) \propto E_\pi[G_t \nabla_\theta \ln \pi\,(A_t|S_t, \theta)], \qquad (4)$$

where the gradient of the REINFORCE objective, $\nabla_{\boldsymbol{\theta}}\, J(\boldsymbol{\Theta}_t)$ is proportional to the discounted returns $G_t{=}\Sigma^t{}_{j=0}\gamma^j R_j$, so that higher return actions are favoured, and is inversely proportional to the probability of being selected, so that higher probability actions are not at an advantage compared to low probability ones. Equation 4 also provides a value that can be sampled at each time step and only depends on the policy π.

2.4. Related Work

Many works have dealt with the training of GANs in the discrete realm, starting with SeqGAN [7], LeakGAN [8] and RankGAN [9], where all of them share a similar structure, mostly differing in the form of the discriminator, and require MLE pre-training followed by adversarial training with REINFORCE. ScratchGAN [10] is the first model to show that MLE pre-training can be avoided by carefully combining existing techniques. Other works that train GANs using continuous relaxations include ARAE [13] and LATEXT-GAN [15], which use autoencoders to learn a continuous latent representation. Models based on the Gumbel-Softmax distribution are RelGAN [12] and GSGAN [11]. On the comparison of these methods and the evaluation metrics, [16][25] have shown the inadequacy of current GANs when compared to MLE and the need for metrics that can better measure the quality and diversity of the models.

On the topic of exploration of text GAN models and RL are ColdGANs [26], which delve deeper into the effects of temperature for the text generation. An approach similar to ours, which involves the use of large pre-trained models and RL is TextGAIL [27], where both GPT-2 as the generator, and RoBERTa as the discriminator are used for the task of text generation.

Regarding text VAEs, Optimus [21] is the first large pre-trained model of such kind, whereas previously researchers had tried developing VAEs using either recurrent neural networks [28] or semi-amortized inference [29].

3. OPTAGAN

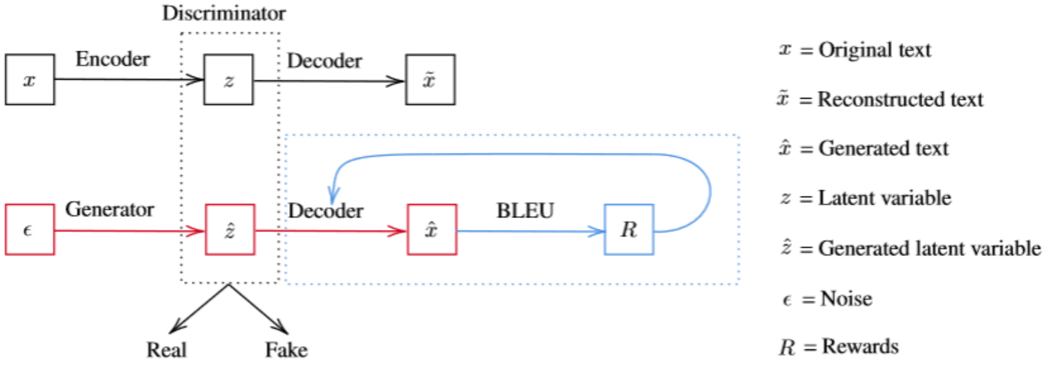


Figure 1. Structure of our proposed model OptAGAN. In red is the main process to generate new text, while in black we show the parts that are present in the training of the model. Finally, in blue is the last bit of the RL finetuning process.

The architecture that we present in this work is composed of three main processes, as can be seen from Figure 1. Each process is independent of the others, so each part is trained sequentially.

- In order to fully utilize the strengths of Optimus, we finetune both the encoder and the decoder on the target dataset. The end results are a more separated and distinct latent space for each sentence, and a decoder which better reconstructs the original sentences.
- Next, we train the GAN model composed of the generator and the discriminator. In the case of conditional generation, we also add a classifier, whose loss is then passed to the

generator. Both the generator and the discriminator only consider the continuous latent embeddings, so they are much lighter and faster to train compared to other text GANs.

- Finally, we finetune the decoder on discrete text using a value head, which estimates the reward of each single token in a sentence for the generated sequences. The estimated rewards are also augmented by considering the model entropy of each generated token. The gradient is then passed to the decoder through simple policy gradient.

The structure of both the generator and the discriminator is a simple feed-forward neural network. Current literature does not give clear answers about which loss function specification for GANs is best for continuous data such as text latent embeddings. Our experiments show that Wasserstein GANs with gradient penalty (WGAN-GP) perform the best over sliced Wasserstein distances, which produce very homogeneous outputs. Thus, the loss function to optimize for the generator and discriminator is:

$$\min_G \max_D E_{x\sim p_D}\,[D(x)] - E_{\epsilon\sim p(\epsilon)}\big[D\big(G(\epsilon)\big)\big] + \lambda E_{\hat{x}\sim p(\hat{x})}[(|\nabla_{\hat{x}} D(\hat{x})|_2 - 1)^2], \qquad (5)$$

3.1. Update Strategy

When training GANs, the common update strategy is to have *k*, usually in the range [5][10], update steps for the discriminator for each step of the generator, as it has been shown to give stable training for GANs. We indeed also use this update method, but we also experiment with an adaptive update strategy proposed in [22], where the choice to update the discriminator or the generator is given by a comparison of the loss change ratio of the two networks.

$$r_G = \frac{\left|\left(L_G^c - L_G^p\right)\right|}{L_G^p + c}, \qquad (6) r_D = \frac{\left|\left(L_D^c - L_D^p\right)\right|}{L_D^p + c}, \qquad (7)$$

where the relative change between the current loss L^c and previous loss L^p for both networks is used in determining which one gets updated. We also add an arbitrarily small constant *c* in case the losses are too close to 0.

A weight λ can be also introduced, so that if $r_D > \lambda r_G$ the discriminator is updated, and vice versa. Contrarily to the original paper, which suggests a value λ□ □≥ 1, we notice that at the beginning of training there is a stark imbalance in the number of updates between the networks resulting in slower convergence. To better balance the training, we end up using a value of $\lambda < 1$, which converges to 1 with each passing epoch of training.

3.2. Value Head

Due to the high computational costs of implementing a text discriminator with a vocabulary of size equal to GPT-2, we rely on an external value head, whose scalar output for each token corresponds to the intermediate reward. Let the hidden states of the decoder ρ and the value head v, the rewards calculated from an external metric R^{ex} and each state S^t are:

$$R_t = \nu(S_t|\rho), \qquad with\; \nu(S_t|\rho)\; minimizing \left|\sum_{t=0}^{T}\big(\nu(S_t|\rho)\big) - R^{ex}\right|, \qquad (8)$$

The external head takes as input the frozen hidden states of GPT-2. Freezing the hidden parameters is necessary because we are not modelling the parameters of the VAE model during RL. The loss is then calculated according to a MAE objective and passed back to the value head.

This process estimates how much each token contributes to the reward, and comparatively to a text discriminator is faster to train and showed better results.

3.3. Entropy-Based Rewards

Our external rewards are calculated based on the quality metric BLEU, which we briefly describe in Section 4.1. Under many considered reward specifications, which included the addition of diversity metrics, maximum entropy RL, changes of temperature, or a combination of these, the increase in quality is counterbalanced with a drop in the diversity of the generated sentences. One approach that managed to balance the quality-diversity trade-off was the addition of an intrinsically motivated penalty based on the confidence of the model when generating the token, calculated by the entropy. If again we consider the hidden states ρ, the last layer calculating the logits as our policy π with parameters $\boldsymbol{\Theta}$ and the entropy as $H(\pi(\bullet|\rho,S_t))$, we calculate the intrinsic rewards and the performance objective as:

$$R_t^{in} = \log\left(clamp\left(H\left(\pi(\cdot\,|\rho, S_t)\right), 0.2{,}1\right)\right), \qquad (9)$$

$$\nabla_\theta J(\theta_t) = E_\pi\left[\left(\sum_{k=0}^{t}\left(\gamma^k R_k\right) + R_t^{in}\right)\nabla_\theta \ln \pi\,(A_t|\rho, S_t, \theta)\right], \qquad (10)$$

This specification favours high-reward actions with high entropy, while low-entropy actions have to have a high enough reward to be able to keep their high probability, resulting in a more diverse generation. As a rule of thumb, we found out that penalties should be lower than the maximum overall reward.

4. EXPERIMENTAL SETTINGS

In this section, we introduce the automatic metrics and the datasets used for evaluation. For comparison we consider a MLE model, SeqGAN, RankGAN, as implemented by the benchmarking platform Texygen[30], and ScratchGAN. We also provide further details on our RL finetuning and present issues with the current evaluation metrics.

4.1. Evaluation Metrics

BLEU is a metric that measures the overlapping n-grams between a hypothesis text and all the reference texts. The final score is calculated as the average of the scores over all hypothesis sentences. Studies have shown [16][25] that BLEU can only detect small syntax problems, resulting in poor correlation with human evaluations, however it still remains the standard when evaluating the quality of generated texts. To measure diversity we utilize Backwards-BLEU (BBLEU) where the generated texts are the reference and the test set becomes the hypothesis, giving a measure of how much the test set is represented.

Additionally, we consider the Fréchet Distance with the InferSent embedding model (FID). It has been shown that FID responds better than BLEU at identifying mode collapse and changes in words usage. However, we show that it can be biased due to its distributional assumptions, mainly for differences in sentence length distribution. Nonetheless, it can be useful in identifying very homogeneous outputs, especially in conjunction with BLEU and BBLEU scores.

4.2. Datasets

We consider three of most widely used datasets for unconditional text generation: image COCO [30], Stanford Natural Language Inference (SNLI) [31] and the EMNLP News 2017 dataset[2].Moreover, we consider the YELP review dataset for conditional text generation [33].

Table 1. Average length of the train set and number of sentences for each of the datasets used for evaluation.

	COCO	SNLI	EMNLP	YELP
Conditional	X	X	X	✓
Average sentence length	11.3	9.7	28.8	96.4
Size of train set	10k	100k	270k	100k
Size of dev set	10k	10k	10k	10k
Size of test set	10k	10k	10k	10k

Each of the datasets presents different challenges when training: COCO and SNLI are a small and medium-sized dataset with short sentences, respectively. EMNLP is a large dataset with longer sentences. Lastly, the YELP dataset is a conditional, medium-sized dataset with very long sentences. The preprocessing on the datasets is minimal and only on the YELP dataset.

4.3. Experimental Results

Tables 2, 3 and 4 show the results for the quality and diversity of the models. The changes between standard and adaptive updates mostly favour the adaptive one, with larger gains in diversity, with the exception of the SNLI dataset. Additional considerations about the two updates can be found in Appendix B. Therefore, we only apply the RL finetuning on the adaptively trained model to obtain the OptAGAN results. Our approach shows improvements under all metrics, albeit small for the COCO and SNLI datasets. In comparison with the other GAN models, OptAGAN boasts the highest quality, and average, or higher diversity. In comparison with the MLE model, the quality-diversity trade-off favours our model for quality, and the MLE approach for diversity. Notably, for the EMNLP dataset, the curiosity-driven finetuning allows OptAGAN to surpass all models for both BLEU and BBLEU.

We believe that the difference in the magnitude of change between the EMNLP task and the COCO and SNLI ones is due the starting quality of the model. In fact, as the RL finetuning slightly prioritizes quality, so increases in BLEU score, over diversity, the actual changes on the word-by-word generation are very few for those two datasets.

Compared to the other methods, ours is also better able to reproduce longer sequences of words, as the growing differences between 2,3 and 4 n-grams metrics show. Regarding the FID scores, they mostly show the same behaviour as BBLEU. However, we show in section 4.5 that the FID is biased for the sentence length distribution, that, in contrast with other methods such as ScratchGAN, we do not model.

[2] http://www.statmt.org/wmt17/

Table 2. EMNLP results of the automatic metrics for OptAGAN, the two base VAE-GAN models with standard and adaptive updates and the other models implemented for comparison.

Metrics	MLE	SeqGAN	RankGAN	ScratchGAN	Standard	Adaptive	OptAGAN
BLUE-2	0.829	0.796	0.764	0.835	0.825	0.816	**0.860**
BLUE-3	0.548	0.471	0.399	0.556	0.554	0.544	**0.605**
BLEU-4	0.304	0.228	0.159	0.313	0.285	0.284	**0.356**
BBLEU-2	0.840	0.762	0.728	0.824	0.765	0.805	**0.841**
BBLEU-3	0.563	0.563	0.383	0.545	0.488	0.526	**0.586**
BBLEU-4	0.317	0.317	0.221	0.303	0.261	0.278	**0.350**
FID	0.926	1.934	3.509	**0.466**	1.153	0.784	0.674

Table 3. SNLI results of the automatic metrics for OptAGAN, the two base VAE-GAN models with standard and adaptive updates and the other models implemented for comparison.

Metrics	MLE	SeqGAN	RankGAN	ScratchGAN	Standard	Adaptive	OptAGAN
BLUE-2	0.841	0.838	0.784	0.795	0.867	0.888	**0.889**
BLUE-3	0.635	0.599	0.514	0.564	0.693	0726	**0.727**
BLEU-4	0.428	0.380	0.309	0.363	0.484	0.524	**0.525**
BBLEU-2	**0.843**	0.768	0.771	0.800	0.786	0.764	0.764
BBLEU-3	**0.639**	0.546	0.523	0.564	0.570	0.547	0.548
BBLEU-4	**0.433**	0.347	0.347	0.362	0.374	0.361	0.362
FID	**0.376**	0.919	1.486	0.539	1.424	1.764	1.765

Table 4. COCO results of the automatic metrics for OptAGAN, the two base VAE-GAN models with standard and adaptive updates and the other models implemented for comparison.

Metrics	MLE	SeqGAN	RankGAN	ScratchGAN	Standard	Adaptive	OptAGAN
BLUE-2	0.854	0.866	0.834	0.862	0.917	0.919	**0.920**
BLUE-3	0.664	0.667	0.641	0.678	0.792	0.792	**0.794**
BLEU-4	0.459	0.452	0.423	0.478	**0.627**	0.617	0.620
BBLEU-2	**0.844**	0.813	0.775	0.791	0.755	0.775	0.778
BBLEU-3	**0.656**	0.616	0.553	0.573	0.550	0.579	0.584
BBLEU-4	**0.455**	0.415	0.344	0.377	0.368	0.396	0.399
FID	**0.588**	0.756	2.347	1.001	2.297	1.908	1.903

From the computational cost point of view, the full training of our model took at most 24 hours using a single Tesla V100 GPU. Additional details about the training can be found in Appendix A.

4.4. Conditional Generation

For the conditional generation task, we follow the same procedure as the unconditional one, with the only exception of the addition of a classifier network to better model the GAN generation depending on the label. The results of table 5 are very similar to the ones of the COCO and SNLI, where the entropy regularized finetuning performs slightly better than the base adaptive VAE-GAN model, with small gains in diversity and quality. We present some examples of the generated sentences in Appendix C.

Table 5. Yelp results of the automatic metrics models and the entropy regularized OptAGAN.

Metrics	Standard	Adaptive	OptAGAN
BLUE-2	0.880	0.886	**0.887**
BLUE-3	0.675	0.683	**0.685**
BLEU-4	0.448	0.456	**0.458**
BBLEU-2	**0.860**	0.854	0.854
BBLEU-3	**0.666**	0.660	0.661
BBLEU-4	**0.453**	0.449	0.451
FID	2.598	**2.539**	2.562

4.5. Analysis on FID Space

We discuss problems with the FID metric by showing an heatmap of the two-dimensional principal component analysis (PCA) representation and the length distribution of the sentences for the SNLI dataset. Previous works [10] already investigated the dependency of the FID scores on the length distribution, which can overshadow other problems with the generated samples.

We further reinforce those analysis, as a prime example of this issue can be seen in Figure 2, where the FID scores get progressively higher the more the length distribution of the generated sentences is close to the one of the test data, while ignoring the fact that the actual distribution may not match the test one.

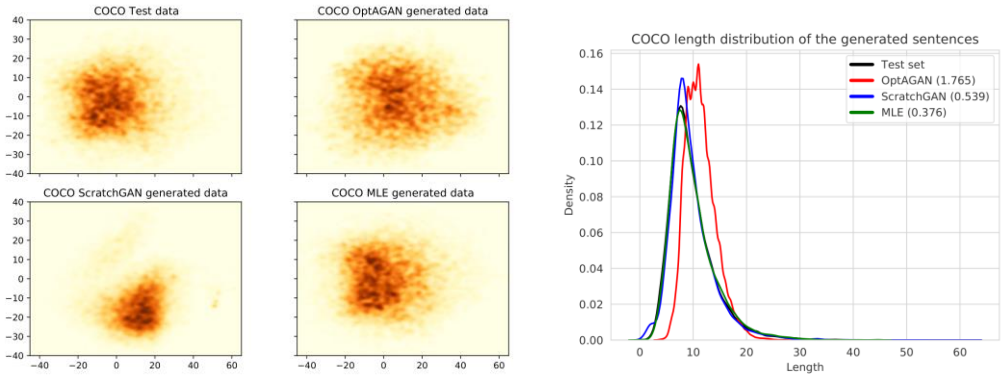


Figure 2. On the left, the PCA representation of the FID space of the SNLI test set (top left), OptAGAN (top right), ScratchGAN (bottom left) and MLE (bottom right) dataset. On the right, the length distribution of the four datasets and the FID scores for the generated data.

In fact, the score for the ScratchGAN model is much lower than the one for OptAGAN, although the distribution of the sentences of ScratchGAN in the space misses most of the distribution of the test sentences, as can be seen from the PCA representation. Although only 10-15% of the overall variability, over the InferSent embedding dimensions, is explained by PCA, there is a huge mismatch that is not addressed by the use of the Fréchet distance, which favours homogeneous length distributions over correct representation of the space.

4.6. Experiment on RL Finetuning

In order to fully gauge the strengths and weaknesses of our entropy penalty approach we set up an experiment where we train a GAN model using the pre-trained Optimus encoder and decoder that are not finetuned on the dataset, so we can finetune a lower quality model. We compare this model with two curiosity-driven regularized model for 1000 and 5000 epochs, respectively. We also use a higher learning rate as we are not interested in preserving the structure of the original sentences and present the results in Table 6.

The starting model achieves much worse results than the fully optimized OptAGAN, especially with regards to the BLEU score. After 1000 epochs of RL finetuning, the model improves over both quality and diversity. However, when finetuning for longer, the model cannot balance anymore between exploration and exploitation.

Table 6. Yelp results of the automatic metrics models and the entropy regularized OptAGAN.

Metrics	Base VAE-GAN	RL 1000 Epochs	RL 5000 Epochs
BLUE-2	0.641	0.707	**0.829**
BLUE-3	0.370	0.444	**0.624**
BLEU-4	0.198	0.243	**0.402**
BBLEU-2	**0.752**	**0.752**	0.717
BBLEU-3	0.478	**0.497**	0.479
BBLEU-4	0.266	**0.286**	0.283
FID	2.441	**2.369**	2.892

We believe this behaviour is because of the bias that the model has in finding high reward tokens. The RL finetuning does not evaluate all the tokens in the vocabulary, so there might be multiple good scoring tokens which are never considered during the finetuning. This means that our approach is limited by the quality of the original model.

A countermeasure that might prevent this kind of behaviour from happening could be to increase the penalty for models with higher average entropy, and tuning its value depending on the use case, to further encourage heterogeneous generation.

5. Conclusions and Discussion

In this work, we benchmark the combination of Optimus and GANs for a text VAE-GAN model, with results that already surpass current methods for the quality of generated texts. We further improve this baseline using entropy-based curiosity-driven rewards to improve both the quality and the diversity of the model. This novel approach could benefit many models utilizing RL for text generation, and supplementary research could be done into exploring advantage policy gradient, or proximal policy optimization with intrinsic rewards. This specification of the reward also allows researchers to prioritize quality, diversity, or to balance between both.

Due to our limited computational resources, we utilize smaller batch sizes than we would otherwise have preferred, as larger batch sizes could help in reducing the high variance gradients of these approaches. Moreover, further research on the automatic metrics could be beneficial not only for evaluation, but also for better reward signals to improve the speed and quality of the finetuning.

REFERENCES

[1] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, NishantPatil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. *Google's neural machine translation system: Bridging the gap between human andmachine translation*.

[2] Mehdi Allahyari, Seyed Amin Pouriyeh, Mehdi Assefi, SaeidSafaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys J. Kochut. *Text summarization techniques: A brief survey*.

[3] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. *Deep reinforcement learning for dialogue generation*.

[4] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. *Scheduled sampling for sequence prediction with recurrent neural networks*.

[5] Ofir Press, Amir Bar, Ben Bogin, Jonathan Berant, and Lior Wolf. *Language generation with recurrent generative adversarial networks without pre-training*.

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and Yoshua Bengio. *Generative adversarial networks*, 2014.

[7] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. *Seqgan: Sequence generative adversarial nets with policy gradient.*

[8] Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. *Adversarial ranking for language generation.*

[9] Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. *Long text generation via adversarial training with leaked information*, 2017.

[10] Cyprien de Masson d'Autume, Mihaela Rosca, Jack Rae, and Shakir Mohamed. *Training language gans from scratch,* 2020.

[11] Matt J. Kusner and José Miguel Hernández-Lobato. *Gans for sequences of discrete elements with the gumbel-softmax distribution*, 2016.

[12] WeiliNie, Nina Narodytska, and Ankit Patel. *RelGAN: Relational generative adversarial networks for text generation.* InInternational Conference on Learning Representations, 2019.

[13] Junbo Jake Zhao, Yoon Kim, Kelly Zhang, Alexander M. Rush, and Yann LeCun. *Adversarially regularized autoencoders for generating discrete structures*.

[14] David Donahue and Anna Rumshisky. *Adversarial text generation without reinforcement learning*.

[15] Md. Akmal Haidar, Mehdi Rezagholizadeh, Alan Do-Omri, and Ahmad Rashid. *Latent code and text-based generative adversarial networks for soft-text generation*.

[16] Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. *Language gans falling short*, 2020.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. *Attention is all you need.*CoRR, abs/1706.03762, 2017.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. *BERT: pre-training of deep bidirectional transformers for language understanding.*

[19] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. *Language models are unsupervised multitask learners*. 2019.

[20] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. *Language models are few-shot learners*, 2020.

[21] Chunyuan Li, Xiang Gao, Yuan Li, Baolin Peng, Xiujun Li, Yizhe Zhang, and Jianfeng Gao. *Optimus: Organizing sentences via pre-trained modeling of a latent space*, 2020.

[22] Xu Ouyang and GadyAgam. *Accelerated wgan update strategy with loss change rate balancing,* 2020.

[23] Yanghoon Kim, Seungpil Won, Seunghyun Yoon, and Kyomin Jung. *Collaborative training of gans in continuous and discrete spaces for text generation*, 2020.

[24] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. *Fine-tuning language models from human preferences.*

[25] Stanislau Semeniuta, Aliaksei Severyn, and Sylvain Gelly. *On accurate evaluation of gans for language generation*,2019.

[26] Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. Coldgans: Taming language gans with cautious sampling strategies, 2020.

[27] Qingyang Wu, Lei Li, and Zhou Yu. *Textgail: Generative adversarial imitation learning for text generation*, 2021.

[28] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, RafalJózefowicz, and SamyBengio. *Generating sentences from a continuous space*.

[29] Yoon Kim, Sam Wiseman, Andrew C. Miller, David Sontag, and Alexander M. Rush. *Semi-amortized variational autoencoders*, 2018.

[30] Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. *Texygen: A benchmarking platform for text generation models*.

[31] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. *Microsoft COCO captions: Data collection and evaluation server*.

[32] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. *A large annotated corpus for learning natural language inference*. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[33] Nabiha Asghar. *Yelp dataset challenge: Review rating prediction.*

APPENDIX

A. TRAINING DETAILS

For the finetuning of Optimus we follow the original work and train for one epoch with the hyperparameters that give the best reconstruction quality, namely:

- Pre-trained model epoch = 508523
- Training epochs[3] = 1
- Learning rate = 5 • 10^{-5}
- Batch size = 5
- Latent size = 768
- $\beta = 0$
- Annealing ratio = 0.5
- Ratio increase = 0.25

We follow a similar approach for the training of the GAN part of the model, where we use standard hyperparameters for the training of WGAN-GP. The best performing epoch of the GAN according to the sum of the BLEU and BBLEU partial scores with 500 texts for both reference and hypothesis is saved.

- Training epochs = 50
- Learning rate = 10^{-4}
- Batch size = 256
- Latent size = 768
- Maximum sequence length = 100
- Number of blocks of generator and discriminator = 10
- Gradient penalty $\lambda = 10$

[3] Due to the size of the COCO dataset, it is the only one where we finetune for 5 epochs

Finally, these are the details for the entropy regularized finetuning. Empirical results showed next to no difference for the BLEU n-gram choice, so we choose 1-gram due to slightly faster computational times, and it also translates into a clearer understanding of the intermediate values. Moreover, we use a small learning rate, in order to keep the same structure as the original sentences.

- BLEU reward n-grams = 1
- Finetuning epochs = 1000
- Learning rate = 10^{-6}
- Batch size = 32
- Epochs value head pre-training = 200
- Learning rate value head pre-training = 10^{-4}

B. Adaptive Strategy Details

Figure 3 shows the validation BLEU and BBLEU for the COCO and YELP datasets. We show the results over these two datasets due to the stark difference between them.

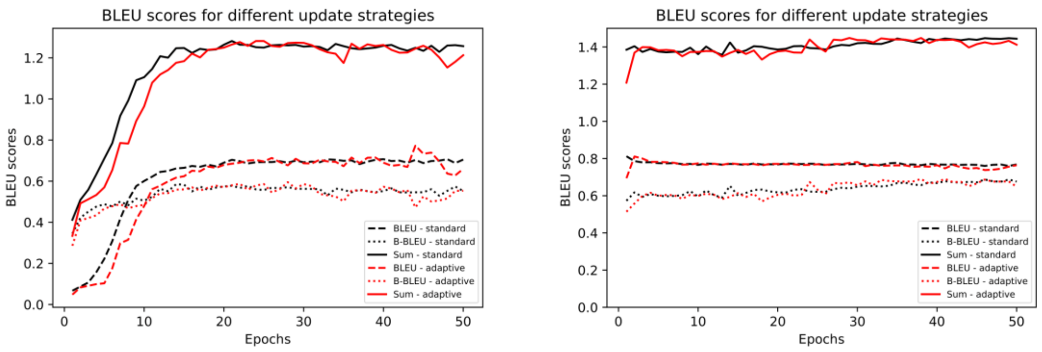


Figure 3. Differences in BLEU scores duringtraining for the standard and adaptive updatein GANs, evaluated on the COCO dev set on the left and on the YELP dev set on the right.

It is evident that the adaptive strategy is slightly slower to converge. Since the YELP dataset is 10 times larger than COCO, in both cases after about 200,000 samples the two models reach the same quality. The remainder of the training appears to be very stable for the standard update, with little to no changes in both the scores. Meanwhile, the adaptive updates look slightly more volatile, with changes that impact the two scores both positively and negatively, usually following the quality-diversity trade-off principle. However, it also means that the adaptive updates is more likely to find higher scores than the standard one, as happens for all the datasets analysed in this work.

C. Generated Samples

Table 7. Examples of the generated unconditional sentences from OptAGAN trained on COCO, SNLI and EMNLP dataset.

Dataset	Sentences
COCO	a man posing with a bike inside of a forest . a man sitting on a swinging chair pulled by some purple ducks . a woman holding a bird over some flowers on the beach .

SNLI	the man is sweating because they are blue . the man is getting more thank with the dog . a man in white clothes stands next to a marketplace where he can store plastic .
EMNLP	Republican presidential nominee , Donald Trump , has said that 20 to 30 years might be the way to try and narrow it out . Whether or not agenda minutes can deliver , it would , therefore , encourage the majority of Scottish MPs to think about that . Earlier in the day , Bell travelled to Sydney ' s Supreme Court and was effectively blocking a vote of no - one that would produce the album .

Table 8. Examples of the generated sentences from OptAGAN trained on the YELP dataset, conditioned on the label.

Stars	Generated review
1	i was disappointed with this company . for the 2 visits to this location on the _UNK highway i paid a ridiculous amount to visit . got stuck there , and ignored and messed up on my money . . the wait staff must know the difference between sink of vinegar and fresh vinegar !haha . i have seen orange juice _UNK when they used to serve it but now the juice i am getting from juice bar in sanfrancisco had no consistency at all .i asked the waitress if they could rotate me out of soda in the microwave in exchange for a new glass ... hmmm , if that's your way of saying bad customer service you are talking a rip off . wait staff can do anything .
2	i have been to some great buffets , but this was mediocre at best . my husband ordered a turkey sandwich and it was just like anyone else's sandwich . for the service , there wasn't much seating . the food and the _UNK were stale , awful bread . something to do if you go to vegas for dinner and want to have some classic awesomeness ... maybe try a dim sum instead .
3	this place is pretty good .i like the burgers , the selection is pretty good and they have a ginormous amount of steak . being a vegetarian ,i took one bite of everything i ordered and went back again . for the friday afternoon rush - they brought me the french fries instead of the turkey , a mousse and mgr .
4	loved this place .i had the corned beef burrito , which was very good , though a bit greasy and lacking . they have a good selection of veggie options and happy hour specials and are very attentive to your meal . it's clean , spacious , and the ambiance is great . i have definitely come here when my friends visit vegas to see if they have any other option . the best part about going here is the sitting area outside where you can hang out while eating all you could eat .
5	love this place .i have visited all of the great restaurants that offer all sorts of flavors , and to top it all off , all the facilities are extremely clean . _UNK is the guy . he came for a quick check up , took my dad to the bar and came out free of charge !! seriously !

PARA-SOCIAL INTERACTION ANALYSIS OF VALENTINO VIRTUAL SPOKESPERSON BASED ON WEIBO DATA AND SEARCH POPULARITY

Dandan Yu[1], Wuying Liu[1, 2]

[1]Shandong Key Laboratory of Language Resources Development and Application, Ludong University, 264025 Yantai, Shandong, China
[2]Laboratory of Language Engineering and Computing, Guangdong University of Foreign Studies, 510420 Guangzhou, Guangdong, China

ABSTRACT

Brand image is one of the most important factors influencing the competitiveness of the commodity market. In order to reduce the negative impact of potential scandals of celebrity spokespersons, Valentino began to use a virtual spokesperson named ***noonoouri*** *for promotion. In this context, we first describe the application model of Valentino's virtual spokesperson. Then, based on the Valentino Weibo data we collected and Google search popularity, we analyzed the impact of virtual spokespersons on the para-social interaction between brands and consumers. Finally, according to the analysis conclusion of the impact of Valentino's virtual spokesperson in the para-social interaction, a corresponding marketing-competitive brand image promotion proposal is put forward.*

KEYWORDS

Para-social Interaction, Valentino, Virtual Spokesperson, Weibo Data, Search Popularity.

1. INTRODUCTION

Last decade, we have witnessed the rapid development of China economy and the popularity of China mobile networking. The mobile social era has arrived. Digitization and rejuvenation are the inevitable trends in the development of the luxury industry. The mainly users of major social media are young people, and most of them tend to personalize consumption such as co-branded merchandise, limited merchandise and customized merchandise. Italian high-end luxury brand Valentino uses technology to lead product promotion. What Valentino need to do is to let young consumers to recognize and shape a young corporate image through brand marketing on social networking platform. Valentino use a "para-social" model and virtual spokespersons for digital marketing on traffic platforms such as Weibo, and use digital marketing to narrow the distance with consumers and acquire new users.

In the experiment, we analyze the trends of search popularity of virtual spokespersons in global search engines and the distribution of Weibo users' age in recent years. The experiment analysis results based on 4063 interactive data published on Valentino's official Weibo discusses whether virtual spokespersons will affect the formation of para-social interaction relationship between Valentino and their consumers. Finally, it discusses the problems existing in the process of establishing para-social interaction relationship between Valentino's virtual spokespersons and its consumers, and puts forward corresponding development suggestions.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 45-51, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112304

2. RELATED WORK

This experiment has consulted a large number of relevant literatures. According to the viewpoints of Daniel Langer and Oliver Haier on luxury brands functions, the prices of luxury merchandise contain high added-value and consumer perceived-value. Consumer perceived-value enables luxury to have the functions of enhancing the owner's self-esteem, self-awareness and attractiveness, liquidity of funds, social status and the ultimate sense of experience [1]. On the basis of such characteristics and functions, luxury merchandise added-value of is also reflected in the social status and social image which can be used as an emotional social symbol to distinguish people, so as to satisfy the psychological demands of consumers.

Virtual spokesperson refers to a virtual character which is based on the characteristics of luxury brand and comprehensive analysis of the market environment, competitors and consumer psychology. Virtual Spokesperson can be used as a carrier of brand image [2]. Like real spokesperson, virtual spokesperson not only does benefit luxury brand image brand image, merchandise and brand awareness, but also establish the reputation and loyalty of brand, stimulate brand associations, and promote brand development and promotion [3].

In 1956, psychologists Horton and Wall, based on the survey of the interaction between the audience and the host in the American radio show "Lonely Girl", put forward: Para-social interaction refers to the phenomenon in which the audience reacts to media characters as real characters and forms a para-social relationship with them. In the process of para-social activities, a person's self-identity depends on whether he can successfully play a social role and interact with others." In para-social interaction, individuals will communicate and establish relationship bonds with media characters with purpose, which can bring psychological comfort to individuals to a certain degree [4].

The para-social interaction behaviors of luxury goods in social media are increasing. Domestic academia's have gradually shifted their researches focus from exploring the use of new media in luxury marketing [5] and "How Does Luxury Advertising Make Use of Digital Media to Transform-Take Luxury Micro-Films as An Example" [6] to "The Influence of Social Interaction and Para-social Interaction in Virtual Brand Community on Brand Relationship Quality" [7] also gradually increasing. This experiment will build on the existing researches and based on a total of 4063 pieces of interaction data published on Valentino's official Weibo, including fans interaction, merchandise promotion, topic forwarding lottery and endorsements by real celebrity or virtual spokespersons, to explore how the brand's employment of virtual spokespersons affects the formation of para-social interactions between itself and its consumers.

3. PARA-SOCIAL INTERACTION

We collected and described the virtual spokesperson's application model in Valentino's advertisements based on the relevant information reviewed. We analyze the image characteristics of virtual spokespersons and explore the formation process of the para-social interaction between virtual spokespersons and consumers.

3.1. Application Mode of Virtual Spokesperson in Valentino advertising

In order not to affect the original advertising of Valentino that relies on celebrity spokespersons to promote merchandise and to retain consumers who tend to trust traditional advertising. Most of Valentino's endorsement advertisements use a combination of virtual spokespersons and celebrity spokespersons [8].

When the fame, social influence and fan base of the virtual spokesperson reach a certain level, brands will only adopt virtual character as the brand spokesperson. With the popularity of virtual spokespersons such as Lil Miquela and noonoouri in the fashion field, various virtual characters have set up the "Vmodel" development model. Vmodel poster's presentation, dialogue text, contextual, collaboration content and life image are completely team-operated to reduce negative images.

Valentino uses virtual spokespersons to make risks more controllable. Virtual spokespersons can fully meet the needs of Valentino and ensure the safety of promotion. Considering from the ideological level of young people, the image of virtual spokesperson represents the idealized lifestyle of the younger generation. The idealization and controllability of virtual spokesperson make them easier to be chosen by Valentino. After the virtual spokesperson lays the foundation for consumer acceptance, market has also made targeted changes. According to the advertisements published by Valentino's official Weibo, it can be seen that Valentino's promotion of virtual spokesperson advertisements has gradually shifted from appearing at the same time with celebrities known to young people in the early days to shooting advertisements and participating in various marketing activities alone.

3.2. Formation of Para-social Interaction between Virtual Spokespersons and Consumers

Consumers' emotional projection and personal identity come from the rich character images of virtual spokespersons. In the process of para-social interaction with virtual spokespersons, consumers acquired information is transmitted as symbols. Designers can use virtual status symbol and text symbols to shape the character and construct dialogue situations for virtual spokespersons, so that it can communicate information on social media in its own language.

All of characteristics make virtual spokespersons act as media characters to a certain extent. When virtual spokespersons cooperate with brands, brand consumers act as media audience when they receiving advertisements that contains virtual spokespersons. The important prerequisite for the formation of para-social interaction is the charm of media characters and perceived similarity of media audience. The deeper the interaction between consumers and virtual spokespersons, the more they can seek spiritual sustenance and emotional needs from virtual spokespersons. The para-social interaction between consumers and virtual spokespersons depends on the imagination of consumers rather than the actual contact behavior between the two. Only when the two parties have established a symbolic interpretation method that can share meanings can they truly establish a para-social interaction relationship between the two parties. In turn, luxury consumers will have a sense of identity with their personal social status and identity, as well as a sense of identity with the brand when interacting with virtual spokespersons.

4. EXPERIMENT

We obtained experiment's data from search engines and data mining. In the experiment, we store the search popularity data of keywords obtained from search engines in Excel format, and draw charts based on these data. We use Python3 code to obtain Weibo data. First, we obtain the APP_KEY and AP_SECRET of the newly registered application on the Weibo open platform and set the authorization return page and authorization cancellation page. Next, we create a MySQL database named Weibo to store the obtained ID and Weibo comment data. Then, we use the sinaweibopy3 module to obtain authorization from the Weibo interface and request data. Finally, compare the obtained data format with the data format in the interface document, and store the obtained Weibo data in the MySQL database after confirming that it is correct.

4.1. Result and Discussion about Search Popularity

As shown in Figure 1, according to the search engine data from 2004 to the present, the statistics of the global virtual influencer keyword search popularity show that since the virtual influencer Lil Miquela was created in 2016, the attention of online audience to the virtual has increased year by year. Taking 2018 as the node, virtual spokespersons such as Shudu Gram and noonoouri have been created and entered the public sight during this year. Since 2018, the attention of online audience to virtual spokespersons has increased exponentially and there is still a growth trend in the future.

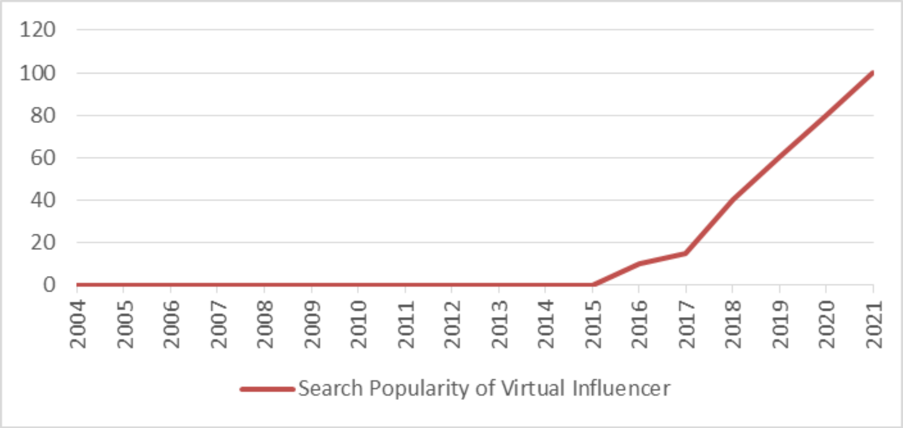


Figure 1. The search popularity statistics of global Virtual Influencer keywords since 2004.

Since the image of digital virtual spokesperson can be more conveniently transmitted to consumers through various social media. Digital virtual spokesperson can endorse more than one brand. The image of the virtual spokesperson represents the image and interests of itself, the creative team and various cooperative brands, which also makes its image more independent. At present, digital virtual spokespersons are mainly virtual spokespersonss such as Lil Miquela and noonoouri.

As shown in Figure 2, according to ***iResearch***'s survey data on PC and mobile Weibo users, the majority of them are young people. Among them, the users groups of 25 to 30 years old on the PC and APP respectively accounted for 21.6% and 29.1%, the users groups under the age of 24 includes teenagers on the PC and APP respectively accounted for 37.7% and 37.5%.

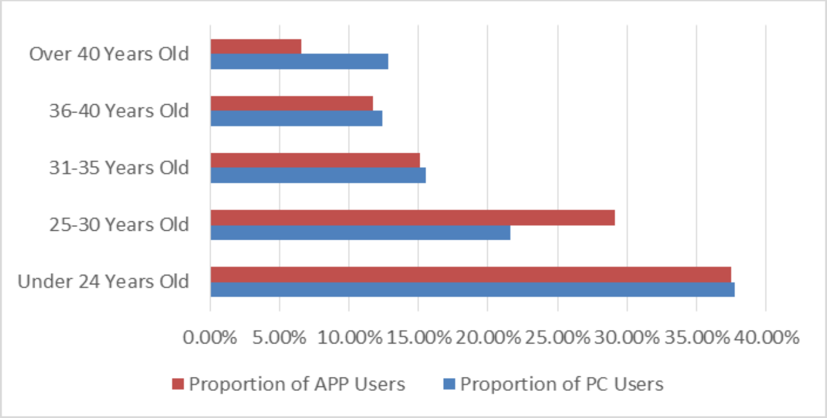


Figure 2. Age distribution of Weibo Users on PC and APP.

According to the survey, the fan groups of virtual spokespersons such as noonoouri employed by luxury brands are mainly concentrated in the 18-24 age groups. Virtual spokespersons such as noonoouri represent the ideal lifestyle of young people. In order to enhance "young consumers' willingness to purchase Valentino, the core of Valentino marketing strategy is gradually becoming younger. The Weibo has a wide range of younger audiences, which provides an opportunity for virtual spokespersons of Valentino and young consumer groups to form a para-social interaction relationship and also provides favorable conditions for the transformation of Valentino to younger generations.

4.2. Result and Discussion about Weibo Data

In order to test whether Valentino's use of virtual spokespersons for advertising will affect the formation of para-social interaction between Valentino and its consumers, this experiment takes Valentino as an example and collects the brand's official Weibo data from 2012 to present 3806 in total. The experiment uses whether there is a virtual spokesperson (VS) in the published Weibo advertisement content as an independent variable and use Weibo users' like (DZ), forward (ZF), and comment (PL) data on each original Weibo of Valentino brand as the criteria for judging the effect of its para-social interaction.

In order not to affect the results of the experiment, we screened the obtained Weibo data and excluded data containing celebrity spokespersons and certain brand marketing activities that have a greater impact on user likes and comments. Finally, from the remaining data, 3 original Weibo data including virtual spokesperson noonoouri and 97 ordinary advertising and marketing Weibo originals were selected. We sorted out a total of 100 original Weibo content and user replies and comments data and stored them in Excel form.

This experiment uses the SPSS statistical tool to count the average and standard deviation of the respective replies and comment data of the marketing Weibo containing virtual spokespersons in the Valentino brand official Weibo and the ordinary merchandise marketing Weibo (As shown in Table1). Then, we performed an independent sample t test on the data (As shown in Table2).

Although the average and standard deviation of the interactive data of marketing Weibo with virtual spokespersons in Valentino's official Weibo are higher than those of ordinary marketing Weibo, the significance of the values obtained by independent sample t-test is greater than 0.05 that is no statistically significant difference. The experimental results show that the use of virtual spokespersons for advertising by Valentino does not significantly affect the formation of para-social interactions between Valentino and their consumers.

Table 1. Results of Statistical Evaluation of Valentino Weibo Interactive Data Set.

VS		Number of Cases	Average Value	Standard Deviation	Standard Error Average
DZ	**Advertising Marketing Weibo with Virtual Spokesperson**	3	685.33	528.905	305.363
	General Product Advertising Marketing Weibo	97	62.96	39.559	4.017

ZF	**Advertising Marketing Weibo with Virtual Spokesperson**	3	426.33	681.732	393.598
	General Product Advertising Marketing Weibo	97	7.09	14.494	1.472
PL	**Advertising Marketing Weibo with Virtual Spokesperson**	3	210.00	279.807	161.547
	General Product Advertising Marketing Weibo	97	7.46	12.521	1.271

Table 2. Independent Sample Test Results of Valentino's Weibo Replies and Comments Interactive Data.

Levene Variance Equivalence Test		**Mean Equality T Test**						
F	**Significance**	**T**	**Degree of Freedom**	**Sig.(Double tail)**	**Mean Difference**	**Standard Error Difference**	**Difference 95% Confidence Interval**	
							Lower Limit	**Upper Limit**
261.589	0.000	12.476	98	0.000	622.375	49.886	523.377	721.372
		2.038	2.001	0.178	622.375	305.390	-691.176	1935.925
632.652	0.000	7.265	98	0.000	419.241	57.707	304.722	533.759
		1.065	2.000	0.398	419.241	393.601	-1274.242	2112.723
413.174	0.000	8.256	98	0.000	202.536	24.533	153.852	251.220
		1.254	2.000	0.337	202.536	161.552	-492.482	897.554

At present, Valentino marketing is still based on its exquisite craftsmanship and style. The promotion of virtual spokespersons is relatively lacking. Valentino has only published three promotional Weibos about virtual spokesperson noonoouri in its official Weibo account. In its official Weibo account and the data of user replies under each virtual spokesperson’s publicity Weibo was only about 100.

5. CONCLUSIONS

Based on the marketing data on virtual spokespersons and ordinary Weibo published in Valentino's official Weibo, this research found the following conclusions.

Although Valentino’s marketing strategies for virtual spokespersons and para-social interactions abroad are relatively complete, its social media marketing strategies are still in the development stage in our country. Valentino’s promotion of virtual spokespersons on its official Weibo is relatively lacking, which is not conducive to giving full play to the advantages of virtual spokespersons and not conducive to the construction of para-social interactions between consumers and Valentino.

In order to solve the problem of inadequate promotion of virtual spokespersons, Valentino can try to create a brand virtual character that can endorse the brand and has customer Q&A services and sales services, which can enhance the sense of interaction between Valentino virtual spokespersons and consumers to gain consumer trust, so as to better form a pseudo-social interaction relationship with them.

ACKNOWLEDGEMENTS

The research is supported by the Social Science Foundation of Shanghai (No. 2019BYY028), the Humanity and Social Science Research Project of Ministry of Education of China (No. 20YJAZH069, No. 20YJC740062), and the Guangdong Basic and Applied Basic Research Foundation (No. 2018A030313672).

REFERENCES

[1] Daniel A. Langer. Luxury Marketing and Management [M]. Beijing: China Renmin University Press, 2016.

[2] Jie Yao. An Analysis of the Promotion Strategy of Virtual Spokesperson in the Experience Economy. China Circulation Economy, 17:5-6, 2019.

[3] Xin Li. Unlock Virtual Spokesperson Marketing. Toy Industry, 10:36-37, 2019.

[4] Alan M Rubin, Elizabeth, Robert A Powell. Loneliness, Para-social Interaction, and Local Television News Viewing. Human Communication Research, 12(2):155-180, 1985.

[5] Lin Yang. The Use of New Media in Luxury Marketing-Taking the CHANEL Brand as An Example. News Research, 08:42-43, 2012.

[6] Jing Xu. How Does Luxury Advertising Make Use of Digital Media to Transform-Take Luxury Micro-Films as An Example. Journal of News Research, 7(16):302, 2016.

[7] Xuhui Wang, Wenqi Feng. The Influence of Social Interaction and Para-social Interaction in Virtual Brand Community on Brand Relationship Quality. Collected Essays on Finance and Economics, 05:78-88, 2017.

[8] Yiming Hu, Feifei Ding. Research on the Application of Advertising Virtual Spokesperson. The Fortune Times, 05:127, 2018.

AUTHORS

Dandan Yu, Shandong Key Laboratory of Language Resources Development and Application, Ludong University, 264025 Yantai, Shandong, China

SIGN LANGUAGE RECOGNITION FOR SENTENCE LEVEL CONTINUOUS SIGNINGS

Ishika Godage, Ruvan Weerasignhe and Damitha Sandaruwan

University of Colombo School of Computing, Colombo 07, Sri Lanka

ABSTRACT

It is no doubt that communication plays a vital role in human life. There is, however, a significant population of hearing-impaired people who use non-verbal techniques for communication, which a majority of the people cannot understand. The predominant of these techniques is based on sign language, the main communication protocol among hearing impaired people. In this research, we propose a method to bridge the communication gap between hearing impaired people and others, which translates signed gestures into text. Most existing solutions, based on technologies such as Kinect, Leap Motion, Computer vision, EMG and IMU try to recognize and translate individual signs of hearing impaired people. The few approaches to sentence-level sign language recognition suffer from not being user-friendly or even practical owing to the devices they use. The proposed system is designed to provide full freedom to the user to sign an uninterrupted full sentence at a time. For this purpose, we employ two Myo armbands for gesture-capturing. Using signal processing and supervised learning based on a vocabulary of 49 words and 346 sentences for training with a single signer, we were able to achieve 75-80% word-level accuracy and 45-50% sentence level accuracy using gestural (EMG) and spatial (IMU) features for our signer-dependent experiment.

KEYWORDS

Sign Language, Word-Level Recognition, Sentence-Level Recognition, Myo Armband, EMG, IMU, Supervised Learning.

1. INTRODUCTION

According to Wikipedia, communication is the act of conveying meanings from one entity or group to another through the use of mutually understood signs and semiotic rules. There are many approaches for communication. Such as voice and speech, writing, manual signs, and gestures etc.

These communication methods can be divided into two different forms. The first is verbal communication methods and the second, non-verbal communication methods. Verbal communication describe the processes of communicating with words, whether written or spoken. Non-verbal communication is defined as the process of using the wordless message to generate meaning. Examples of nonverbal communication include haptic communication, chronemic communication, gestures, body language, facial expressions and eye contact.

There is no doubt that communication plays a vital role in human life. Communication helps to share information and knowledge. It also helps humans make new relationships, and express ideas, feelings, emotions and thoughts.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 53-73, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112305

There are two conditions to be satisfied for a successful communication, namely,

1) There must be at least two parties who involve in the communication
2) Both parties must use a common communication platform

Most ordinary people (without any hearing/speaking disability) use verbal communication methods (e.g. voice and speech) for their communication. However, deaf and speaking-impaired people use non-verbal communication methods (mostly signs and gestures) for their communication. These two groups (ordinary people and deaf and speaking impaired people) therefore use different platforms for their communication. Because of this problem, there is a communication barrier between these two people groups when they need to communicate with each other.

Figure 1 demonstrates the communication barrier between a deaf person and an ordinary person. A deaf person uses sign language and the ordinary person uses voice or text. As mentioned previously, there are two conditions to be satisfied for a successful communication. Ordinary person to ordinary person and deaf person to deaf person communications satisfies those conditions. However, deaf person to ordinary person communication does not satisfy the second condition of using a common communication platform. In the figure, they attempt to use sign language as their communication platform which deaf person can understand but the ordinary person cannot. Since they do not use a common communication platform, their communication fails.

In the proposed solution, we create a sign language translator which can recognize sentence level continuous signings and translates them into a natural language. While it is translating signs into text/voice, it improves the practical usability of the system by employing a simple wearable device to capture the signs.

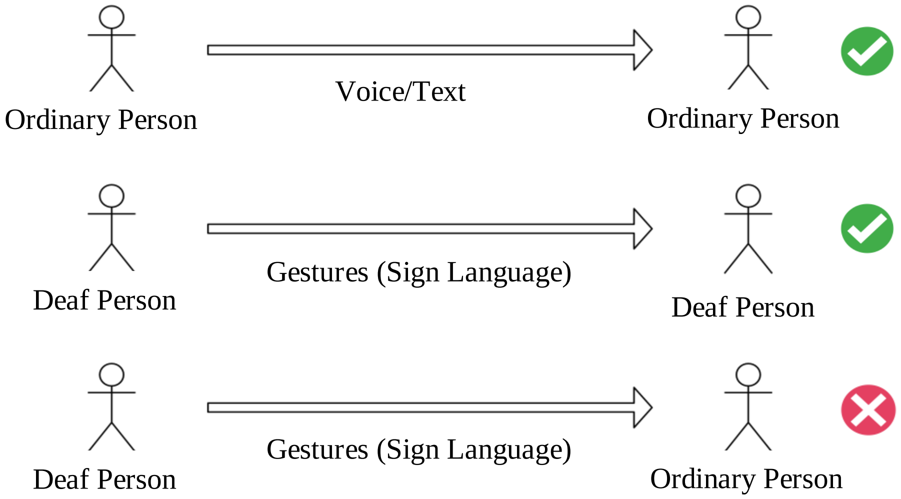


Figure 1. Communication methods

Figure 2 elaborates how our solution simulates a common platform. It captures the signs of a sign language and translates them into text/voice. As a result, the ordinary person would be able to understand the sign that the deaf person has performed. In this research, we do not concern ourselves with the communication in the other direction: i.e. from the hearing person to the hearing-impaired person. In our solution, we capture sentence level continuous signings in Sri Lankan Sign Language and translate them into Sinhala natural language.

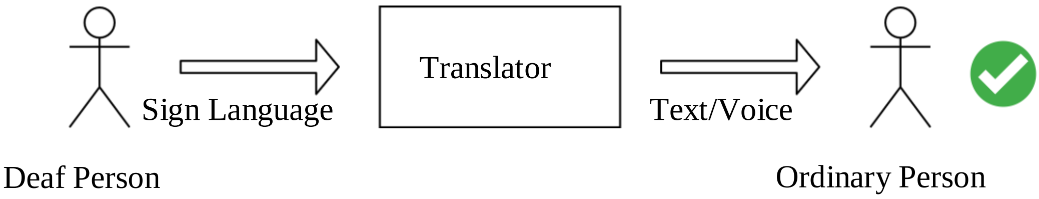


Figure 2. The communication method of the proposed solution

2. BACKGROUND

2.1. Sign Language

There are around 300 sign languages in the world [1]. These languages are different from each other. Word order in sentences can differ between these languages as well as from written text. Sign language is a visual language that incorporates gestures, facial expressions, head movements, body language and even the space around the speaker. Hand signs are the foundation of the sign language. Many signs are iconic, meaning the sign uses a visual image that resembles the concept it represents. Actions are often expressed through hand signals that mimic the action being communicated.

2.2. Sri Lankan Sign Language (SLSL)

Sri Lankan Sign Language is a visual-gestural language based on hand movements and the body (including facial expressions, lip moments, head movement). In Sri Lankan Sign Language, it can represent alphabets of normal languages (Sinhala, English) and it can represent other sings for each word. Currently, Sri Lankan Sign Language contains around 2000 signs [18, 19]. It also has regional signs across Sri Lanka.

British introduced the sign language to Sri Lanka. Hence, Sri Lankan Sign Language has been developed for years with the influence of British Sign Language (BSL). Because of that, there are some similarities in between Sri Lankan Sign Language and British Sign Language.

2.3. Electromyography (EMG) and Initial Measurement Units (IMU).

2.3.1. Electromyography (EMG)

Electromyography (EMG) is the detection and recording of the electrical signal produced by muscle tissue as it contracts. EMG depends on several factors such as the thickness and temperature of the skin, the thickness of the fat between the muscle and the skin, the velocity of the blood flow, and location of the sensors.

2.3.2. Initial Measurement Unit (IMU)

According to the Wikipedia, an IMU is an electronic device that measures and reports a body's specific force, angular rate, and sometimes the magnetic field surrounding the body, using a combination of accelerometers and gyroscopes, sometimes also magnetometers.

3. RELATED WORK

3.1. Word Level Sign Language Recognition Systems

3.1.1. Kinect Device Based Solution [2]

Kalin Stefanov and Jonas Beskow proposed a method for automatic recognition of isolated Swedish Sign Language (SSL) signs for the purpose of educational signing-based games. Two datasets consisting of 51 signs have been recorded from a total of 7 (experienced) and 10 (inexperienced) adult signers. Signer-dependent recognition rate is 95.3% for the most consistent signer. Hidden Markov Model (HMM) have been used as the model. Signer-independent recognition rate is on average 57.9% for the experienced signers and 68.9% for the inexperienced.

3.1.2. Data Glove Device Based Solution [3]

Wu jiangqin et al proposed a Chinese Sign Language recognition system based on data glove. In this paper, a simple word-level sign language recognition system is presented. 26 sign language words were used for this experiment. There are primarily 3 methods were used for sign language recognition. Such as template matching, neural networks and Hidden Markov Model HMM. The Recognition rate of testing samples is over 90%.

3.1.3. Leap Motion Device Based Solution [4]

Deepali Naglot and Milind Kulkarni proposed a system for recognition of 26 different alphabets of American Sign Language (ASL) using leap motion controller (LMC). LMC is 3D non-contact motion sensor which can track and detects hands, fingers, bones and finger-like objects. Multi-Layer Perceptron (MLP) is executed on a dataset of total 520 samples and Recognition rate of the proposed system is 96.15%.

3.1.4. Image/Video Based Solution (Vision Based) [5]

Manar et al introduce the use of different types of neural networks in human hand gesture recognition for static images as well as for dynamic gestures. A static gesture is a particular hand movement represented by a single image, while a dynamic gesture is a moving gesture represented by a sequence of images. This work focuses on the ability of neural networks to assist in Arabic Sign Language (ArSL) hand gesture recognition. This work focuses on the 28 letters of the Arabic alphabet. Fully recurrent architecture has had a performance with an accuracy rate of 95% for static gesture recognition.

3.1.5. EMG and IMU Based Solution [6]

Jian Wu et al proposed a real-time American SLR system leveraging fusion of surface electromyography (sEMG) and a wrist-worn inertial sensor at the feature level. A feature selection is provided for 40 most commonly used words and for four subjects. SVM was used as the classifier model. Their system achieves 95.94% recognition rate.

3.2. Sentence Level Sign Language Recognition Systems

3.2.1. Kinect Device Based Solution [7]

Edon Mustafa and Konstantinos Dimopoulos developed a system which uses SigmaNIL framework to recognize alphabet, number, word, and sentence of Kosova Sign Language. The recognition rate for one sentence from three testers is 73%.

3.2.2. Data Glove Device Based Solution [8]

Noor Tubaiz et al proposed a glove-based Arabic Sign Language recognition system using a novel technique for sequential data classification. The dataset contains 40 sentences using an 80-word lexicon. Data labelling is performed using a camera to synchronize hand movements with their corresponding sign language words. Modified k-Nearest Neighbor (MKNN) approach is used for classification. The proposed solution achieved a sentence recognition rate of 98.9%.

3.2.3. Image/Video Based Solution (Vision Based) [9]

Daniel Kelly et al presented a multimodal system for the recognition of manual signs and non-manual signals within continuous Irish sign language sentences. In this paper, they proposed a multichannel HMM-based system to recognize manual signs (hand gestures) and non-manual signals (E.g. facial expressions, head movements, body postures, and torso movements). Signer has to make pauses between words, to segment the words in a sentence. They have considered about 8 words. Using 4 words at a time they have created sentences. Their system achieved a detection ratio of 95.7%.

3.2.4. EMG and IMU Based Solution [10]

Xu Zhang et al presented a framework for hand gesture recognition based on the information fusion of a three-axis accelerometer (ACC) and multichannel electromyography (EMG) sensors. In this framework, the start and end points of meaningful gesture segments are detected automatically by the intensity of the EMG signals. 72 Chinese Sign Language (CSL) words and 40 CSL sentences are classified using a decision tree and multi-stream hidden Markov models. Overall word recognition accuracy is 93.1% and a sentence recognition accuracy is 72.5%.

We observed that EMG and IMU based solutions have sufficient accuracy, they can be enhanced as mobile solutions and they improve the practical usability of the system. Therefore, we planned to use EMG and IMU based device for this research. Instead of using electrodes, we chose Myo gesture control armband which is a commercial-off-the-shelf device for this research as the data capturing device [20]. The following 3.3 and 3.4 sections show the existing works which use the Myo gesture control armband as the data capturing device.

3.3. Word Level Sign Language Recognition Systems Using Myo Gesture Control Armband

Celal Savur and Ferat Sahin proposed a system [11] to identify recognize the American Sign Language alphabet letters (26) and a one for the home position. As a classification method, Support Vector Machine (SVM) and Ensemble Learning algorithm were used. Accuracies are 80% and 60.85% respectively. Only one hand use to perform gestures.

Prajwal Paudyal et al proposed SCEPTRE [12] which utilizes two non-invasive wrist-worn devices (Both arms were used) to decipher gesture-based communication. The system uses a multitiered template-based comparison system for classification on input data from accelerometer, gyroscope, and electromyography (EMG) sensors. They tried to identify 20 signs of American Sign Language and the system was able to achieve an accuracy of 97.72 % for ASL gestures.

3.4. Sentence Level Sign Language Recognition Systems Using Myo Gesture Control Armband

Best of our knowledge, we were unable to find literature which tries to recognize sentence level continuous signing using Myo gesture control armband. However, the proposed system used Myo gesture control armband to recognize sentence level continuous signings.

3.5. Related Research Projects about Sri Lankan Sign Language Interpretation

Dulan Manujith and Nihal Kodikara invented a Sinhala figure spelling interpretation system [13]. Herath et al proposed an image-based sign language recognition system for Sinhala sign language [14]. Kulaveerasingam et al invented a system which is gesture based intercommunication platform for hearing-impaired people [15]. Pumudu Fernando and Prasad Wimalaratne proposed sign language translation approach to Sinhalese language [16]. Madushanka et al introduced a framework for Sinhala sign language recognition and translation using a wearable armband [17].

4. DESIGN

Initially, we created a framework for recognizing sentence level continuous signings of Sri Lankan Sign Language and for translating them into a natural language (Sinhala). Figure 3 shows the flow of the study.

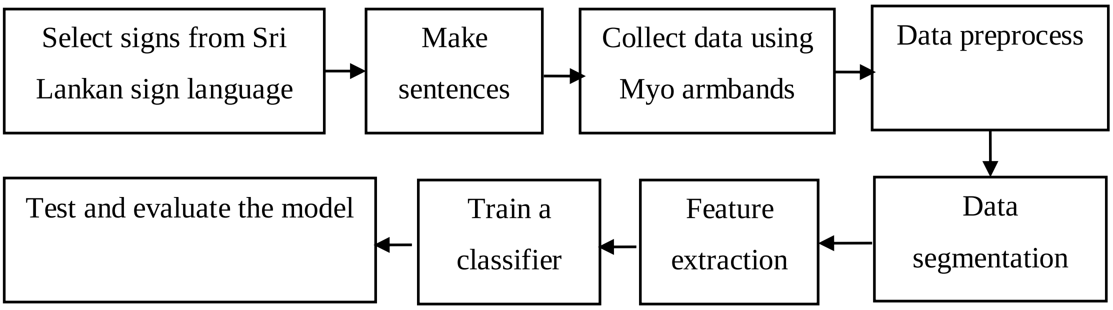

Select signs from Sri Lankan sign language
Make sentences
Collect data using Myo armbands
Data preprocess
Data segmentation
Feature extraction
Train a classifier
Test and evaluate the model

Figure 3. The flow of the study

4.1. Creating a Sri Lankan Sign Language Corpus

Sri Lankan Sign Language was selected as the sign language for this research project. There are around 2000 signs in Sri Lankan Sign Language [18, 19]. For this study, 49 of them were selected. The selected signs are common and useful signs in our day to day life. These 49 signs include nouns, pronouns nouns, and verbs only.

In this sentences creation process, we used SOV (Subject + Object + Verb) structure as the structure of the sentences. Each sentence consists of three words. Those are subject, object, and verb. 346 sentences were created using those 49 selected signs. Table 1 shows that 49 words which were selected as subjects, objects, and verbs. Figure 4 shows the frequencies of each sign which are selected for this study.

Table 1. Selected signs.

Class	Subject	Class	Object	Class	Verb
1	මම（*mama*）/I	19	මේසය (*mēsaya*) / Table	36	අදිනවා (*adinavā*) / Pull
2	අපි (*api*) / We	20	පුටුව (*puṭuva*) / Chair	37	අඳිනවා (*aňdinavā*) / Draw
3	ඔහු (*ohu*) / He	21	බර (*bara*) / Weight	38	අරිනවා (*arinavā*) / Open
4	ඇය (*æya*) / She	22	චිත්‍රය (*citraya*) / Painting	39	ඉගෙනගන්නවා (*igenagannavā*) / Study
5	ඔවුන් (*ovun*) / They	23	දොර (*dora*) / Door	40	ඉරනවා (*iranavā*) / Tear
6	අම්මා (*ammā*)/ Mother	24	ජනේලය (*janēlaya*) / Window	41	උණුකරනවා (*uṇukaranavā*) / Boil
7	තාත්තා (*tāttā*) / Father	25	ඉංග්‍රීසි (*iṁgrīsi*) / English	42	උයනවා (*uyanavā*) / Cook
8	අක්කා (*akkā*) / Elder Sister	26	පොත (*pota*) / Book	43	එල්ලනවා (*ellanavā*) / Hang
9	අයියා (*ayiyā*) / Elder Brother	27	පත්තරය (*pattaraya*) / News Paper	44	පලනවා (*palanavā*) / Split
10	නංගී (*naṁgī*) / Younger Sister	28	කඩදාසිය (*kaḍadāsiya*) / Paper	45	යනවා (*yanavā*) / Go
11	මල්ලී (*mallī*)/ Younger Brother	29	වතුර (*vatura*) / Water	46	බොනවා (*bonavā*) / Drink
12	දුව (*duva*)/ Daughter	30	මාලු (*mālu*) / Fish	47	ලියනවා (*liyanavā*) / Write
13	පුතා (*putā*) / Son	31	එළවළු (*eḷavaḷu*) / Vegetable	48	විසිකරනවා (*visikaranavā*) / Throw
14	නැන්දා (*nændā*)/ Aunt	32	රෙදි (*redi*) / Clothes	49	සිටුවනවා (*siṭuvanavā*) / Plant
15	මාමා (*māmā*)/ Uncle	33	දර (*dara*) / Firewood		
16	ආච්චී (*āccī*) / Grand Mother	34	ගෙදර (*gedara*) / Home		
17	සීයා (*sīyā*)/ Grand Father	35	ගස (*gasa*) / Tree		
18	යාලුවා (*yāluvā*) / Friend				

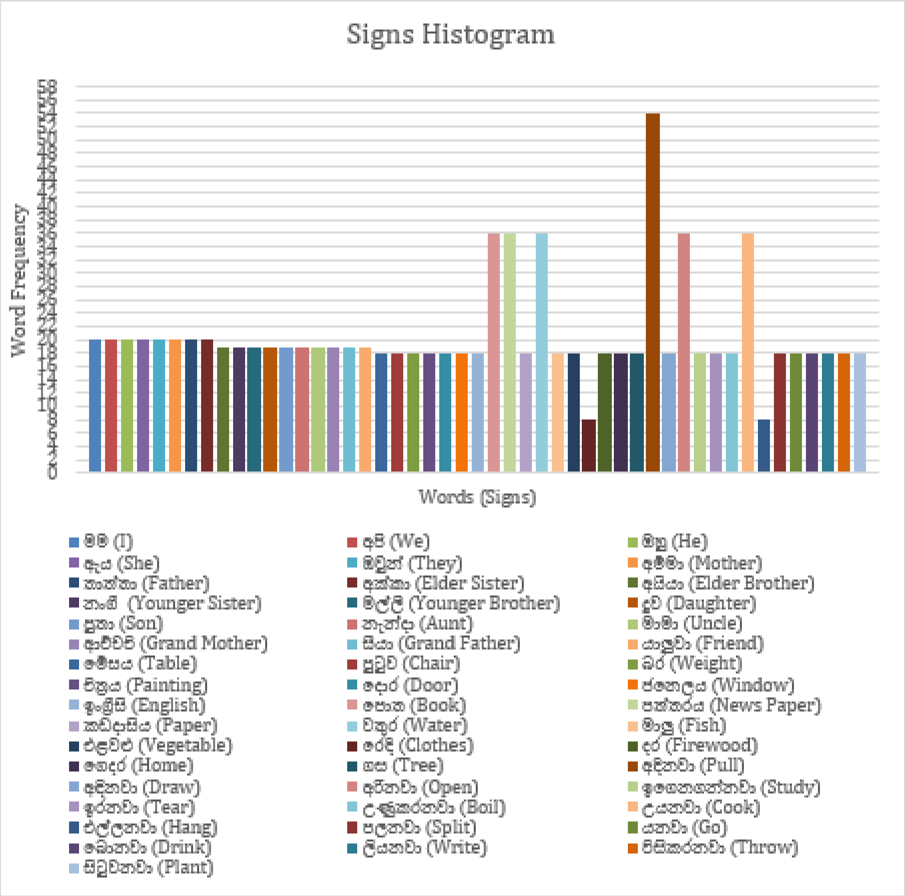


Figure 4. Signs histogram

4.2. Device Selection

In this research, Myo gesture recognition armband was selected as our data capturing device. This device was developed and introduced by Thalmic Labs Inc as a new way of using hand gestures to interact with computers and mobile devices (especially as an input/controlling device). Before Myo armband was selected as the data capturing device we had to consider three things.

1) Sign recognition accuracies of EMG and IMU based techniques
2) The mobility of the device
3) User convenience of the device

A Myo armband gives EMG and IMU data. There are 8 EMG signals and 10 IMU signals. Myo armband has 8 EMG sensors, 1 accelerometer sensor, 1 gyroscope sensor, and 1 magnetometer sensor.

4.3. Data Collection

After creating the sentences, data were collected using a sign language interpreter. Myo armband was used as our data collection device. Since we use both arms to perform signs, two Myo armbands were used. The Myo armbands were connected to two different computers using two Bluetooth adapters. After connecting armbands with the computers via Bluetooth, by running a

C++ program with the help of Myo SDK, EMG and IMU data were captured and stored in CSV files separately.

To avoid the speed variations when performing signs, a metronome was used as a supporting tool. A metronome is a device that produces an audible click or another sound at a regular interval that can be set by the user, typically in beats per minute. Thus, then the signer performs all the sentences (346) in the same rhythm. In the metronome, 5seconds were considered. In each second, the signer performed the particular sign according to the sentence. Rest sign was performed in 1st and 5th seconds. The first sign, second sign and third sign in the sentence were performed in 2nd, 3rd, and 4th seconds respectively. The metronome was screened in a separate display while performing signs. Figure 5 depicts the data collection design. Moreover, it's necessary to have a common starting and ending point for all the sentence to recognize a particular sentence when it gets started or ended.

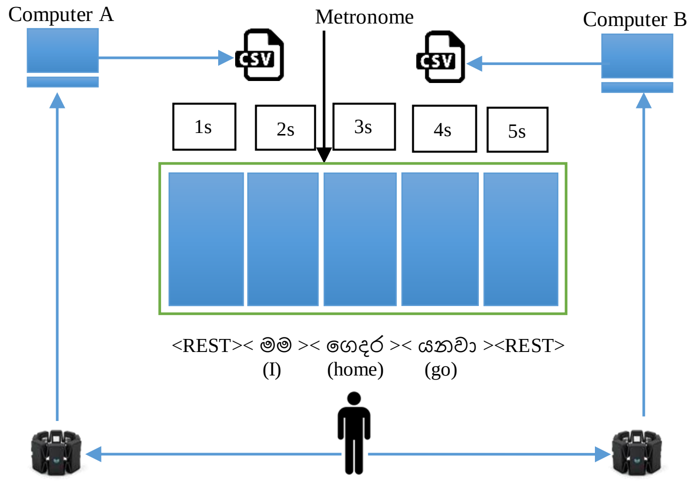


Figure 5. Data collection design

4.4. Data Pre-process

It is not a better idea to use raw data as it is for the classification process. Because there are many issues with the raw data. Such as unwanted data (e.g. noise), incomplete data, inconsistent data etc. Our collected data contain EMG data and IMU data. Therefore, we had to use digital signal processing (DSP) techniques to pre-process the collected raw data.

1) Pre-processing methods of EMG data

- Resampled the signals
- Removed the DC offset
- Applied full wave rectification
- Used low pass Butterworth filter
- Conducted zero-phase digital filtering

2) Pre-processing methods of IMU data

- Used the moving average filter

4.5. Data Segmentation

Each sentence has 3 words and each signal contains 3 signs. The aim of this step is to segment each sign separately. As a result of segmentation, there will be 3 segments per sentence. For a single sentence, one Myo armband gives 18 signals (8 EMG and 10 IMU). Since two armbands were used for the data collection, we had to segment 36 signals and saved the segmented signs separately.

We carried out a manual segmentation method. Since we used a metronome as a supporting tool, we knew that the length of a sentence which is 5 seconds and the rest sign was performed in 1st and 5th second. First, second and third signs in the sentence were performed in 2nd, 3rd, and 4th seconds respectively. Since 2nd, 3rd, and 4th seconds contain the valid signs of a particular sentence, all signals were segmented within that each time period.

4.6. Feature Extraction

Features are the unique attributes of a particular data. Features are the input to the machine learning models. According to the existing work [17], we selected the following features.

1) Mean Absolute Value

$$MAV = \frac{1}{N}\sum_{n=1}^{N}|x_n|$$

2) Variance

$$VAR = \frac{1}{N-1}\sum_{n=1}^{N}x_n^2$$

3) Standard Deviation

$$SD\ (\sigma) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Since we are interested in 3 features, there are 108 (=36*3) features for an each sign.

4.7. Feature Reduction and Feature Selection Methods

Feature reduction and selection are two techniques of feature engineering. Basically, what it does is, identifying the most important features. In this study, we have used below feature reduction and feature selection methods.

1) PCA - Principal Component Analysis
2) US - Univariate Selection
3) SVD - Singular Value Decomposition
4) RFE - Recursive Feature Elimination
5) RF - Random Forest

4.8. Machine Learning Model Training

In this research project, we applied supervised learning techniques. Because this is a classification problem. Therefore, we had to train a classifier. We selected 5 classifiers and trained all them using the training data. The training data set was composed of 241 sentences

(723 words) and the test data set included 105 sentences (315 words). We got the 10-fold cross-validation accuracy of all the models and selected the highest accuracy given classifier as the final classifier for this study.

1) NB - Gaussian NB
2) LDA - Linear Discriminant Analysis
3) RFC - Random Forest
4) LR - Logistic Regression
5) RC - Ridge Classifier

Then we created a framework for carrying out the task covered in previously in real-time. Real-time hand gesture recognition is one of the most challenging research areas in the human computer interaction field. In the initial experiment, we conducted an offline training and offline testing. However, in this experiment, we conducted offline training and online testing.

Data capturing, pre-processing, segmentation and feature extraction techniques are the same as the initial experiment. However, we used a previously trained model to recognize and translate sentence level continuous signings in real-time. Figure 6 shows the flow diagram of the real-time classification experiment.

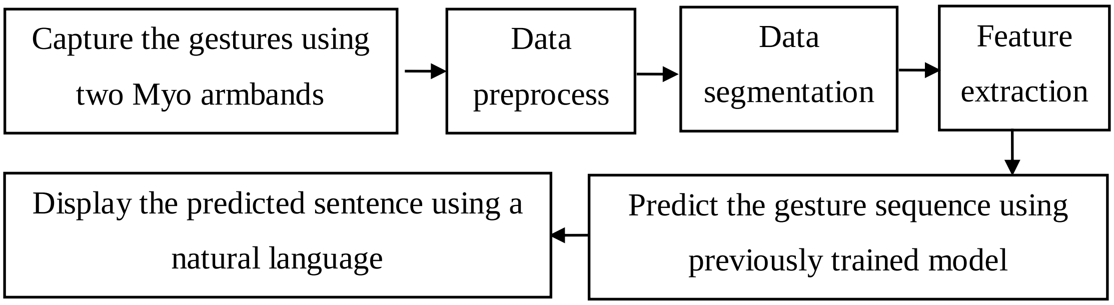

Capture the gestures using two Myo armbands
Data preprocess
Data segmentation
Feature extraction
Predict the gesture sequence using previously trained model
Display the predicted sentence using a natural language

Figure 6. Flow diagram of real-time classification experiment

5. Results and Evaluation

As the first experiment, we input all features to each model. As mentioned in the design section, 5 models were trained using 5 Machine Learning algorithms and feature vectors. We then compared the cross-validation accuracies of each model. The cross-validation results are given in Table 2. The Linear Discriminant Analysis classifier performed best for average 10-fold cross-validation accuracy.

Since Linear Discriminant Analysis (LDA) classifier showed the highest cross-validation accuracy, we selected LDA as the classifier for the final study. We then trained the LDA classifier using all the features (108). Finally, we got a word level testing accuracy of between 75% - 80%. The full sentence level accuracy varies from 45% to 50%. Figure 7 shows the confusion matrix of the LDA classifier while Table 3 shows the average precision, recall, and F1-score of the LDA model.

Table 2. Average 10-fold cross validation accuracy.

Model	**Average 10-Fold cross-validation score**	**standard deviation**
Logistic Regression (LR)	0.597774	0.047929
Linear Discriminant Analysis (LDA)	0.761796	0.064298
Ridge Classifier (RC)	0.675114	0.067444
Random Forest Classifier (RFC)	0.603387	0.052306
Gaussian Naïve Bayes (NB)	0.560731	0.067124

Table 3. Average precision, recall and F1-score of the LDA model.

Precision	**Recall**	**F1-score**
0.81	0.79	0.79

Precision - Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. The question that this metric answer is of all signs that labeled as a correct sign, how many actually correct signs? High precision relates to the low false positive rate. We have got 0.81 average precision value which is pretty good.

Recall (Sensitivity) - Recall is the ratio of correctly predicted positive observations to all observations in actual class. The question recall answer is: Of all the signs that have true class, how many did we label? We have got an average recall of 0.79 which is good for this model as it's above 0.5.

F1-score - F1-score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1-score is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have a similar cost. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall. In our case, the average F1-score is 0.79. Table 4, 5 and 6 show the class, precision, recall and F1-score of the LDA model in different F1-score ranges.

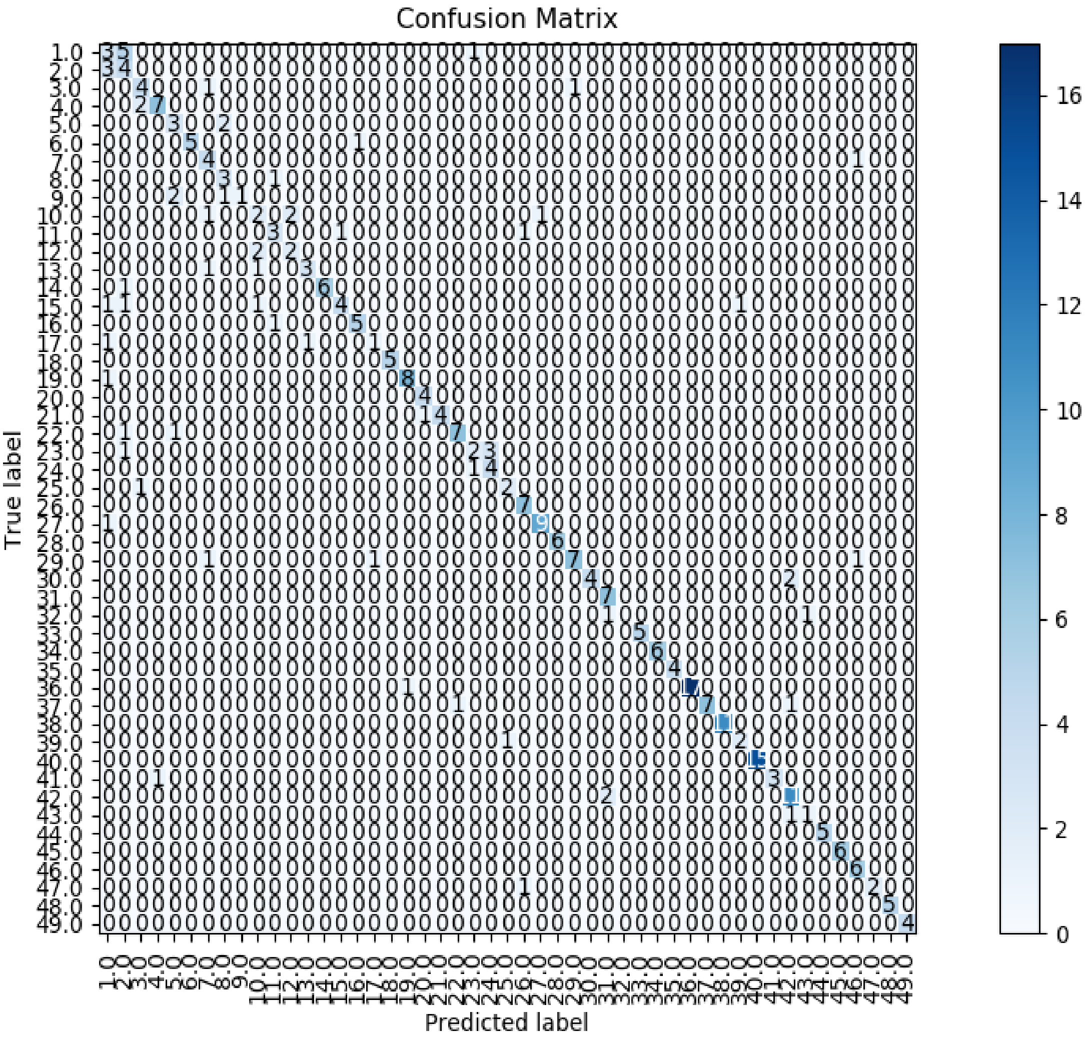


Figure 7. Confusion matrix

Table 4. Precision, recall and F1-score < 0.6.

Class	Precision	Recall	F1-score	Support
1	0.3	0.33	0.32	9
2	0.31	0.57	0.4	7
5	0.5	0.6	0.55	5
9	1	0.25	0.4	4
10	0.33	0.33	0.33	6
12	0.5	0.5	0.5	4
17	0.5	0.33	0.4	3
23	0.5	0.33	0.4	6
32	0	0	0	2
43	0.5	0.5	0.5	2

Table 5. Precision, recall and 0.6 <= F1-score <=0.8.

Class	Precision	Recall	F1-score	Support
3	0.57	0.67	0.62	6
7	0.5	0.8	0.62	5
8	0.5	0.75	0.6	4
11	0.6	0.6	0.6	5
13	0.75	0.6	0.67	5
15	0.8	0.5	0.62	8
24	0.57	0.8	0.67	5
25	0.67	0.67	0.67	3
29	0.88	0.7	0.78	10
30	1	0.67	0.8	6
39	0.67	0.67	0.67	3
42	0.73	0.85	0.79	13
47	1	0.67	0.8	3

Table 6. Precision, recall and F1-score > 0.8.

Class	Precision	Recall	F1-score	Support
4	0.88	0.78	0.82	9
6	1	0.83	0.91	6
14	1	0.86	0.92	7
16	0.83	0.83	0.83	6
18	1	1	1	5
19	0.89	0.89	0.89	9
20	0.8	1	0.89	4
21	1	0.8	0.89	5
22	0.88	0.78	0.82	9
26	0.78	1	0.88	7
27	0.9	0.9	0.9	10
28	1	1	1	6
31	0.7	1	0.82	7
33	1	1	1	5
34	1	1	1	6
35	1	1	1	4
36	1	0.94	0.97	18
37	1	0.78	0.88	9
38	1	1	1	11
40	1	1	1	15
41	1	0.75	0.86	4
44	1	1	1	5
45	1	1	1	6
46	0.75	1	0.86	6
48	1	1	1	5
49	1	1	1	4

However, F1-scores of classes 1, 2, 5, 9, 10, 12, 17, 23, 32 and 43 are less than 0.60 (Table 4). Especially, F1-score of class 32 is 0.00. The reason would be there are not enough examples (There are only 2 examples). Table 7 shows the category of each class which has the F1-score less than 0.6.

Table 7. Category of each class which has the F1-score less than 0.6.

Classes	Category (subject, verb, object)
1, 2, 5, 9, 10, 12, 17	Subject
23, 32	Object
43	Verb

According to Table 7 above, we can observe that most of the classes which have F1-score less than 0.60 belong to the subject category. Therefore, we can come to the decision that our model is unable to recognize signs which belong to the subject category. By the way, it is not possible to observe a clear diagonal and values are spread in the class range 1 – 18 in the confusion matrix (Figure 7). Therefore, we can confirm that our model is unable to recognize signs which appear in the subject category of the sentence more than those in other categories by looking at the confusion matrix further.

Figure 8 shows the ROC curve of the LDA classifier. It shows that all classes (49 signs) have high percentages of the area under the ROC curve. A rough guide for classifying the accuracy of a diagnostic test is the traditional academic point system. Table 8 shows AUC-ROC results according to the academic point system. According to the traditional academic point system also we can confirm that all the signs belong to the excellent category. Finally, we can conclude that model performed well at word level classification in the continuous sentence signing task.

As mentioned previously, the sentence level accuracy of the recognizer varies between 45% and 50%. A sentence consists of 3 words (subject + object + verb). If at least one of the signs is predicted wrongly, the entire sentence will be classified as incorrect. Table 9 shows, how the position of the sign contributes to sentence level errors.

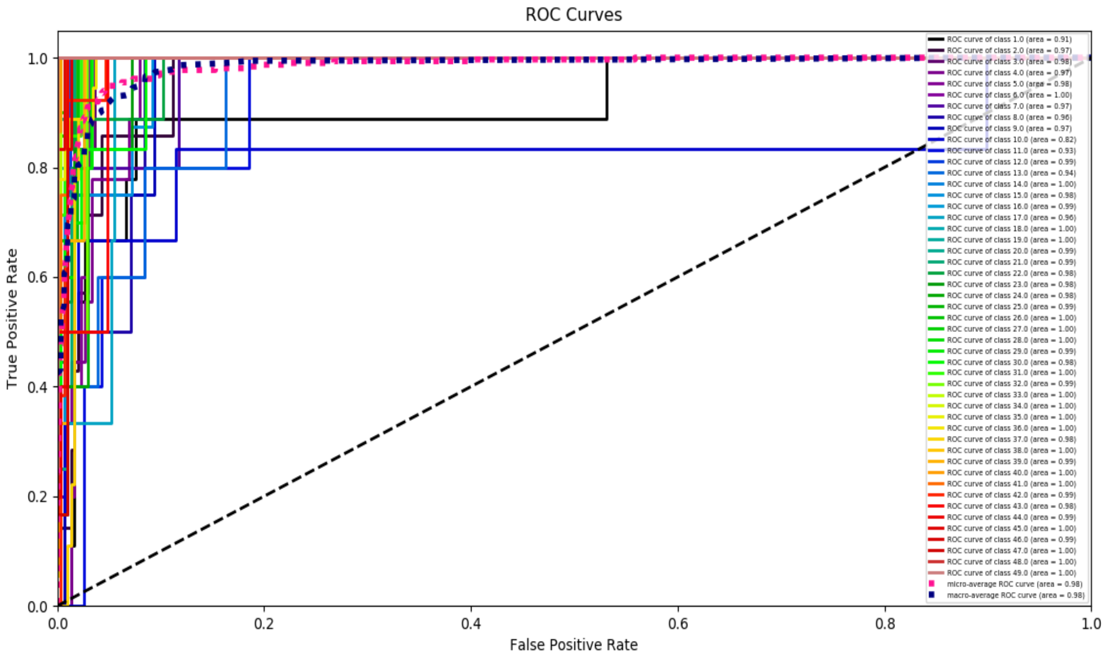


Figure 8. ROC curve

Table 8. AUC-ROC results according to the academic point system.

Points	**No. of classes (Signs)**
0.90 - 1.00 = excellent (A)	49 (all signs)
0.80 - 0.90 = good (B)	0
0.70 - 0.80 = fair (C)	0
0.60 - 0.70 = poor (D)	0
0.50 - 0.60 = fail (F)	0

Table 9. The contribution of positions of sign for sentences misclassification.

1st Sign (Subject)	**2nd Sign (Object)**	**3rd Sign (Verb)**	**No. of Misclassified Sentences (n)**	**Percentage (n/104) *100%**
Misclassified	Correctly Classified	Correctly Classified	31	29.8%
Misclassified	**Misclassified**	Correctly Classified	5	4.8 %
Misclassified	**Misclassified**	**Misclassified**	1	0.9 %
Misclassified	Correctly Classified	**Misclassified**	2	1.9 %
Correctly Classified	**Misclassified**	Correctly Classified	9	8.7 %
Correctly Classified	**Misclassified**	**Misclassified**	3	2.9 %
Correctly Classified	Correctly Classified	**Misclassified**	3	2.9 %
Number of misclassified sentences (total)			54	51.9 %

According to Table 9 above, we can observe that predicting the 1st sign of a sentence incorrectly accounts for 57% of all errors. Therefore, we can conclude that our model is weak at recognizing the 1st sign (Subject of the sentence) correctly relative to other positions. We have already discussed this issue by looking at the confusion matrix (Figure 7) and the F1-score (Table 4, 5 and 6). Misclassification of one sign directly affects the overall sentence level accuracy. We observed that there are two main reasons for such misclassification.

1) Similarities of signs
2) Incorrect sign segmentation

Then we wanted to experiment, how feature reduction and feature selection techniques affect the accuracy of the model. We observed the 10-fold cross-validation accuracy of each classifier with 5 feature reduction and selection techniques.

We reduced the original number of features (108) to 20, 40, 60, 80 and 100 by using the above mentioned 5 methods. However, we were unable to observe any significant improvement of the model, when we trained the model using 20 and 60 features. The model did show significant improvement in the 40, 80 and 100 feature cases.

Figure 9 shows how the cross-validation accuracy varies in each model after training them with the reduced set of 40 features. The LDA model shows the highest cross-validation accuracy (0.757831) after reducing the features using the random forest method. The LDA model showed the highest cross-validation accuracy (0.780723) after reducing the features to 80 using the principal component analysis (PCA) method. The LDA model showed the highest cross-validation accuracy (0.791566) after reducing the features to 100 using the singular value decomposition (SVD) method.

Initially, we had 108 features and the accuracy varied between 75% and 80% (Baseline). However, we observed that the LDA model has the highest cross-validation accuracy even with reduced features. The accuracies and numbers of features can be summarized as follows.

1) 40 features - 0.757831
2) 80 features - 0.780723
3) 100 features - 0.791566

Even though 80 feature and 100 features instances have high accuracy than 40 feature instances, the number of features is closer to the initial number of features (108). In practical deployments, we need to consider both the model's accuracy and the number of features in order to select the best feature reduction method. Hence, we selected the 40 feature (random forest) model as the final model.

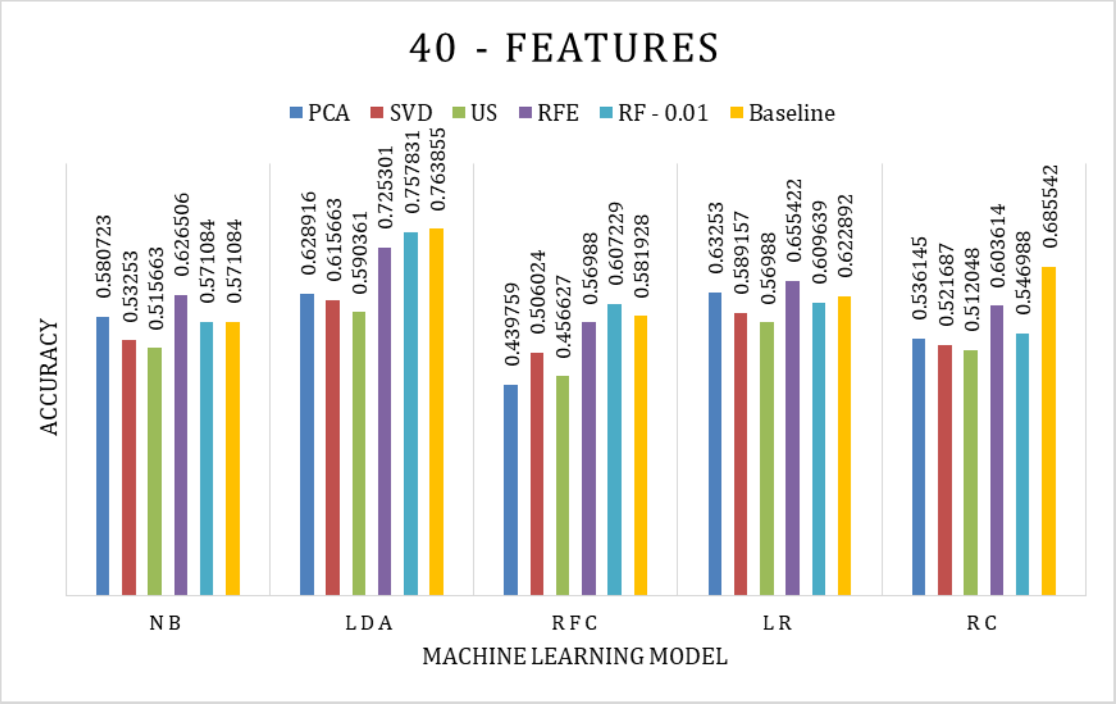


Figure 9. Cross-validation accuracies of each model using the reduced set of 40 features

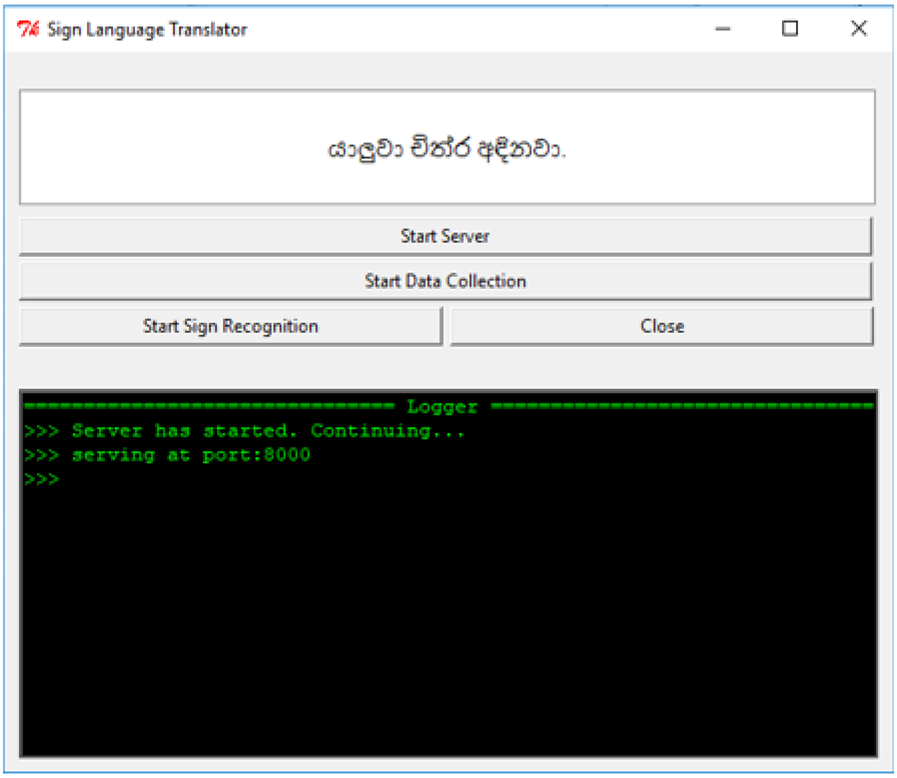
We have conducted the above mentioned experiments as offline experiments which all the training data and testing data collected previously. Then pre-process, segment, extract the features from the raw data and finally trained the classifier. As the next experiment, we wanted to conduct gesture recognition in a real-time manner. Because the final goal is to use this system in real-world scenarios. Here, we used two previously trained classifiers to predict the signs.

1) The classifier which has been trained using all the features (108 features) (Classifier-1)
2) The same classifier which has been trained after feature reduction (40 features). (Classifier-2)

Average prediction time of a sentence using classifier-1 (LDA and 108 features) in real-time is 17.4 seconds. Average prediction time of a sentence using classifier-2 (LDA and 40 features) in real-time is 13.6 seconds.

We can observe that the classifier-2 which was trained using 40 features for the sentence prediction shows less prediction time than classifier-1 in the real-time scenario. Even though, that 13.6 seconds of time is not suitable for the real-time scenario, we can observe that the prediction time of a sentence is reduced when we reduced the number of features. Therefore, we can state that there is an effect to the prediction time when we reduced the features.

We translated the gestures in a real-time manner. In order to do that we have created a python application. Example of a real-time classification output is shown in Figure 10. It shows the output as “යාලුවා චිත්‍ර අඳිනවා” (*yāluvā citra aňdinavā*). (Friend draws paintings).



යාලුවා චිත්‍ර අඳිනවා.

```
================================ Logger ================================
>>> Server has started. Continuing...
>>> serving at port:8000
>>>
```

Figure 10. Example output of a real-time gesture classification

5.1. Comparison of the Results of the Proposed Solution and Related Work

This research project is an extended version of previous research which has been conducted at University of Colombo School of Computing and title of that publication is “Framework for Sinhala Sign Language Recognition and Translation Using a Wearable Armband” [17] 2016. Prajwal Paudyal et al proposed another work which is SCEPTRE [12] 2016. Table 10 shows the comparison between the proposed solution and the above mentioned two research projects ([12], [17]).

Table 10. The comparison of the results of the proposed solution and two main references.

	Myo Armband [17] (EMG and IMU based solution)	**Myo Armband [12] (EMG and IMU based solution)**	**Proposed Solution**
Sign Language	Sri Lankan Sign Language	American Sign Language	Sri Lanka Sign Language
Word Level	Yes	Yes	Yes
Sentence Level	No	No	Yes
User Dependent	Yes	Yes	Yes
Accuracy (around)	100% (Word Level)	97.72% (Word level)	75%-80% (Word Level) 45%-50% (Sentence Level)
Number of Signs	3	20	49 (Words) 346 (Sentences)
Method	ANN	Multitiered template-based comparison system	Linear Discriminant Analysis
Real-time	No	Yes	Yes
Real-time recognition time	-	0.552 S (Word Level)	13.6 S (Sentence Level)

6. CONCLUSION

The aim of this research was to bridge the communication gap between hearing/speaking impaired and ordinary people by proposing a framework for recognize sentence level continuous signings of Sri Lankan Sing Language and translate them into a natural language (Sinhala).

In order to conduct the research, we created a dataset using a single sign language interpreter and Sri Lankan Sign Language was selected as the sign language. After completing this research project, that dataset will be publicly available. Then we trained models and got promising results for sign language recognition for both word level and sentence level continuous signings which are 75%-80% accuracy for the word level and 45% - 50% accuracy for the sentence level continuous signing.

The accuracy of the recognition of sentence level continuous signings in real time manner vary between 45% - 50%. However, we used the model which is trained using only 40 features after feature reduction for this scenario, because after feature reduction, we got less prediction time (13.6s) than prediction time when we used all features (17.4s).

Finally, the proposed solution improves the usability and mobility, because we use the wireless, lightweight wearable device and we did not use any unnatural method to identify the moment epenthesis.

7. FUTURE WORK

Our proposed solution showcased a proper outcome based on the scope of the research study which can be extended in several ways.

1) Increase the number of signs
2) Increase the words per sentence
3) Identify an automatic way to segment the signs
4) Reduce real-time classification time

REFERENCES

[1] "List of sign languages", En.wikipedia.org. [Online]. Available: https://en.wikipedia.org/wiki/List_of_sign_languages. [Accessed: 01- Jan- 2019]

[2] K. Stefanov and J. Beskow, "A Real-time Gesture Recognition System for Isolated Swedish Sign Language Signs", Proceedings of the 4th European and 7th Nordic Symposium on Multimodal Communication (MMSYM 2016), Copenhagen, 29-30 September 2016, no. 141, pp. 18-27, 2017. [Accessed 1 January 2019]

[3] Wu jiangqin, Gao wen, Song yibo, Liu wei and Pang bo, "A simple sign language recognition system based on data glove", ICSP '98. 1998 Fourth International Conference on Signal Processing (Cat. No.98TH8344). Available: 10.1109/icosp.1998.770847 [Accessed 1 January 2019]

[4] D. Naglot and M. Kulkarni, "Real time sign language recognition using the leap motion controller", 2016 International Conference on Inventive Computation Technologies (ICICT), 2016. Available: 10.1109/inventive.2016.7830097 [Accessed 1 January 2019]

[5] M. Maraqa, F. Al-Zboun, M. Dhyabat and R. Zitar, "Recognition of Arabic Sign Language (ArSL) Using Recurrent Neural Networks", Journal of Intelligent Learning Systems and Applications, vol. 04, no. 01, pp. 41-52, 2012. Available: 10.4236/jilsa.2012.41004 [Accessed 1 January 2019]

[6] J. Wu, Z. Tian, L. Sun, L. Estevez and R. Jafari, "Real-time American Sign Language Recognition using wrist-worn motion and surface EMG sensors", 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN), 2015. Available: 10.1109/bsn.2015.7299393 [Accessed 1 January 2019]

[7] E. Mustafa and K. Dimopoulos, "Sign Language Recognition using Kinect", Conference: Conference: 9th South East European Doctoral Student Conference, pp. 271-285, 2014. [Accessed 1 January 2019]

[8] N. Tubaiz, T. Shanableh and K. Assaleh, "Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode", IEEE Transactions on Human-Machine Systems, vol. 45, no. 4, pp. 526-533, 2015. Available: 10.1109/thms.2015.2406692 [Accessed 1 January 2019]

[9] D. Kelly, J. Reilly Delannoy, J. Mc Donald and C. Markham, "A framework for continuous multimodal sign language recognition", Proceedings of the 2009 international conference on Multimodal interfaces - ICMI-MLMI '09, 2009. Available: 10.1145/1647314.1647387 [Accessed 1 January 2019]

[10] Xu Zhang, Xiang Chen, Yun Li, V. Lantz, Kongqiao Wang and Jihai Yang, "A Framework for Hand Gesture Recognition Based on Accelerometer and EMG Sensors", IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 41, no. 6, pp. 1064-1076, 2011. Available: 10.1109/tsmca.2011.2116004 [Accessed 1 January 2019]

[11] C. Savur and F. Sahin, "American Sign Language Recognition system by using surface EMG signal", 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2016. Available: 10.1109/smc.2016.7844675 [Accessed 1 January 2019]

[12] P. Paudyal, A. Banerjee and S. Gupta, "SCEPTRE", Proceedings of the 21st International Conference on Intelligent User Interfaces - IUI '16, 2016. Available: 10.1145/2856767.2856794 [Accessed 1 January 2019]

[13] D. M. Wathugala and N. D. Kodikara, "A Sinhala finger spelling interpretation system using nearest neighbor classification", proc of 4th International infonnation technology coference Envisioning an eNation, Colombo, Sri Lanka, 2002. [Accessed 1 January 2019]

[14] H. C. M. Herath, W. A. L. V. Kumari, W. A. P. B. Senevirathne and M. B. Dissanayake, "Image based sign language recognition system for Sinhala sign language", Proceedings of SAITM Research Symposium on Engineering Advancements 2013, pp. 107-110, 2013. [Accessed 1 January 2019]

[15] N. Kulaveerasingam, S. Wellage, H. M. P. Samarawickrama, W. M. C. Perera and J. Yasas, “The Rhythm of Silence” - Gesture Based Intercommunication Platform for Hearing-impaired People (Nihanda Ridma)", 2014. [Accessed 1 January 2019]

[16] P. Fernando and P. Wimalaratne, "Sign Language Translation Approach to Sinhalese Language", GSTF Journal on Computing (JoC), vol. 5, no. 1, 2016. Available: 10.7603/s40601-016-0009-8 [Accessed 1 January 2019]

[17] A. Madushanka, R. Senevirathne, L. Wijesekara, S. Arunatilake and K. Sandaruwan, "Framework for Sinhala Sign Language recognition and translation using a wearable armband", 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), 2016. Available: 10.1109/icter.2016.7829898 [Accessed 1 January 2019]

[18] "Sri Lankan Sign Language", Lankasign.lk. [Online]. Available: http://www.lankasign.lk/index.html. [Accessed: 01- Jan- 2019]

[19] Sri Lanka Sign Dictionary, 2nd ed. SRI LANKA CENTRAL FEDERATION OF THE DEAF, 2012

[20] "Myo Gesture Control Armband tech specs", Welcome to Myo Support. [Online]. Available: https://support.getmyo.com/hc/en-us/articles/202648103-Myo-Gesture-Control-Armband-tech-specs. [Accessed: 01- Jan- 2019]

RELATION EXTRACTION BETWEEN BIOMEDICAL ENTITIES FROM LITERATURE USING SEMI- SUPERVISED LEARNING APPROACH

Saranya M[1], Arockia Xavier Annie R[2] and Geetha T V[3]

[1]Computer Science and Engineering, CEG, Anna University, India
[2]Assistant Professor, Computer Science and Engineering,
CEG, Anna University, Chennai, India
[3]UGC-BSR Faculty Fellow, Computer Science and Engineering, former
Dean CEG, Anna University, Chennai, India

ABSTRACT

Now-a-days, people around the world are infected by many new diseases. The cost of developing or discovering a new drug for the newly discovered disease is very high and prolonged process. These could be eliminated with the help of already existing resources. To identify the candidates from the existing drugs, we need to extract the relation between the drug, target and disease by textming a large-scale literature. Recently, computational approaches which is used for identifying the relationships between the entities in biomedical domain are appearing as an active area of research for drug discovery as it needs more man power. Due to the limited computational approaches, the relation extraction between drug-gene and gene-disease association from the unstructured biomedical documents is very hard. In this work, we proposed a semi-supervised approach named pattern based bootstrapping method to extract the direct relations between drug, gene and disease from the biomedical literature. These direct relationships are used to infer indirect relationships between entities such as drug and disease. Now these indirect relationships are used to determine the new candidates for drug repositioning which in turn will reduce the time and the patient's risk.

KEYWORDS

Text mining, drug discovery, drug repositioning, bootstrapping, machine learning.

1. INTRODUCTION

For developing the new chemical compound into the market for treating appropriate disease is called as drug development or design process which is very expensive and takes minimum 12-15 years from the starting stage to the marketing. Currently, the arrival of new diseases is increased and most of those are not treating with proper vaccine or medicine (Cummings J. 2021). To produce the proper medicine, the molecular level of the diseases must be understood by the scientists and it needs domain experts over various resources. Even though the amount and the time spent on designing a drug, there is no guarantee for the success of drug. During the interval of 2006-2015 only 9.6% was the attainment level of the chemicals entering into the trail (Hwang et al. 2016). A well-known alternative way to eliminate the risk and cost of discovering new drug is drug repurposing, i.e. finding new candidate (disease) for drugs that are already available in the

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 75-84, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112306

market (Talevi et al, 2020). Drug repositioning (Rudrapal M et al. 2020) diminishes the risk, time, cost and struggle during the early stages of drug discovery. To determine new candidates for available chemicals several methods have been done scientific publications, Electronic Health Records (EHR), health forums, clinical trial reports, etc. (Shahab 2017). Computational methods can be broadly classified into knowledge-based, similarity-based and network-based inference methods for extracting the biomedical association.

To extract the useful information from the unstructured biomedical literature, text mining and Natural Language Processing techniques (NLP) are used to make it in an understandable form. Most fundamental step in extracting the relation between the entities is recognizing or tagging the respective words as drug, target, disease with the help of Named Entity Recognition (NER) technique. After, the relation is extracted from the unstructured text via many approaches like co-occurrence based, rule based and machine learning based. Co-occurrence based approach is very easy and simple. The entities are associated with each other if they co-occurred frequently in a sentence, abstract or the documents. In PPI extraction found that the proteins are associating with each other when two proteins are co-occurred together across more abstracts. Co-occurrence based approach does not work well for the sentence or document that has multiple entities and the sentence which has negative relation between the entities. For example, “During pregnancy the patients are not advised to take ibuprofen”. From this sentence co-occurrence based method not able to identify the negative relation.

To overcome this Zhao et al in 2017 has introduced rule based method called regular expressions with the help of the word-level features and grammatical features namely Parts Of Speech (POS) tagging, dependency parsing, phrasal argument structures, predicate structures, syntactic and semantic analysis for preparing the rule definition and this leads to increased performance. Though this method improves the performance, it is very difficult to build the rule for variety of sentences, which needs rich domain expert and more time. Automatically generating the pattern gives the solution to the problem of rule-based approach. For extracting, the drug-side-effect relations from MEDLINE documents Xu & Wang (2014) generated the patterns from the POS tags and verbs automatically and it produces better performance than the manually defined patterns. But, sometimes the generated patterns are too generalized and it does not handle all varieties of sentences.

Next, supervised learning methods are used to extract the association. Mostly, supervised methods use n-dimensional feature vector or kernel functions for classifying the sentences. Features may be bag-of-words, syntactic (POS tag, chunk tag), lexical, semantic knowledge. For discovering Drug-Drug Interactions (DDIs) from the biomedical literature, Zhang et al. (2012) used a single hash subgraph pairwise kernel method effectively. After a while SVM (Support Vector Machine), Naïve Bayes, BeFree (bravo et al. 2015) algorithms were used for relation classification. From the above discussion, we concluded that the supervised learning approach requires powerful annotated corpora, but it requires longer time and more man power.

To migrate from the issues of supervised learning approach, researchers utilize the unsupervised learning approach. Initially, Madkour et al (2007) has introduced the BioNoculars method to extract PPI from MEDLINE corpus by generating a pattern using NER, POS tag followed by graph based mutual reinforcement method for extracting pattern from the literature. Though unsupervised approach extracts does not require annotated corpora and extracting more relations than other methods, precision is very poor. Erkan et al. (2007) introduced the method for PPI extraction namely transductive SVM with two types of similarity functions. To build a model when there is less annotated corpus, semi-supervised learning can be hired. Hence, we have designed a semi-supervised algorithm called pattern based bootstrapping to extract different biomedical associations between various entities from text (Batista et al. 2015). Using these relationships heterogeneous network is constructed to infer the new candidates for drug repurposing (Hsih-Te Yang et al 2016).

2. Biomedical Relation Extraction

2.1. Overall Methodology

In this framework, MEDLINE database (Source: www.ncbi.nlm.nih.gov) containing a large number of research articles has been considered as the unlabeled corpus. PubTator, a web based tool is used to do NER which is the basic text processing for any type of relation extraction task. Later, sentences with more than one tagged entity were represented as a pattern with the help of dependency-tree feature. Bootstrapping starts with an initial seed set and iteratively learns new patterns by using entity and dependency –level masking techniques. The generated patterns are given scores to select the appropriate patterns for the next iteration.

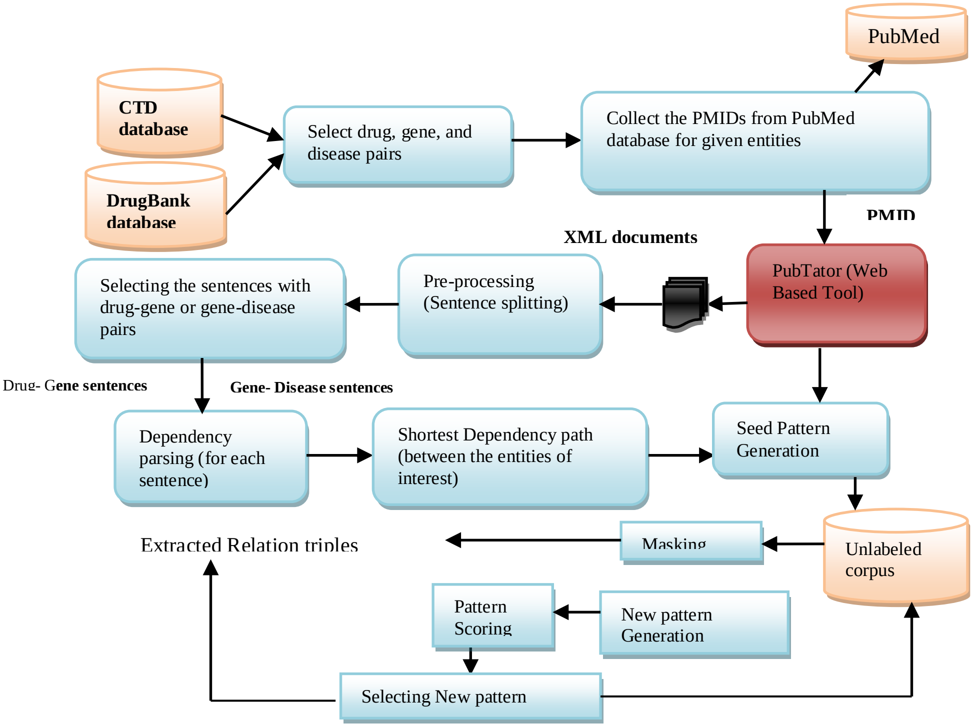

PubMed
CTD database
DrugBank database
Select drug, gene, and disease pairs
Collect the PMIDs from PubMed database for given entities
PMID
XML documents
PubTator (Web Based Tool)
Selecting the sentences with drug-gene or gene-disease pairs
Pre-processing (Sentence splitting)
Drug- Gene sentences
Gene- Disease sentences
Dependency parsing (for each sentence)
Shortest Dependency path (between the entities of interest)
Seed Pattern Generation
Extracted Relation triples
Masking
Unlabeled corpus
Pattern Scoring
New pattern Generation
Selecting New pattern

Figure 2.1 Diagram of pattern based bootstrapping algorithm

The extracted biomedical binary relations are stored in the form of triples i.e. {ENT1-I, TW, ENT-II}. ENT-I and ENT-II are the biomedical entities and TW is the trigger word which indicates the semantic relationship between the entities.

2.2. Pre-processing and Dependency Tree Parsing

The downloaded abstracts are split into sentences and the sentences which have both the drug and gene or gene and disease only are selected for generating the seed pattern. Sarafraz F (2013) and Cruz Dıaz N.P et al. (2015) discussed that most of the system does not consider the possibility of negative relationships that could lead to false positives in the literature. The proposed system will treat the negative sentences which can lead to false positives. De Marneffe MC (2006) discussed about dependency grammar which represents the sentences with a syntactic tree and analyzes the relationships between the words. In dependency grammar, usually verbs are perform as the root and other words are dependent on root words either directly or indirectly dependent on the root. Later he used natural language processor tool (http://nlp.stanford.edu/software/lex-parser.shtml) namely Stanford to generate the dependency tree of the sentence. The shortest dependency path between the entities of interest is extracted.

Example: CYP3A4 mRNA expression was significantly increased by rifampicin exposure in human hepatocytes.

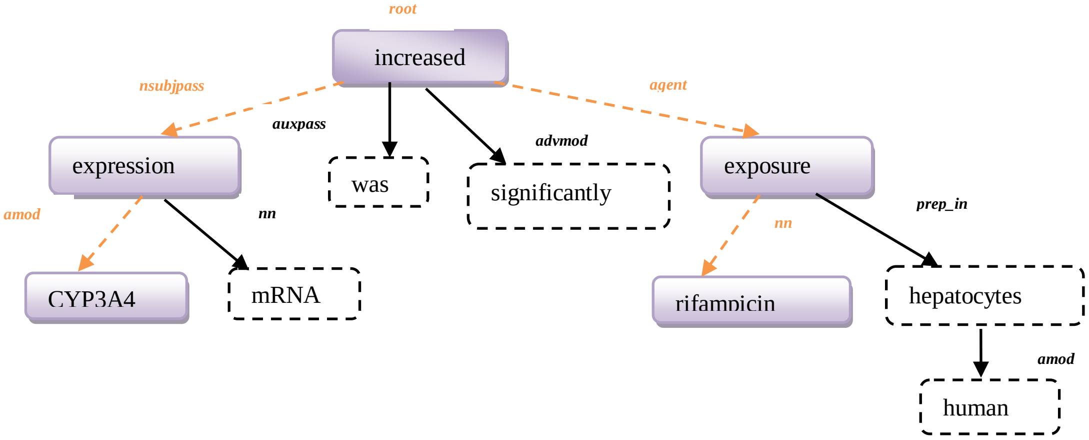


Figure 2.2 Dependency graph and shortest dependency path connecting the CYP3A4 and rifampicin

Shortest Dependency Path (SDP) will be generated by removing the irrelevant terms and phrases from the original sentence and focus on part of the sentence that are directly relevant to the relationship between the two entities as discussed by Yifan Peng (2015). Sometimes more than one dependency path can be generated for the same sentence when it has drug, gene and disease in the sentence.

Sentence: Gemfibrozil and the glucuronide inhibit CYP2C8 and OATP1B1. Consider this sentence and the relation have to be extracted between the following pairs of entities. (i) Gemfibrozil, CYP2C8, (ii) Gemfibrozil, OATP1B1, (iii) glucuronide, CYP2C8, (iv) glucuronide, OATP1B1. Here, a single sentence contains more than one relation.

2.3. Pattern Representation

For identifying the new patterns from the seed set, representing the patterns with features are the important step in bootstrapping procedure. As discussed in 2.2, Shortest path connecting the entities is taken from the dependency graph by neglecting the edge direction is used for representing the pattern and it gives compact representation for the sentences that are too long. (Bunescu & Mooney 2005). Figure 2.2 represents the dependency graph of the sentence and its shortest path is indicated in orange color between the biomedical entities. Figure 2.4 indicates the pattern representation for bootstrapping algorithm. Three components taken place in the pattern representation namely two biomedical entities (present within a sentence), the words in the shortest path and dependency relations connecting those words in the shortest path. According to the length of the dependency path between the entities, the pattern length varies in size. For a relation to happen, minimum path-length of five is needed (Bunescu & Mooney 2005). Entities connected through path length of less than five are not taken for consideration. Pattern formation of the sample sentence in 2.2 is given below.

CYP3A4	amod	**Expression**	nsubjpass	**increase**	agent	**exposure**	nn	rifampicin

Figure 2.3 Pattern derivation from the shortest dependency path- example. Red color, violet color text-entities, blue color text- dependency relation, black color text-word

Hereafter the pattern is termed as 5-window, 7-window and so on. The pattern of length 5 is denoted as 5-window pattern. For window of size five, the pattern consists of two biomedical entities, two dependency relations and a single word. For every increment in the pattern length one dependency relation and one word gets increased as shown in figure 2.4.

E1	DR1	W1	DR2	E2

5-window pattern

E1	DR1	W1	DR2	W2	DR3	E2

7-window pattern

E1, E2-entities
DR1, DR2.. – dependency relation
W1, W2..- words in SDP

Figure 2.4 Pattern Representation

More words are present in the higher order patterns. These higher order patterns use the lower order patterns to extract the relation between the biomedical entities. Once the representation of pattern is done, next step is to select the initial seed set.

2.4. Selection of Seed pattern

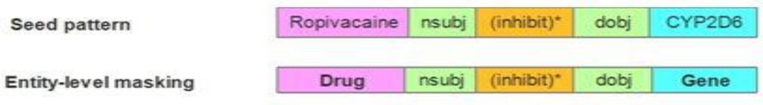
Seed pattern is needed for initializing the bootstrapping algorithm. The seed patterns contain a list of patterns and this list is chosen from the available EUADR corpus based on the frequency of occurrence. As the pattern length varies in size, a single seed pattern is chosen for each length in the seed set. Based on the number of relation types (drug-gene & gene-disease) and the varying pattern length for each type, the seed pattern count differs.

2.5. Masking

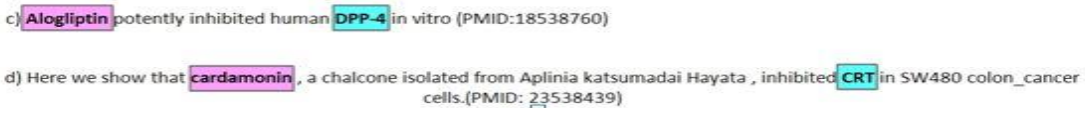
For identifying relations and trigger words from unlabeled corpus, first do the exact match with the seed patterns. Then for generating and identifying new patterns, bootstrapping algorithm masks the seed patterns. Here the entity-level and dependency-level masking is done for generating new patterns.

2.5.1. Entity based Masking

In this level, the exact entity names are masked and replaced with type of the entity For example, in Figure 2.5, the exact entity names 'Ropivacaine' and 'CYP2D6' are masked, and replaced with their corresponding entity type 'Drug' and 'Gene' respectively. This identifies new entity pairs which are expressed in the same way as the seed pattern with the same trigger word with the 5-window pattern. The entity-level masked pattern is used as the seed pattern for dependency-level masking.



Seed pattern	Ropivacaine	nsubj	(inhibit)*	dobj	CYP2D6
Entity-level masking	Drug	nsubj	(inhibit)*	dobj	Gene



c) Alogliptin potently inhibited human DPP-4 in vitro (PMID:18538760)

d) Here we show that cardamonin , a chalcone isolated from Aplinia katsumadai Hayata , inhibited CRT in SW480 colon_cancer cells.(PMID: 23538439)

Figure 2.5 Entity based Masking

2.5.2. Dependency Relation based Masking

Due to the variations in the sentence expression, dependency relations in the pattern also differed. Hence, the dependency path relations in the pattern are masked one at a time to generate new patterns. The new patterns derived out of masking the dependency relations ‘nsubj’ and ‘dobj’ in seed pattern along with examples are shown in Figure 2.6. Masking ‘nsubj’ produces new patterns with ‘nmod’

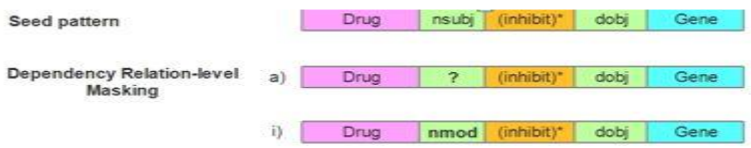


Figure 2.6 Dependency Relation based masking

2.6. Scoring of Pattern

Next step in the bootstrapping algorithm is to identify the candidate patterns by scoring the newly generated patterns. The dependency-level masking generates a large number of new patterns, but we are not able to use all the generated patterns in the next iteration as it decrease the performance of the system. Hence, we choose the patterns which are having high score and it is used in the next iteration. Scoring technique is based on the unique relation identified (support-based scoring) by the given pattern. The generated new pattern extracts new relation triples from the unlabeled corpus. Support based method calculates the score based on the unique relation triples identified by new pattern with respect to the seed. $S_{r,}$ is the support-based score is mentioned in equation (1).

$$S_r = \frac{\text{support}\{T_i\}}{\text{support}\{T_{seed}\}} \qquad (1)$$

T_i − relation triples identified by pattern i ∈ unlabelledcorpus

T_{seed} − relation triples identified by seed pattern ∈ unlabelledcorpus

3. Result and Discussion

3.1. Dataset Description

The bootstrapping framework learns new patterns from the unlabeled data. The unlabeled data is collected from the PubMed articles of April 2018 version of PubTator (Wei et al. 2013), which has approximately 21 million PubMed. PubTator annotations consists of title and abstract of PubMed articles. PubTator make use of the following tools to recognize the entities. GeneTUKit (Huang et al. 2011) and GenNorm (Wei & Kao 2011) for gene mentions, DNorm (Leaman et al. 2013) for diseases, a dictionary-based lookup technique (Davis et al. 2012) for chemicals. Seed pattern for the two types of relations (drug-target, target-disease) are taken from the EU-ADR (Van Mulligen et al (2012) corpus and it has 100 abstracts for each type. Comparative

Toxicogenomics Database (CTD) (Davis et al. 2017) contains the information about drug-target and target-disease relationships which is manually curated.

3.2. System Setup

Drug-target and target-disease are the two relations evaluated by the proposed framework. The sentences have at least two different types of entities (drug, gene, disease) are considered for the unlabeled data. Stanford dependency parser (Bunescu et al 2014) is applied to determine the dependency relations in the given sentences and the SDP between the entities of interest is extracted. If a single sentence has more than two entities, all the entities in combinations are taken into account. So, a single sentence can be applied many times for different relation between the entities. To avoid erroneous relation, the dependency parser gives the label as ‘dep’ for the words in which the exact relation it is not able to determine.

Table 3.1 Number of patterns identified in the unlabelled corpus for relation type each window size

Window-size	Drug-Gene	Gene-Disease
Five-window	5,28,626	6,23,316
Seven window	8,31,058	12,91,239
Nine-window	7,49,765	11,22,994

3.3. Estimation of Bootstrapping Framework for Relation Extraction

The bootstrapping framework learns new patterns in each iteration and in turn extracts biomedical relations and trigger words from the unlabeled text corpus. Here this proposed system is evaluated based on the ability to learn the new patterns. The total number of patterns learned by the framework (including all pattern length) for the drug-gene, gene-disease relations are 6367, 10404 respectively. Figure 3.1 shows that the number of patterns extracted by using 7-window size is high for drug-gene relation and 9-window size is high for gene-disease relation. It can be seen that the bootstrapping framework is able to learn a large number of new patterns from the unlabeled corpus, using only a minimum set of seed patterns. The number of patterns generated by using the proposed system is 6367, 10404 for drug-gene and gene-disease relation respectively. From this we infer that the proposed system learns higher number of patterns from gene-disease relation compared to the other one.

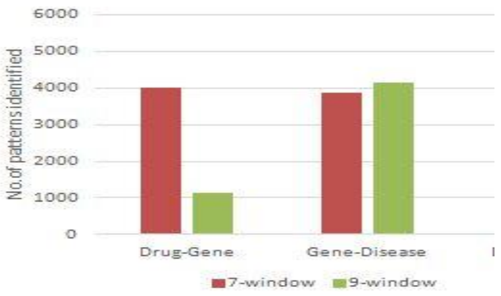


Figure 3.1. No. of patterns extracted by our method for different window-size

Table 3.2 provides the number of relations extracted by the bootstrapping framework along with the number of relations that have evidence in the CTD database. Comparatively drug-gene relation has less evidence as 35%, while gene-disease relation has more-evidence as greater than 60%, as number of inferred associations is high for gene disease in CTD.

Table 3.2 Performance of the proposed system

Relation pair count	Drug-Gene	Gene-Disease
No. of relations extracted by bootstrapping	50,105	1,21,576
No. of relations extracted by bootstrapping that have evidence in CTD database.	14,116	78,014

3.4. Relationship Identified by the Bootstrapping Framework

Table 3.3 provides the information about the number relationship identified in each relation type. In each relation type, the top five frequently occurred trigger word is provided in Table 3.4.

Table 3.3 Relationship identified by bootstrapping Framework

Relation -word count	Single
Drug-Gene	283
Gene-Disease	339

Table 3.4 Top-5 Relation words

Drug-Gene	Gene-Disease
inhibitor	expression
receptor	gene
activity	level
antagonist	mutation
phosphorylation	associate

3.5. Comparison with Existing State-of-the-art Method

Bravo et al. (2015) compared the existing supervised method with proposed semi-supervised pattern-based bootstrapping framework for the biomedical relation extraction task. The bootstrapping framework is compared with Befree based on Precision, Recall and F_1 score evaluation metrics and the results are provided in Table 3.5. Since, BeFree system was trained using EU-ADR corpus for the two relation types, the patterns learnt by the bootstrapping framework is used to identify the relation pairs in the gold standard dataset EU-ADR. For all the three considered relation types, bootstrapping achieves a higher F_1 score compared to the baseline approach.

Table 3.5 Comparison of Bootstrapping with existing state-of-the-art method

Association Type	Method	Precision	Recall	F_1 score
Drug-Target	Supervised	74.2	97.4	83.3
	Bootstrapping	86.1	83.9	84.36
Target-Disease	Supervised	75.1	91.8	82.4
	Bootstrapping	85.7	84.9	85.29

4. Conclusion and Future Work

In this system, an improved approach for relationship extraction between drug, gene and disease entities in the biomedical domain is proposed. This approach involves identification of new relation by giving some initial seeds to the bootstrapping method. The results prove that the direct relationships from the biomedical text have been extracted successfully. The proposed system was able to learn a large number of useful patterns (16,771) from a small seed set (6). These patterns in turn were able to identify 171,881 relation pairs with 644 trigger words that convey the semantics of the biomedical relation. And bootstrapping method attains approximately 85% of f-score for both types (drug-gene and gene-disease) which is better than supervised method. Out of the identified relations more than 50% had evidence in the CTD database. By using the drug-gene and gene-disease direct relationships, we cannot infer more number of hidden relations for identifying repurposing drugs. So pattern based bootstrapping method can be performed for other biomedical relation types (like drug-disease, drug-drug, drug-adverse effect and so on) to automatically extract all the biomedical relations from the unlabeled text corpus (PubMed) to get more number of repurposing drugs.

The proposed method will be extracting the relation between the entities within the sentences. It will not be effective for the entities across the sentences. In the future work, the above method can be extended to extract the relation between the biomedical entities across the sentence.

REFERENCES

[1] Cummings, J., 2021. New approaches to symptomatic treatments for Alzheimer’s disease. *Molecular Neurodegeneration*, *16*(1), pp.1-13.

[2] Hwang, Thomas J., Daniel Carpenter, Julie C. Lauffenburger, Bo Wang, Jessica M. Franklin, and Aaron S. Kesselheim, (2016) "Failure of investigational drugs in late-stage clinical development and publication of trial results." *JAMA internal medicine* 176, no. 12: 1826-1833.

[3] Tobinick, Edward L, (2015) "The value of drug repositioning in the current pharmaceutical market." *Drug News Perspect* 22, no. 2: 119-125.

[4] Talevi, A. and Bellera, C.L., 2020. Challenges and opportunities with drug repurposing: finding strategies to find alternative uses of therapeutics. *Expert opinion on drug discovery*, *15*(4), pp.397-401.

[5] Ashburn, Ted T., and Karl B. Thor, (2015) "Drug repositioning: identifying and developing new uses for existing drugs." *Nature reviews Drug discovery* 3, no. 8: 673-683.

[6] Rudrapal, M., Khairnar, S.J. and Jadhav, A.G., 2020. Drug Repurposing (DR): An Emerging Approach in Drug Discovery. In *Drug Repurposing-Hypothesis, Molecular Aspects and Therapeutic Applications*. IntechOpen.

[7] Shahab, Elham, (2017) "A short survey of biomedical relation extraction techniques." *arXiv preprint arXiv:1707.05850*.

[8] Zhao, Z., Yang, Z., Sun, C., Wang, L. and Lin, H., 2017, November. A hybrid protein-protein interaction triple extraction method for biomedical literature. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (pp. 1515-1521). IEEE.

[9] Xu, Rong, and QuanQiu Wang, (2014) "Toward creation of a cancer drug toxicity knowledge base: automatically extracting cancer drug—side effect relationships from the literature, " *Journal of the American Medical Informatics Association* 21, no. 1: 90-96.

[10] Zhang, Yijia, Hongfei Lin, Zhihao Yang, Jian Wang, and Yanpeng Li, (2012) "A single kernel-based approach to extract drug-drug interactions from biomedical literature." *PLoS One* 7, no. 11: e48901.

[11] Bravo, Àlex, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I. Furlong, (2015) "Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research." *BMC bioinformatics* 16, no. 1: 55.

[12] Madkour, Amgad, Kareem Darwish, Hany Hassan, Ahmed Hassan, and Ossama Emam, (2007) "BioNoculars: extracting protein-protein interactions from biomedical text." In *Biological, translational, and clinical language processing*, pp. 89-96.

[13] Erkan, Gunes, Arzucan Özgür, and Dragomir Radev, (2007) "Semi-supervised classification for extracting protein interaction sentences using dependency parsing." In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 228-237.

[14] Batista, David S., Bruno Martins, and Mário J. Silva, (2015) "Semi-supervised bootstrapping of relationship extractors with distributional semantics." In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 499-504.

[15] Yang, Hsih-Te, Jiun-Huang Ju, Yue-Ting Wong, Ilya Shmulevich, and Jung-Hsien Chiang, (2016) "Literature-based discovery of new candidates for drug repurposing." *Briefings in bioinformatics* 18, no. 3: 488-497.

[16] Sarafraz, Farzaneh, and Goran Nenadic, (2010) "Using SVMs with the command relation features to identify negated events in biomedical literature." In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pp. 78-85.

[17] De Marneffe, Marie-Catherine, Bill MacCartney, and Christopher D. Manning, (2006) "Generating typed dependency parses from phrase structure parses." In *Lrec*, vol. 6, pp. 449-454.

[18] Peng. Yifan, Samir Gupta, Cathy Wu, and K. Vijay-Shanker, (2015) "An extended dependency graph for relation extraction in biomedical texts." In *Proceedings of BioNLP 15*, pp. 21-30.

[19] Bunescu, Razvan, and Raymond Mooney, (2005) "A shortest path dependency kernel for relation extraction." In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 724-731.

[20] Wei, Chih-Hsuan, Hung-Yu Kao, and Zhiyong Lu, (2013) "PubTator: a web-based text mining tool for assisting biocuration." *Nucleic acids research* 41, no. W1: W518-W522 (2013)

[21] Huang, Minlie, Jingchen Liu, and Xiaoyan Zhu, (2011) "GeneTUKit: a software for document-level gene normalization." *Bioinformatics* 27, no. 7: 1032-1033.

[22] Wei, Chih-Hsuan, and Hung-Yu Kao, (2011) "Cross-species gene normalization by species inference." *BMC bioinformatics* 12, no. S8: S5.

[23] Leaman, Robert, Rezarta Islamaj Doğan, and Zhiyong Lu, (2013) "DNorm: disease name normalization with pairwise learning to rank." *Bioinformatics* 29, no. 22: 2909-2917.

[24] Van Mulligen, Erik M., Annie Fourrier-Reglat, David Gurwitz, Mariam Molokhia, Ainhoa Nieto, Gianluca Trifiro, Jan A. Kors, and Laura I. Furlong, (2012) "The EU-ADR corpus: annotated drugs, diseases, targets, and their relationships." *Journal of biomedical informatics* 45, no. 5: 879-884.

[25] Davis, Allan Peter, Cynthia J. Grondin, Robin J. Johnson, Daniela Sciaky, Benjamin L. King, Roy McMorran, Jolene Wiegers, Thomas C. Wiegers, and Carolyn J. Mattingly, "The comparative toxicogenomics database: update 2017." *Nucleic acids research* 45,no. D1: D972-D978.

AUTHORS

Saranya M is a research scholar in Anna University, College of Engineering, Guindy campus, Tamil Nadu, India. She received a bachelor’s degree and master’s degree in CSE from Anna University, Chennai , Tamil Nadu, India. She is currently interested in doing the research in Text Mining, NLP, data mining, AI etc.

Arockia Xavier Annie R is an Assistant Professor in Anna University, College of Engineering, Guindy campus, Tamil Nadu, India. She received a bachelor’s degree and master’s degree in CSE from Anna University, Chennai, Tamil Nadu, India. Her interest is Text Mining, Multimedia, Networks, Compilers, Video Technology, AI etc.

Geetha T V is a UGC-BSR Faculty Fellow, Retired Senior Professor and former Dean of CEG campus in Anna University. She received a bachelor’s degree in ECE and Master’s degree in CSE from Anna University, Chennai, Tamil Nadu, India. She is guiding many students from last 20 years ago. Her interest is NLP, Web Search, Social Network Analysis, Text Mining, AI etc.

EXTRACTION OF LINGUISTIC SPEECH PATTERNS OF JAPANESE FICTIONAL CHARACTERS USING SUBWORD UNITS

Mika Kishino[1] and Kanako Komiya[2]

[1]Ibaraki University, Ibaraki, Japan
[2]Tokyo University of Agriculture and Technology, Tokyo, Japan

ABSTRACT

This study extracted and analyzed the linguistic speech patterns that characterize Japanese anime or game characters. Conventional morphological analyzers, such as MeCab, segment words with high performance, but they are unable to segment broken expressions or utterance endings that are not listed in the dictionary, which often appears in lines of anime or game characters. To overcome this challenge, we propose segmenting lines of Japanese anime or game characters using subword units that were proposed mainly for deep learning, and extracting frequently occurring strings to obtain expressions that characterize their utterances. We analyzed the subword units weighted by TF/IDF according to gender, age, and each anime character and show that they are linguistic speech patterns that are specific for each feature. Additionally, a classification experiment shows that the model with subword units outperformed that with the conventional method.

KEYWORDS

Pattern extraction, Characterization of fictional characters, Subword units, Linguistic speech patterns, word segmentation

1. INTRODUCTION

There is research in the field of natural language processing that focuses on linguistic styles and characterizes utterances of confined groups categorized by some features like gender or age. Japanese is a language whose expressions vary depending on gender, age, and relationships with dialog partners. In particular, Japanese anime and game characters sometimes speak with emphasis on character rather than reality. Furthermore, the way of talking of Japanese fictional characters is sometimes different from real people. For example, Funassyi, a Japanese mascot character, usually ends each utterance with "なっしー, nassyi" yet this ending is not found in a Japanese dictionary. Additionally, a cat character tends to add "にゃん, nyan", an onomatopoeia that expresses a cry of a cat at the end of each utterance. Human characters also have character-specific linguistic speech patterns in novels, anime, and games. They are known as role language [1] and it is related to characterization; the role language shows what role the speaker plays, and sometimes it is different from real conversation. For example, "僕, boku, I" is a first-person singular usually used for boys in novels, anime, and games, but it is also used for men and boys in real life. Therefore, in this study, we extracted and analyzed the linguistic speech patterns that characterize these characters using utterances of anime or game characters. In Japanese, morphological analysis is a basic technology for natural language processing because Japanese does not have word delimiters between words. Word segmentation and morphological analysis

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 85-96, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112307

are now widely performed using morphological analyzers like MeCab and Chasen and their performances are usually very high level. However, they are unable to segment broken expressions or the endings of utterances that are not found in the dictionary, which often appears in lines of anime or game characters (refer to Section 2). To hinder this problem, we propose using subword units to segment lines of Japanese anime or game characters and extracting strings that occur frequently (refer to Section 3). The subword units are usually used with deep learning technologies and their robustness for out-of-vocabulary words is often noted. However, they are less interpretable than the original words because the segmentation are depending on the frequencies or occurrence probabilities rather than the meanings. In the current study, however, we show that the expressions extracted using subword units are more interpretable than those using the original words for the extractions of linguistic speech patterns of fictional characters, which is the case where many words are not listed in the dictionary using data collected from publications on the internet (refer to Section 4). We also show that the subword units are effective even though no deep learning technology is used with them. In the experiment, we weighted the subword units by TF/IDF according to gender, age, and each anime character (refer to Sections 5) and show that they are linguistic speech patterns that are specific for each feature (refer to Sections 7 and 8). Additionally, we performed a classification experiment using a support vector machine (SVM) based on linguistic speech patterns we extracted to classify the characters into a character group (refer to Sections 6) and showed that a subword unit model outperformed a conventional morphological analyzer (refer to Sections 7 and 8). Finally, we conclude our work in Section 9.

2. RELATED WORK

Japanese does not have word delimiters between words and word boundaries in Japanese are unspecific. Therefore, there has been much research on Japanese word segmentation or morphological analysis and there are many morphological analyzers for Japanese texts like MeCab [2], Chasen, Juman++ [3], and KyTea [4], These morphological analyzers segment words with high performances but sometimes the performances decrease for the noisy texts. For Japanese word segmentation of noisy texts, Sasano et al. [5] proposed a simple approach to unknown word processing, including unknown onomatopoeia in Japanese morphological analysis. Saito et al. [6] also recommend using character-level and word-level normalization to address the morphological analysis of noisy Japanese texts. Recently, algorithms for subword unis such as Byte Pair Encoding (BPE) [7] and unigram language model [8] are proposed. They are mainly proposed for neural machine translation and usually used with deep learning technologies. We used the unigram language model for word segmentation of Japanese lines of fictional characters. There are some studies on interpretability and usability of words depending on the word segmentation for information retrieval (IR). Kwok [9] investigated and compared 1-gram, bigram, and short-word indexing for IR. Nie et al. [10] proposed the longest-matching algorithm with single characters for Chinese word segmentation for IR. In addition, there has been much research on characterization. PERSONAGE (personality generator) developed by Mairesse and Walker [11] as, the first highly parametrizable conversational language generator. They produced recognizable linguistic variation and personality, and our work also focused on each character's personality. Walker et al. [12] reported a corpus of film dialog collected and annotated for linguistic structures and character archetypes. Additionally, they conducted experiments on their character models to classify linguistic styles depending on groups such as genre, gender, directors, and film period. Miyazaki et al. [13] conducted a fundamental analysis of Japanese linguistic expressions that characterize speeches for developing a technology to characterize conversations by partially paraphrasing them. In their subsequent research, Miyazaki et al. [14] reported categories of linguistic peculiarities of Japanese fictional characters. Miyazaki et al. [15] conducted an experiment to see whether the reader can understand the characterization of a dialog agent by paraphrasing the functional part of each sentence with a probability suitable for the target character, as a way to characterize the speech and to enrich the variation of the speeches. Another study focused on Japanese sound change expressions to characterize speeches of Japanese fictional characters; they collected these expressions and classified them [16]. Additionally, Okui and Nakatsuji [17] used a pointer generating mechanism to generate various responses for a Japanese dialog system, referring to several different character responses. They learned the characterization of the responses with a small amount of data.

3. EXTRACTION OF LINGUISTIC SPEECH PATTERNS USING SUBWORD UNITS

Many terms not included in the dictionary such as expressions with characterization at the endings of utterances and broken expressions appear in fictional character dialogs. As a result, using existing morphological analyzers with dictionaries to segment the lines of fictional characters are challenging. Therefore, we propose using subword units for the segmentation of lines of fictional characters. The concept behind subword units is that the frequency of occurrence of a word is studied in advance, and low-frequency words are broken down into letters and smaller words. In other words, using subword units, we can treat a string with a high frequency of occurrence as a single unit, not a word in a dictionary. We used software referred to SentencePiece [18] for word segmentation of Japanese lines of fictional characters. SentencePiece learns the segment method directly from the text and segments the text into subword units. It supports BPE and unigram language model, but we employed unigram language model because it slightly outperformed BPE when they were used for machine translation.

3.1. Unigram Language Model

We explain the algorithm of unigram language model quoting from [8]. The unigram language model makes an assumption that each subword occurs independently, and consequently, the probability of a subword sequence $X = (x_m, \ldots, x_m)$ is formulated as the product of the subword occurrence probabilities $p(x_i)$. The most probable segmentation X^* for the input sentence X is obtained with the Viterbi algorithm. Because the vocabulary set V is unknown, they seek to find them with the following iterative algorithm.

1. Heuristically make a reasonably big seed vocabulary V from the training corpus.
2. Repeat the following steps until |V| reaches a desired vocabulary size.

   (a) Fixing the set of vocabulary, optimize $p(x)$ with the EM algorithm.
   (b) Compute the $loss_i$ for each subword x_i, where $loss_i$ represents how likely the likelihood is reduced when the subword x_i is removed from the current vocabulary.
   (c) Sort the symbols by $loss_i$ and keep top η % of subwords.

Unigram language model is a method whose objective function is maximization of log likelihood of X.

3.2. Procedures

We extracted linguistic speech patterns that characterize the lines as follows:

1. Collect lines of fictional characters,
2. Segment the lines into subword units using Sentence Piece, and
3. Weighted the subword units using TF/IDF values and obtain the top ten subword units.

In addition to the extraction experiments, we conducted classification experiments of characters. Finally, we compared the results of the method using SentencePiece with that of one of the de facto standard morphological analyzers for Japanese, MeCab. We used ipadic for Japanese dictionary of MeCab.

4. DATA

We collected dialogs of 103 characters from 20 publications on the internet. They are, Anohana: The Flower We Saw That Day, Den-noh Coil, Dragon Quest IV-VIII, Neon Genesis Evangelion, Mobile Suit Gundam, Howl's Moving Castle, Hyouka, Kaguya-sama: Love Is War, Kemono Friends, Harem Days, Whisper of the Heart, Laputa: Castle in the Sky, Spirited Away, Symphogear, My Neighbor Totoro, and The Promised Neverland. This corpus of dialogs is referred as to the “Character Corpus.” The following three methods were used for the collection.

1. They were collected from a compilation site of anime and game dialog on the internet.
2. They were collected from anime video sites.
3. It was converted from manga e-books using a text detection application.

Priority was given to characters with many lines while choosing character in the work. Furthermore, since it was assumed that the majority of the main characters would be mostly classified as boys, girls or younger men or women, we aggressively collected child and older characters with a significant number of dialogs during the selection process. Because we have classification experiments according to age, characters whose ages change drastically during the story have been removed. An example of this is Sophie from Howl's Moving Castle. She changed from 18 to 90 years old in the movie. We also eliminated characters with extremely low amounts of dialog. The minimum, maximum and average numbers of lines of a character are respectively 92, 6,797, and 1,187.17.

5. EXPERIMENTS OF LINGUISTIC SPEECH PATTERN EXTRACTION

The procedure of linguistic speech pattern extraction by SentencePiece is as follows. First, we develop a segmentation model by applying SentencePiece to each character's dialog. Notably, we apply SentencePiece to sub-corpus of each character rather than the entire corpus. This is because the way of talking varies according to each character. The following formula calculates the maximum number of subword units:

$$\text{Vocabulary\_Size} = \text{Basic\_VS} * \left(\frac{L}{l}\right)^{-\frac{1}{5}} \qquad (1)$$

Where, l denotes the number of letters of each character's lines, L denotes the total number of letters of lines of all characters, and Basic_VS denotes basic vocabulary size. We set Basic_VS to 3,000. Simultaneously with the creation of the model, a word list from the vocab file was also constructed. We delete from the word list subword units that consist of a single Chinese character except for the first-person singular (僕, 私, 俺) because we believed that they would not express a characterization. We also deleted 1/5 of the subword units with less emission logarithmic establishment, which is a measure of a subword unit’s occurrence probability. For the next step, we segment the character corpus using the segmentation model we created. The word lists and segmented character corpus were used to obtain the TF/IDF value, which was calculated using the following formula

$$\mathrm{tf}(t, d)= \frac{n_{(t,\ d)}}{\sum_{s\epsilon d} n_{(s,\ d)}} \tag{2}$$

where, tf(t, d) denotes the term frequency of a subword unit t in document d, $n_{(t,\ d)}$ denotes the number of occurrences of a subword unit t in document d, $\sum_{s\epsilon d} n_{(s,\ d)}$ denotes the sum of the number of occurrences of all subword units in the document d.

$$\mathrm{idf}(t)=\log\frac{N}{\mathrm{df}(t)} \tag{3}$$

where, idf(t) denotes the inverse document frequency of a subword unit t, N denotes total number of documents, df(t) denotes number of documents in which a subword unit t occurred.

$$\mathrm{TF/IDF} = \mathrm{tf}(t, d) * \mathrm{idf}(t) \tag{4}$$

We extracted linguistic speech patterns that characterize lines of gender, ages, and characters using TF/IDF value. We considered the lines of all characters of one gender as one document, and the lines of all character of the opposite gender as another document when calculating the TF/IDF value for a gender.

6. Classification Experiment

We performed a classification experiment to evaluate the extracted linguistic speech patterns using a SVM. The obtained TF/IDF values were used as inputs to the SVM to classify the characters into groups categorized by gender and age. The characters were first divided into three categories: children, adults, and seniors. Children and adults were further divided into two categories: male and female whereas seniors have only one group because we had few characters of the ages. As a result, we used five groups: boys, girls, men, women, and seniors. The numbers of the character according to the group are shown in Table1. The group classification was performed based on the character's characterization and not on their actual age or gender because the profiles of fictional characters are sometimes extraordinary. The bias in the amount of data for each category is affected by the bias in characters; Japanese anime and games we collected have a few children and senior characters. The experiment was conducted using five-fold cross-validation. Sklearn was used as a libraryin this experiment. The computational complexity of SVM in sklearn varies between O (number of dimensionality * number of data ^2) and O (number of dimensionality * number of samples ^3), depending on how efficiently the cache is used.

Table 1. Amount of data for Classification Experiment

Boys	**Girls**	**Men**	**Women**	**Senior**
6	8	40	41	7

7. Results

The linguistic speech patterns with the top 10 TF/IDF values are shown in Tables 2-7. In the tables, E represents ending, and F means first-person singular. Tables 2 and 3 list the linguistic speech patterns with gender characterization, and Tables 4 and 5 show those with age characterization. In these tables, Italic means that the pattern is specific for each characterization of fictional characters. Some of the character-specific linguistic speech patterns are also shown as example results in Tables 6 and 7. The example characters are Emma from the anime "The Promised Neverland," Shinji from the anime "Neon Genesis Evangelion," and Yangus from the game “Dragon Quest VIII.” For the experiment of characters, we had a questionnaire to evaluate the linguistic speech patterns. Eight native Japanese speakers were asked if each linguistic speech pattern seems specific for the character. Five people are men, and three were women, and seven people are in their 20's, and one person is in her 30's. They were also asked if they knew each anime or game that the character appears. Tables 8 and 9 summarize the results of the questionnaire. Finally, the results of the classification experiment are shown in Table 10.

Table 2. Linguistic Speech Patterns with Gender Characterization Retrieved by SentencePiece. E represents ending and F denotes the first-person singular. Italic means that the pattern is specific for the people of specific ages as lines of fictional characters.

Male			**Female**		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
ですね	desune	Polite E	わね	wane	*Feminine E*
でござる	degozaru	*Samurai E*	かしら	kashira	*Feminine E*
だぜ	daze	*Masculine E*	のかしら	nokashira	*Feminine E*
でござるな	degozaruna	*Samurai E*	だわ	dawa	*Feminine E*
アルス	Arusu	Name	よね	yone	*E*
だな	dana	*Masculine E*	のね	none	*Feminine E*
なあ	naa	*Old buddy*	ないわ	naiwa	*Feminine E*
ますね	masune	Polite E	わよ	wayo	*Feminine E*
でがすよ	degasuyo	*Dialect E*	ないわね	naiwane	*Feminine E*
でござるよ	degozaruyo	*Samurai E*	アルス	Arusu	Name

Table 3. Linguistic Speech Patterns with Gender Characterization Retrieved by MeCab. E represents ending and F denotes the first-person singular. Italic means that the pattern is specific for the people of specific ages as lines of fictional characters.

Male			**Female**		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
ござる	gozaru	*Samurai E*	あたし	atashi	*F for girls*
ざる	zaru	Error	かしら	kashira	*Feminine E*
俺	ore	*F for male*	アルス	Arusu	Name
アルス	Arusu	Name	私	watashi	*F*
オイラ	oira	*F for boys*	・	.	Mark
げす	gesu	*Dialect E*	しら	shira	Error
・	.	Mark	リュカ	Ryuka	Name
僕	boku	*F for boys*	たし	tashi	Error
ウィル	Will	Name	ましょ	masyo	*Femminine E*
俺	ore	*F for male*	ウィル	Will	Name

Table 4. Linguistic Speech Patterns with Age Characterization Retrieved by SentencePiece. E represents ending and F denotes the first-person singular. Italic means that the pattern is specific for the people of specific ages as lines of fictional characters.

Children			Adults			Seniors		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
なあ	naa	Old buddy	ですね	deshune	Polite E	でござる	degozaru	*Samurai E*
アルス	Arusu	Name	わね	wane	*Feminine E*	でござるな	degozaruna	*Samurai E*
お父さん	otosan	*Dad*	これ	kore	This	でござるよ	degozaruyo	*Samurai E*
オイラ	oira	*F for boys*	です	deshu	Polite E	でござるか	degozaruka	*Samurai E*
だよ	dayo	E	だな	dana	Masculine E	アルス殿	Arushudono	Sir Arusu
いっぱい	ippai	*Many*	かしら	kashira	*Feminine E*	殿	dono	Sir
だぞ	dazo	*Boyish E*	なんて	nante	Exclamatory how	るでござるよ	rudegozaruyo	*Samurai E*
てる	teru	E	アルス	Arusu	Name	とは	towa	C.f. with
るの	runo	*Feminine E*	どこ	doko	Where	るでござる	rudegozaru	*Samurai E*
だね	dane	E	さん	san	*title*	わし	washi	*F for old men*

Table 5. Linguistic Speech Patterns with Age Characterization Retrieved by MeCab. E represents ending and F denotes the first-person singular. Italic means that the pattern is specific for the people of specific ages as lines of fictional characters

Children			Adults			Seniors		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
オイラ	oira	*F for boys*	俺	ore	*F for male*	ござろ	gozaro	*Samurai E*
僕	boku	*F for boys*	ウィル	Will	Name	ござっ	goza	*Samurai suffix*
おっちゃん	ochan	*Pops*	リュカ	Ryuka	Name	など	nado	Such as
ゃっ	ya	Error	げす	gesu	Dialect	フム	humu	*Hm-hum*
ちゃっ	cha	Error	アムロ	Amuro	Name	うむ	umu	*Hmmm*
ちゃう	chau	*End up -ing*	ひすい	hisui	Name	やはり	yahari	As expected
オラ	ora	*F for boys*	ゃっ	ya	Error	サントハイム	santohaimu	Name
うわ	uwa	*Wow*	ちゃっ	cha	Error	いかん	ikan	*No for old men*
じんた	jinta	Wrror	アニキ	aniki	Bro	むう	muu	*Hmmm*
オッチャン	ochan	*Pops*	ドルマゲス	Dhoulmagus	name	ふむ	humu	*Hm-hum*

Table 6. Character-specific Linguistic Speech Patterns Retrieved by SentencePiece. E represents ending and F denotes the first-person singular.

Emma			**Shinji**			**Yangus**		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
てる	teru	E	ですか	desuka	Polite E	でがすよ	degasuyo	Dialect
にも	nimo	And	ミサトさん	Misatosan	Name with title	でがす	degasu	Dialect
ってこと	ttekoto	That means	僕は	bokuha	I am (F for men)	でげす	degesu	Dialect
ちょ	cho	Wait	ないよ	naiyo	there isn't	でげすよ	degesuyo	Dialect
いいよ	iiyo	OK	父さん	tosan	Dad	でがすね	degasune	Dialect
嫌だ	iyada	No	るんだ	runda	E	おっさん	ossan	Pops
の手	note	Hand of	僕	boku	F for men	かい	kai	E
私たちの	watashitachino	Our	だよ	dayo	E	んでがす	ndegasu	Dialect
信じ	shinji	Believe	綾波	Ayanami	Name	アッシは	asshiha	I am for men
もし	moshi	If	んですか	ndesuka	E for question	アッシら	asshira	We for men

Table 7. Character-specific Linguistic Speech Patterns Retrieved by MeCab. E represents ending and F denotes the first-person singular.

Emma			**Shinji**			**Yangus**		
Patterns	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**	**Patterns**	**Sounds**	**Notes**
私	watashi	F	僕	boku	F for men	げす	gesu	Dialect
レイ	Rei	Name	ミ	mi	Error	がす	gasu	Dialect
ノーマン	Noman	Name	サト	sato	Error	アッ	a	Error
マン	man	Error	父さん	tosan	Dad	アッシ	asshi	F for men
思う	omou	Think	さん	san	Title	すね	sune	E
うん	un	Yes	うわ	uwa	Wow	すか	suka	E for question
近寄っ	chikayo	Draw near	スカ	suka	Error	やしょ	yasho	Dialect
折れ	ore	Be folded	アスカ	Asuka	Name	おっさん	ossan	Pops
寄っ	yo	Draw near	トウジ	Touji	Name	姉ちゃん	nechan	Sis
そっ	so	Error	僕ら	bokura	We for men	ダンナ	danna	Master

Table 8. Number of People Who Think the Linguistic Speech Pattern Extracted by SentencePiece is specific for the Character and Its Percentages. W/ represents with knowledge of the anime or game and w/o indicates without knowledge. People represents number of people with and without knowledge of the anime or game.

Emma			Shinji			Yangus		
Patterns	**w/**	**w/o**	**Patterns**	**w/**	**w/o**	**Patterns**	**w/**	**w/o**
People	**4**	**4**	**People**	**7**	**1**	**People**	**3**	**5**
てる	0	0	ですか	0	0	でがすよ	3	4
にも	0	0	ミサトさん	1	0	でがす	3	4
ってこと	0	0	僕は	3	1	でげす	3	4
ちょ	0	0	ないよ	0	0	でげすよ	3	4
いいよ	1	0	父さん	3	1	でがすね	3	4
嫌だ	3	0	るんだ	1	1	おっさん	1	2
の手	0	0	僕	2	1	かい	1	2
私たちの	4	1	だよ	0	0	んでがす	3	4
信じ	3	0	綾波	4	0	アッシは	3	3
もし	0	0	んですか	0	0	アッシら	3	3
Total	11	1	Total	14	4	Total	26	34
Percent	27.50%	2.50%	Percent	20.00%	40.00%	Percent	86.67%	68.00%
Avarage	15.00%		Avarage	22.50%		Avarage	75.00%	

Table 9. Number of People Who Think the Linguistic Speech Pattern Extracted by MeCab is specific for the Character and Its Percentages. W/ represents with knowledge of the anime or game and w/o indicates without knowledge. People represents number of people with and without knowledge of the anime or game.

Emma			Shinji			Yangus		
Patterns	**w/**	**w/o**	**Patterns**	**w/**	**w/o**	**Patterns**	**w/**	**w/o**
People	**4**	**4**	**People**	**7**	**1**	**People**	**3**	**5**
私	0	0	僕	2	1	げす	3	4
レイ	1	0	ミ	0	0	がす	3	4
ノーマン	1	0	サト	0	0	アッ	0	0
マン	0	0	父さん	3	1	アッシ	3	3
思う	0	0	さん	0	0	すね	2	1
うん	0	0	うわ	0	0	すか	2	1
近寄っ	0	0	スカ	0	0	やしょ	2	1
折れ	0	0	アスカ	4	0	おっさん	1	2
寄っ	0	0	トウジ	0	0	姉ちゃん	0	3
そっ	0	0	僕ら	1	1	ダンナ	1	3
Total	2	0	Total	10	3	Total	17	22
Percent	5.00%	0.00%	Percent	14.29%	30.00%	Percent	56.67%	44.00%
Avarage	2.50%		Avarage	16.25%		Avarage	48.75%	

Table 10. Results of Classification Experiment

SentencePiece	**MeCab**
0.627	0.451

8. DISCUSSION

Tables 2-7 shows that regardless of whether the SentencePiece or MeCab model is used, many endings of utterances and first-person singulars are extracted as specific linguistic speech patterns. We believe that they substantially characterize Japanese dialog. Many personal names are also extracted, although they are not linguistic speech patterns, because they often appeared in the lines of characters. MeCab found 13 error expressions whereas the SentencePiece model found none. Here, an error means the expression has no meaning due to a segmentation error. This result indicates that a conventional morphological analyzer sometimes fails to segment unusual sentences such as lines of fictional characters. Furthermore, we can observe from the tables that the SentencePiece model can obtain linguistic speech patterns that consist of many words. For example, “desune” consists of “desu” and “ne” and “wane” consists of “wa” and “ne.” The SentencePiece model could retrieve these linguistic speech patterns because it used subword units.

Furthermore, the SentencePiece model retrieved seven masculine and nine feminine linguistic speech patterns for the gender experiment, whereas MeCab retrieved six masculine and four feminine linguistic speech patterns. For age experiment, the SentencePiece model obtained six, two, and seven linguistic speech patterns that are children, adults, and seniors, respectively, whereas MeCab retrieved six, one, seven linguistic speech patterns. For the experiment of ages, the difference between the two models was smaller than that of gender. We believe that the systems could not extract linguistic speech patterns specific for adults because their talking way is considered normal.

Next, let us discuss the experiment of each character. This is more difficult than the discussion of gender or age because the knowledge of the character can affect the results. Therefore, we had a questionnaire for eight people. Tables 8 and 9 show that the SentencePiece model always obtains more character-specific linguistic speech patterns than MeCab for every character. The knowledge of the characters did not affect this result. However, the people with knowledge considerably feel that the linguistic speech patterns are specific for Emma, but the people without knowledge feel they are not so much. According to English Wikipedia, “The bright and cheerful Emma is an 11-year-old orphan living in Grace Field House, a self-contained orphanage housing her and 37 other orphans.” We believe that people without knowledge tend to think she is an adult woman because Emma is a female name. People without knowledge could think that the extracted patterns are not character-specific because they include no feminine patterns. Additionally, according to the Dragon Quest Wiki, “Yangus is a character in Dragon Quest VIII who accompanies the Hero on his missions.” and “He serves as a powerful tank character over the course of the game.” As for Yangus, the people with knowledge feel that the linguistic speech patterns are more character-specific again. Moreover, the percentage where people think they are character-specific is the highest among the three characters. We believe that this is because Yangus speak a dialect-originated and specific language. According to English Wikipedia, “Shinji is a dependent and introverted boy with few friends, reluctant unable to communicate with other people and frightened by contact with strangers.” Although there could be a bias because only one person did not know him, people who did not know felt that the extracted patterns were more character-specific. These results indicate that the extracted expressions using subword units are more interpretable linguistic speech patterns than those using words. The classification results also showed that the SentencePiece model outperformed MeCab for the classification of character groups. Additionally, it indicates that the patterns are more specific for each character group feature. Notably, the subword units are proposed for deep learning technologies but our classification did not use any of them. The experiments showed that the subword units are effective when no deep learning technologies are used.

9. CONCLUSIONS

In this study, we proposed using subword units to segment dialogs of fictional characters. The experiments revealed that subword units weighted with TF/IDF values are character-specific linguistic speech patterns, that cannot be obtained from existing morphological analyzers using dictionaries. They also showed that the linguistic speech patterns retrieved using SentencePiece are more specific for gender, age, and each character. It indicates that the extracted expressions using subword units are more interpretable than those using words. We also conducted an experiment that classifies the characters into character groups using the extracted linguistic speech patterns as features, and the classification SentencePiece model's accuracy was compared to the case where MeCab was used to segment the dialogs. We showed that subword units are effective even though no deep learning technologies are used with them. In the future, we would like to consider parts of speech when segment terms. Also, we are interested in research to generate sentences with characterization using the linguistic speech patterns extracted in this study.

REFERENCES

[1] Satoshi Kinsui, (2017) Virtual Japanese : Enigmas of Role Language, Osaka University Press.

[2] Taku Kudo and Kaoru Yamamoto and Yuji Matsumoto, (2004) Applying Conditional Random Fields to Japanese Morphological Analysis, the Proceedings of EMNLP 2004, pp230-237.

[3] Hajime Morita and Daisuke Kawahara and SadaoKurohashi, (2015) Morphological Analysis for Unsegmented Languages using Recurrent Neural Network Language Model, the Proceedings of EMNLP 2015, pp 2292-2297.

[4] Graham Neubig and Yosuke Nakata and Shinsuke Mori, (2011) Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis, the Proceedings of ACL-HLT 2011, pp 529-533.

[5] RyoheiSasano and SadaoKurohashi and Manabu Okumura, (2013) A simple approach to unknown word processing in japanese morphological analysis, the Proceedings of the Sixth International Joint Conference on Natural Language Processing, pp 162-170.

[6] Itsumi Saito and KugatsuSadamitsu and Hisako Asano and Yoshihiro Matsuo, (2014) Morphological Analysis for Japanese Noisy Text Based on Character-level and Word-level Normalization, the Proceedings of COLING 2014, pp 1773-1782.

[7] Rico Sennrich and Barry Haddow and Alexandra Birch, (2016) Neural Machine Translation of Rare Words with Subword Units, the Proceedings of the 54th ACL, pp1715-1725.

[8] Taku Kudo, (2018) Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates, the Proceedings of ACL 2018, pp 66-75.

[9] K.L. Kwok, (1997) Comparing Representations in Chinese Information Retrieval, the Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, pp34-41.

[10] Jian-Yun Nie and Jiangfeng Gao and Jian Zhang and Ming Zhou, (2000) On the Use of Words and N-grams for Chinese Information Retrieval, the Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, pp141-148.

[11] François Mairesse and Marilyn Walker, (2007) PERSONAGE: Personality Generation for Dialogue, the Proceedings of ACL 2007, pp496-503.

[12] Marilyn A. Walker and Grace I. Lin and Jennifer E. Sawyer, (2012) An Annotated Corpus of Film Dialogue for Learning and Characterizing Character Style, the Proceedings of LREC 2012, pp1373–1378.

[13] Chiaki Miyazaki and Toru Hirano and Ryuichiro Higashinaka and Toshiro Makino and Yoshihiro Matsuo and Satoshi Sato, (2014) Basic Analysis of Linguistic Peculiarities that Contribute Characterization of Dialogue Agent, the Proceedings of NLP2014, pp232-235(In Japanese).

[14] Chiaki Miyazaki and Toru Hirano and Ryuichiro Higashinaka and Yoshihiro Matsuo, (2016) Towards an Entertaining Natural Language Generation System: Linguistic Peculiarities of Japanese Fictional Characters, the Proceedings of SIGDIAL 2016, pp319–328.

[15] Chiaki Miyazaki and Toru Hiranoand Ryuichiro Higashinaka and Toshiro Makino and Yoshihiro Matsuo, (2015) Automatic conversion of sentence-end expressions for utterance characterization of dialogue systems, the Proceedings of PACLIC 2015, pp307–314.

[16] [16] Chiaki Miyazaki and Satoshi Sato, (2019) Classification of Phonological Changes Reflected in Text: Toward a Characterization of Written Utterances, Journal of Natural Language Processing, Vol. 26, No.2, pp407-440(In Japanese).

[17] Sohei Okui and Makoto Nakatsuji, (2020) Evaluating response generation for character by pointer-generator-mechanism, Proceedings of the 34th Annual Conference of the Japanese Society for Artificial Intelligence, pp1I4-GS-2-01(In Japanese).

[18] Taku Kudo and John Richardson, (2018) SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing, the Proceedings of EMNLP 2018, pp66-71.

AN ADAPTIVE AND INTERACTIVE EDUCATIONAL GAME PLATFORM FOR ENGLISH LEARNING ENHANCEMENT USING AI AND CHATBOT TECHNIQUES

Yichen Liu[1], Jonathan Sahagun[2] and Yu Sun[3]

[1]Shen Wai International School, 29,
Baishi 3rd Road Nanshan Shenzhen China 518053
[2]California State Polytechnic University, Los Angeles, CA, 91748
[3]California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

As our world becomes more globalized, learning new languages will be an essential skill to communicate across countries and cultures and as a means to create better opportunities for oneself [4]. This holds especially true for the English language [5]. Since the rise of smartphones, there have been many apps created to teach new languages such as Babbel and Duolingo that have made learning new languages cheap and approachable by allowing users to practice briefly whenever they have a free moment for. This is where we believe those apps fail. These apps do not capture the interest or attention of the user's for long enough for them to meaningfully learn. Our approach is to make a video game that immerses our player in a world where they get to practice English verbally with NPCs and engage with them in scenarios they may encounter in the real world [6]. Our approach will include using chatbot AI to engage our users in realistic natural conversation while using speech to text technology such that our user will practice speaking English [7].

KEYWORDS

Machine Learning, NLP, Data Mining, Game Development.

1. INTRODUCTION

In the modern age, though many claim there are more opportunities than ever before, there is still a massive disparity between the rich and the poor. This is especially due to the fact that such opportunities cannot be accessed by just anyone. Language is one of the major barriers, especially for English, the most used language both by country and by population, and it is seen by many as a door to unlocking knowledge and catching up to the rest of the world, given that most resources are either written in, or translated to, English. Students in poorer regions do not have an environment conducive to learning all aspects of English, often learning with teachers who can barely speak it themselves, and textbooks decades old. Many rural students end up being Receptively Bilingual (knowing the words and being able to read/write but can't speak), which is obviously unideal, as speaking is what gives first impressions [8]. Thus, it is absolutely critical to create tools pushing for the learning of English in a comprehensive manner, providing an environment that especially encourages speaking [9]. In this way, we can start to bridge the gaps between people of different social economic status with a simple language difference.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 97-106, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112308

The education market is expansive, and given the proliferation of English, there have obviously been hundreds of thousands of companies around the world offering their own take [10]. There are general language apps, such as Duolingo, which focus on all four aspects (reading, writing, listening, speaking) of dozens of languages. However, they're usually aimed at adults (kids versions are all very lacking), and spread themselves too wide to be effective beyond basic conversational level. They also generally operate with a very automated learning style, such as mundane drag and drops or listening exercises. I have also looked at the larger market of English-only apps, especially for children. Yet, what we have found is that while most of them operate on the effective principle of "learn by play," they skew a little bit too much towards the play [11]. There are entire games focused around a single word, often a trivial one, such as "Carrot." Even when more learning-heavy exercises are present, it still focuses on vocabulary and listening, as it becomes a more mundane and responsive task, rather than an active one. It is important to take advantage of the young age of children to create a conducive learning environment, rather than setting them up for more work later on.

In our own method, we try to create an environment that incorporates learn by play but also focuses heavily on speaking. The effectiveness of gamified approaches to learning cannot be denied, but a balance must be met between fun and learning, as always. A good way to do this is by setting learning objectives to being the keys of certain checkpoints or to unlocking certain rewards, instead of, like in so many other apps, having them simply as a "added feature," for example, collecting words while riding a horse. There are many existing tools which can teach vocabulary and basic grammar, or even listening, and there indeed are apps which focus just on that. Thus, our goal is simply to establish a unique environment and differentiate from existing tools—though later on such features can also be added on for a comprehensive software. A focus on speaking aims to directly fight against Receptive Bilingualism, and with an AI able to respond in real time, children are able to develop confidence through real interaction, instead of simply responding [12]. When text to speech is incorporated, students are then given the opportunity to mimic the pronunciation and try it out, with the reward given when the computer deems it accurate, providing real time comparisons.

After we have completed the program design and released the prototype, we need to design some reasonable experiments to verify our hypothesis. In order to make the experiment more realistic and reliable, we sampled from different populations to ensure the stability of the experimental results [13]. Our two experimental designs are as follows：

1) In order to verify the stability of the process, we selected 30 different teachers and teacher assistants from 10 different regions. We ask teachers to record their standard pronunciation and ask teachers to test our English learning games with standard grammar and pronunciation. In the end, most teachers achieved high scores in the game. The accuracy of game prediction is as high as 93%.

2) In order to test the effectiveness of our games on English learning, we find 150 students from 20 different regions in China. They have different backgrounds and experiences in learning English.The final result shows that our software is the most effective for students who have been studying English for 1-3 years.

The rest of the paper is organized as follows: Section 2 gives the details on the challenges that we met during the experiment and designing the sample; Section 3 focuses on the details of our solutions corresponding to the challenges that we mentioned in Section 2; Section 4 presents the relevant details about the experiment we did, following by presenting the related work in Section 5. Finally, Section 6 gives the conclusion remarks, as well as pointing out the future work of this project.

2. CHALLENGES

In order to build the tracking system, a few challenges have been identified as follows.

2.1. Language or cultural barrier

When creating our first chat bot we quickly noticed how many words could be used interchangeably to carry the same meaning or a valid response to a question especially for new learners of a language who may not be familiar with colloquialisms and idioms of the language [14]. For example, colloquially in American English the biggest pizza is referred to as a “large pizza” but ordering a “huge pizza” isn’t grammatically incorrect. Another issue with particular word choices is catching terms and trying to interpret what they meant which can be caused due to a language or cultural barrier. Back to a pizza example, a pizza topping that may be common outside of the US could be considered very unusual in the US. In this case we would want the chatbot understand that the uncommon topping that is being order is a topping that isn’t offered and respond according, for example “Sorry we don’t off that as a topping” vs a more generic unhelpful response of “Sorry we don’t quite get that, may you repeat that.” The former will help the conversation move forward while the latter may stall the flow of the conversation and hinder the user’s learning.

2.2. Answering open ended questions

Our goal is to teach conversational English by simulating real world scenarios. To accomplish thisgoal, we want to simulate a natural conversation. Doing so involves open ended questions, allowing users to answer how they feel is natural which can be difficult to account for. Unfortunately, we cannot fully predict how many new English learners will answer. The best remedy to solve this issue is to grow our user base and collect users’ responses and adjust our chatbot to accommodate how people respond by having more guided responses.

2.3. Delay of response

As we are trying to gamify learning English, one of the main goals is to make the learning process fun and approachable. A big obstacle to fun is waiting. One issue we are facing is the delay between the user’s response and the computer’s response and can cause the user to wait [15]. We are relying on cloud services to handle our chatbot and speech to text. A few solutions we are looking forward to try out include better compression of our audio recording of the user, streaming the audio directly to our speech to text services vs wait sending a audio file after the user finishes speaking, and implementing a webhook that sends information directly to the other cloud services. We had to scrap our implementation of text to speech because the user’s input lag would have been so great it would have made our game practically unplayable.

3. SOLUTION

We want to teach people English by making the learning fun by gamifying the process.

Introductory language textbooks use scenarios like ordering food and asking where the library is to immerse learners into scenarios, they may encounter in countries that speak that language. We are building upon that by using speak to text technology and chatbot artificial intelligence in a video game to further immerse learners and encourage them to practice English by speaking it. By gamifying English learning, we are hoping to build the habits that will motivate and engage with our users to keep practicing English every time the game is launched.

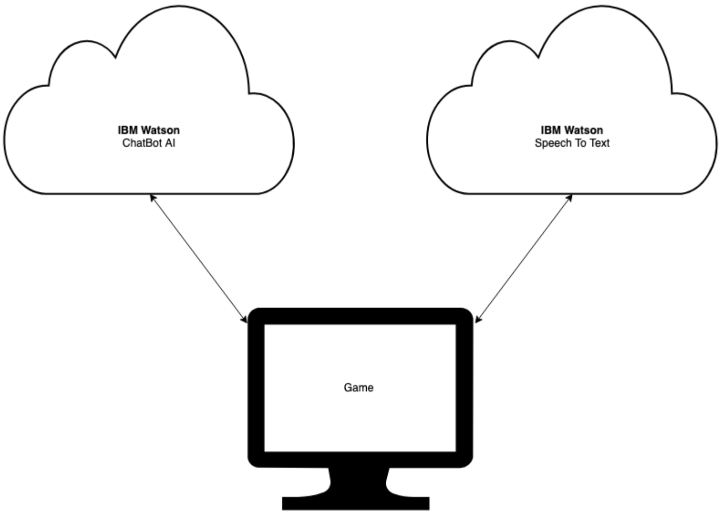
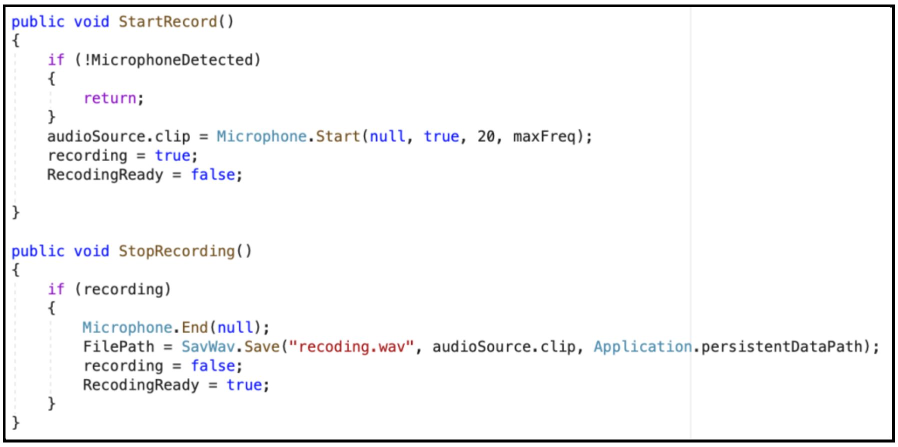


Figure 1. The overview of the project

Our game connects to two cloud services provided by IBM's Cloud services: IBM's Watson Speech to Text and Watson Assistant. We use Watson Speech to Text service to allow our users to practice speaking English and showing them how they pronounce their words encouraging our users to enunciate. The second service is Watson Assistant, which is IBM's chatbot solution. We use the chatbot to simulate real life situations to immerse our users in natural conversation.

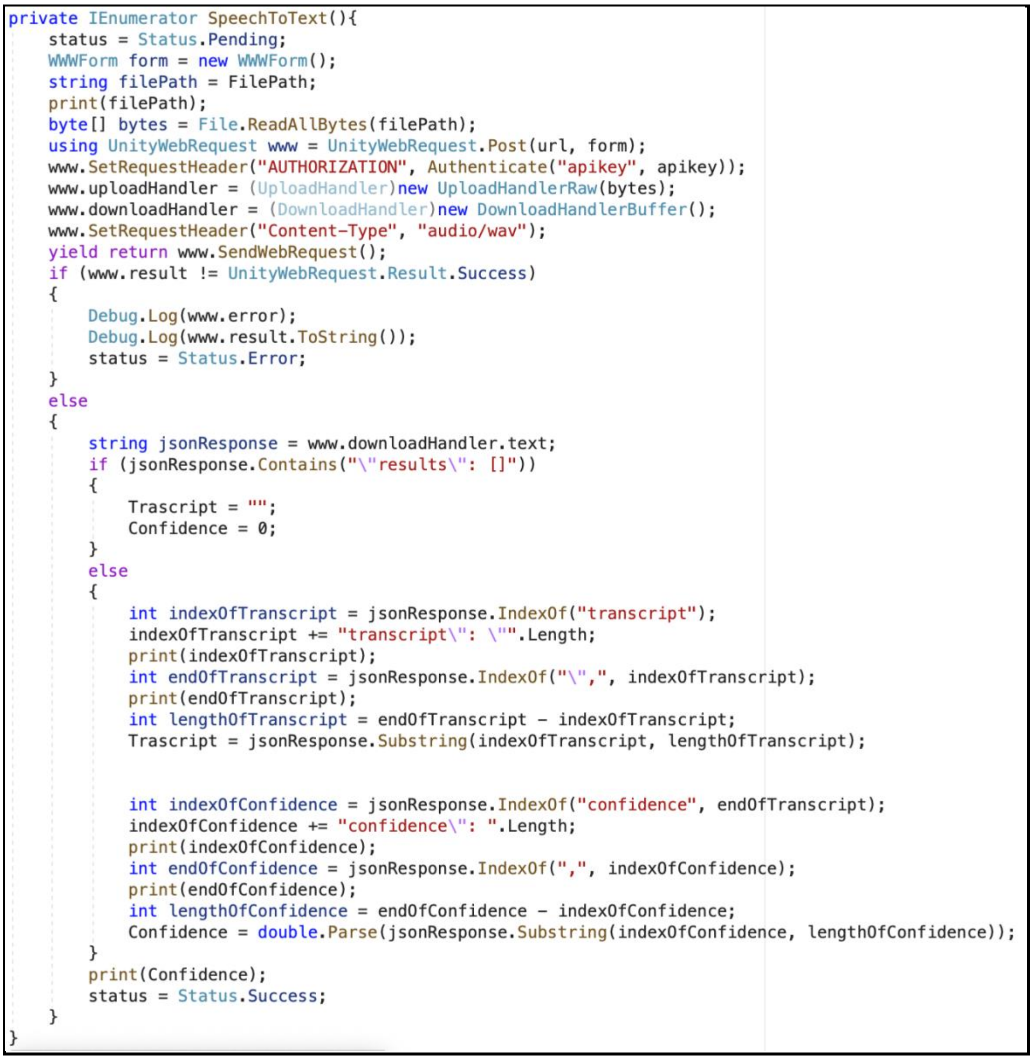
The IBM Watson Speech to Text service transcribes audio to text to enable speech transcription capabilities for applications.



```
public void StartRecord()
{
    if (!MicrophoneDetected)
    {
        return;
    }
    audioSource.clip = Microphone.Start(null, true, 20, maxFreq);
    recording = true;
    RecodingReady = false;

}

public void StopRecording()
{
    if (recording)
    {
        Microphone.End(null);
        FilePath = SavWav.Save("recoding.wav", audioSource.clip, Application.persistentDataPath);
        recording = false;
        RecodingReady = true;
    }
}
```

Figure 2. Code of recording

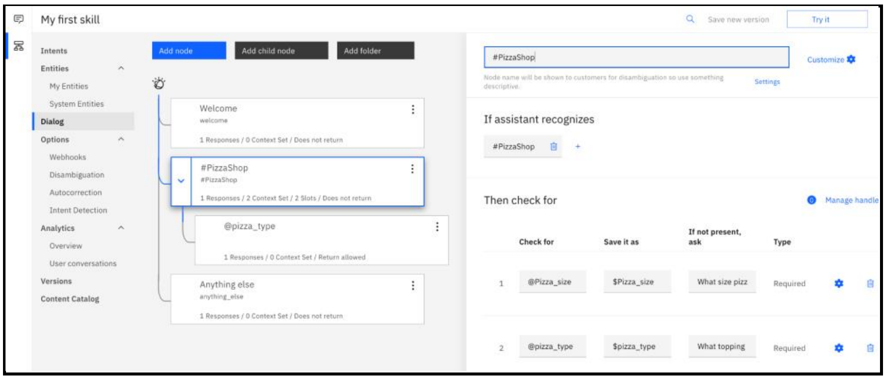
To use IBM's Speech to Text service we first need to have an audio file. Above is the code we use to record our user's voice. We are using Unity's Microphone to record the audio source. Unfortunately, it is in an uncommon audio file used by Unity so we encode it Waveform Audio File (WAV) format before saving to the user's computer. Once it's saved, we are ready to use the Speech to Text service.



```
private IEnumerator SpeechToText(){
    status = Status.Pending;
    WWWForm form = new WWWForm();
    string filePath = FilePath;
    print(filePath);
    byte[] bytes = File.ReadAllBytes(filePath);
    using UnityWebRequest www = UnityWebRequest.Post(url, form);
    www.SetRequestHeader("AUTHORIZATION", Authenticate("apikey", apikey));
    www.uploadHandler = (UploadHandler)new UploadHandlerRaw(bytes);
    www.downloadHandler = (DownloadHandler)new DownloadHandlerBuffer();
    www.SetRequestHeader("Content-Type", "audio/wav");
    yield return www.SendWebRequest();
    if (www.result != UnityWebRequest.Result.Success)
    {
        Debug.Log(www.error);
        Debug.Log(www.result.ToString());
        status = Status.Error;
    }
    else
    {
        string jsonResponse = www.downloadHandler.text;
        if (jsonResponse.Contains("\"results\": []"))
        {
            Trascript = "";
            Confidence = 0;
        }
        else
        {
            int indexOfTranscript = jsonResponse.IndexOf("transcript");
            indexOfTranscript += "transcript\": \"".Length;
            print(indexOfTranscript);
            int endOfTranscript = jsonResponse.IndexOf("\",", indexOfTranscript);
            print(endOfTranscript);
            int lengthOfTranscript = endOfTranscript - indexOfTranscript;
            Trascript = jsonResponse.Substring(indexOfTranscript, lengthOfTranscript);


            int indexOfConfidence = jsonResponse.IndexOf("confidence", endOfTranscript);
            indexOfConfidence += "confidence\": ".Length;
            print(indexOfConfidence);
            int endOfConfidence = jsonResponse.IndexOf(",", indexOfConfidence);
            print(endOfConfidence);
            int lengthOfConfidence = endOfConfidence - indexOfConfidence;
            Confidence = double.Parse(jsonResponse.Substring(indexOfConfidence, lengthOfConfidence));
        }
        print(Confidence);
        status = Status.Success;
    }
}
```

Figure 3. Code of speech to text

Above is our code used to send the WAV file we saved of the user's recording to IBM. It is sent over using a HTTPS request. The request we sent contains the audio recording as a byte array and our API key for authentication. In return we expect a JSON response. There are two keys we are looking for in this JSON, the transcript of the audio recording the IBM's confidence level.



Welcome
welcome
1 Responses / 0 Context Set / Does not return

#PizzaShop
#PizzaShop
1 Responses / 2 Context Set / 2 Slots / Does not return

@pizza_type
1 Responses / 0 Context Set / Return allowed

Anything else
anything_else
1 Responses / 0 Context Set / Does not return

#PizzaShop

Node name will be shown to customers for disambiguation so use something descriptive.

If assistant recognizes

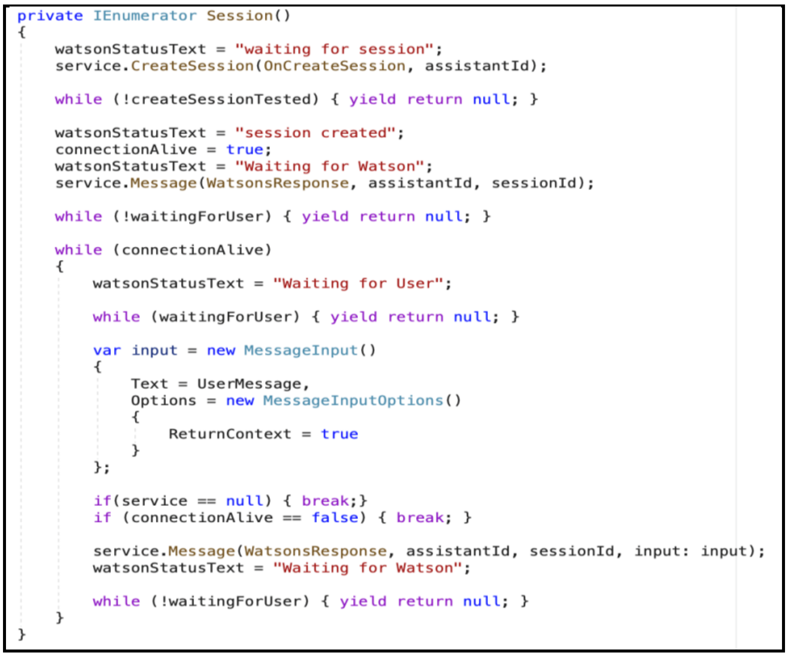
#PizzaShop

Then check for

	Check for	Save it as	If not present, ask	Type
1	@Pizza_size	$Pizza_size	What size pizz	Required
2	@pizza_type	$pizza_type	What topping	Required

Figure 4. Screenshot of sample conversation

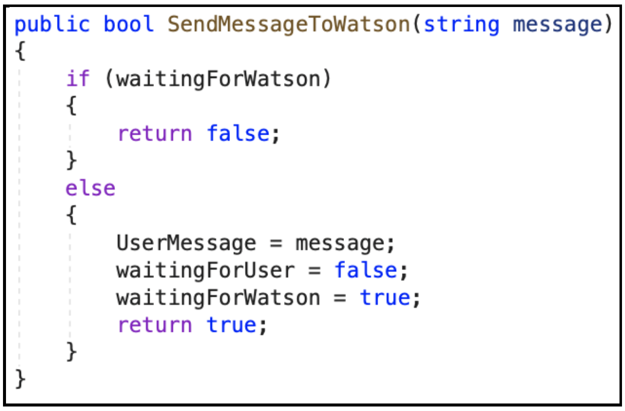
Above is a sample conversation we developed for a pizza shop. To key terms or entities, we check for are pizza size and pizza toppings. The user can ask for both for example “I would like to order a large cheese pizza” and that will satisfy our chatbot. In the case where the user only orders the size, our chat bot is able to adapt the conversion and ask the user what type of pizza they would like.



```
private IEnumerator Session()
{
    watsonStatusText = "waiting for session";
    service.CreateSession(OnCreateSession, assistantId);

    while (!createSessionTested) { yield return null; }

    watsonStatusText = "session created";
    connectionAlive = true;
    watsonStatusText = "Waiting for Watson";
    service.Message(WatsonsResponse, assistantId, sessionId);

    while (!waitingForUser) { yield return null; }

    while (connectionAlive)
    {
        watsonStatusText = "Waiting for User";

        while (waitingForUser) { yield return null; }

        var input = new MessageInput()
        {
            Text = UserMessage,
            Options = new MessageInputOptions()
            {
                ReturnContext = true
            }
        };

        if(service == null) { break;}
        if (connectionAlive == false) { break; }

        service.Message(WatsonsResponse, assistantId, sessionId, input: input);
        watsonStatusText = "Waiting for Watson";

        while (!waitingForUser) { yield return null; }
    }
}
```

Figure 5. Main code used to communicate with Watson

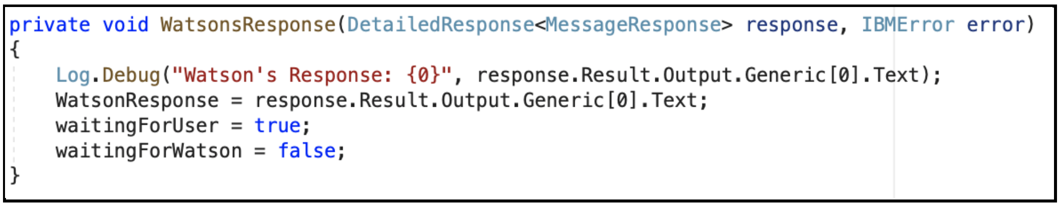
Here we have the main code used to communicate with Watson. The Session function creates a session to Watson Assistant where the assistantID variable is the specific scenario we want to connect to. Once the session has been established, Watson sends a message greeting the user starting the conversion for the user to follow. This function will loop indefinitely either waiting for the user’s response or Watson’s response to the user.



```
public bool SendMessageToWatson(string message)
{
    if (waitingForWatson)
    {
        return false;
    }
    else
    {
        UserMessage = message;
        waitingForUser = false;
        waitingForWatson = true;
        return true;
    }
}
```

Figure 6. Function of sending message

The function we created to send messages requires a string that is saved as an instance variable that will be read once the Session function is ready to use it. The message parameter of the SendMessageToWatson function in practice comes from the transcription of the user’s recording from the Speech to Text service. Once the UserMessage variable is set, the waitingForUser and waitingForWatsonboolean variables are set so the Session function can continue.



```
private void WatsonsResponse(DetailedResponse<MessageResponse> response, IBMError error)
{
    Log.Debug("Watson's Response: {0}", response.Result.Output.Generic[0].Text);
    WatsonResponse = response.Result.Output.Generic[0].Text;
    waitingForUser = true;
    waitingForWatson = false;
}
```

Figure 7. Code of Watson's response

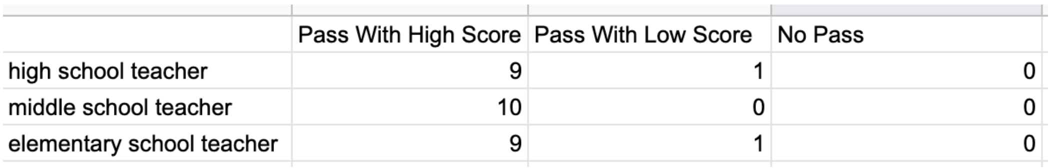
When Watson is sent the user's response we wait for Watson's response. When Watson sends its response to the user that triggers a callback function in which we save Watson's response, used to display the response in game, and we again toggle the waitingForUser and waitingForWatson variables, setting the game state so that the user may respond.

Our biggest limitation we are facing is gathering enough users of varied diverse backgrounds to further program our scenarios to capture common errors and edge cases in how people respond. We plan to keep developing our chat bot to catch more edge cases and develop more scenarios to focus on specific topics such as creating a character who you can ask for directions and creating bus stops that players can use to navigate the world using the characters directions. The biggest technical limitation is the processing time from the speak to text process and then taking that text and having the chatbot process it. Since both processes are done in the cloud the user may experience a long delay from the time the user finishes his speech to the game's on screen response. We originally intended to have the text to speech response to answer the player but that would have delayed the computer's response even further which would make the game feel unplayable.

4. EXPERIMENT

Experiment 1

We considered a lot of issues when designing experiment 1. In order to eliminate all possible influencing factors and to ensure the diversity of samples, we selected 30 different teachers and teacher assistants from 10 different regions in the US. 10 of them are teaching elementary school, 10 of them are teaching middle school, 10 of them are teaching high school. We ask teachers to record their standard pronunciation and ask teachers to test our English learning games with standard grammar and pronunciation. The result Table shows below:



	Pass With High Score	Pass With Low Score	No Pass
high school teacher	9	1	0
middle school teacher	10	0	0
elementary school teacher	9	1	0

Figure 8. Table of data

From the data in the table, we can tell most teachers achieved high scores in the game. The accuracy of game prediction is as high as 93%. Base on the experiment 1 we can prove that users can pass the game if they have good English skill.

Experiment 2

Before designing the second experiment, we considered how to design the experiment to show that our software is effective for English learning. First of all, the selected student sample should not come from a country where English is a native language. Secondly, in order to ensure the diversity of the samples, the distance between these students should be far enough. Finally, we chose China as the sample area for the experiment. We find 150 students from 20 different regions in China and divide them into 3 groups. Group1 is the students who have learned English 1 -3 years before the test. Group2 the students who have learned English 3 - 5years before the test. the students who have learned English more than 5 years before the test. They have different backgrounds and experiences in learning English. We ask them to download the game and try to learn English with it. After few weeks we ask them to finish the survey and collect the result, the graphic shows below:

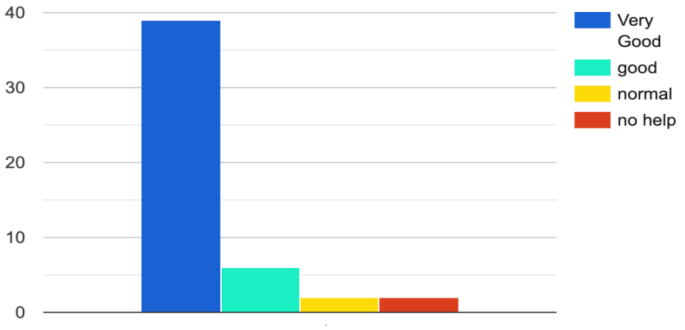


Figure 9. Result of 50 students from 1 - 3 years experience group

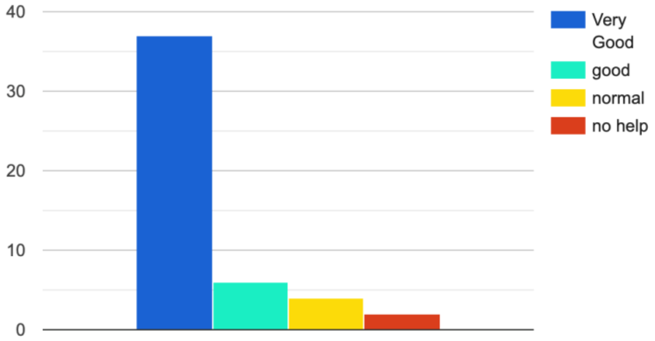


Figure 10. Result of 50 students from 3 - 5 years experience group

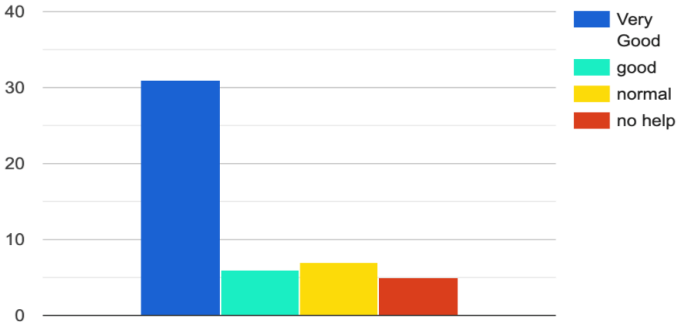


Figure 11. Result of 50 students from more than 5 years experience group

The final result shows that our software is the most effective for students who have been studying English for 1-3 years.

In order to eliminate all possible influencing factors and to ensure the diversity of samples, we selected 30 different teachers from 10 different regions in the US. The accuracy of game prediction is as high as 93%. We can prove that users can pass the game if they have good English skills.

in order to ensure the diversity of the samples, we find 150 students from 20 different regions in China divide them into 3 groups base on their English experiment, we prove that our software is the most effective for students who have been studying English for 1-3 years.

5. RELATED WORK

Chen and Tsai proposed a game-based English learning system with a context-aware interactive learning mechanism, which can appropriately provide learners with corresponding game-based English learning scenarios based on the learner's geo location [1]. The proposed system aims to construct an augmented reality game-based learning environment that combines virtual objects with real scenes. This game does not involve voice-based interaction as what we have accomplished. In addition, this game is based on a hard-coded environment without AI-based conversations.

Wu and Huang developed a game-based system to help learners to improve their vocabulary [2]. Users can choose different settings, portfolio, and levels, in order to adjust the study mode and pace. This project focused on improving the motivation of vocabulary learning, while ours focuses on AI-based interaction and conversation to improve the practical aspect of English.

Hung and Young proposed a complete hardware device that provides an interactive English vocabulary learning board game [3]. A complete study has been done by collecting both quantitative and qualitative data. The results showed that game-embedded handheld devices could increase the interdependence of the group and help the learners to improve the engagement and immersion. Our system is independent from the devices, and the same application can be running cross platform in mobile, web, PC, and embedded environments.

6. CONCLUSIONS

We created this game as a fun way to teach English with the main goal being to teach conversational English by simulating real world scenarios. Our approach was to use chatbot AI and speech to text technology by IBM to allow users to interact with NPCs in natural conversations by speaking with the NPCs. For example, the first scenario we created is a pizza parlor where the user orders a Pizza. The scene starts by walking up to the NPCs and interacting to start the conversation. The NPC starts the conversation by greeting the user and providing the user with info about what the scenario is about and what some appropriate responses are. The conversation continues by giving control to the user and allowing them to record their voice. Once the user is finished responding, their recording is sent to IBMs' a speech to text service and the output of that is sent to our chatbot which responds to the user. This is looped until the conversation and scenario reaches its end.

We sample 3 groups of 50 students from 20 different regions in China. The first group have 1 - 3 years of English experience, the second group have 3 - 5 years and the final group have 5 or more years. We had them take a proficiency test before testing our app for a control. After a week of using our game, we had them take another proficiency test. The final result shows that our software is the most effective for students who have been studying English for 1-3 years.

To further enhance our research, we need more user's data to understand our shortcomings in our scenarios to see where we can improve our wording and guide our user's to appropriate responses. Without this data we can't not make meaningful changes to our conversations to make them feel more natural. We will be looking for users to help us test our chatbot and ways to make our game more fun and engaging.

To better immerse our users in the game we are looking to implement text to speech technology for the NPCs to respond. We also are looking into shortening the delay between the user's response to the NPC's. We believe we can achieve that by streaming the user's microphone to our speech to text service, then having a web hook redirect that outcome to our chatbot.

REFERENCES

[1] Chen, Chih-Ming, and Yen-Nung Tsai. "Interactive location-based game for supporting effective English learning." In 2009 International Conference on Environmental Science and Information Application Technology, vol. 3, pp. 523-526. IEEE, 2009.

[2] Wu, Ting-Ting, and Yueh-Min Huang. "A mobile game-based English vocabulary practice system based on portfolio analysis." Journal of Educational Technology & Society 20, no. 2 (2017): 265-277.

[3] Hung, Hui-Chun, and Shelley Shwu-Ching Young. "An investigation of game-embedded handheld devices to enhance English learning." Journal of Educational Computing Research 52, no. 4 (2015): 548-567.

[4] Katzner, Kenneth, and Kirk Miller. The languages of the world. Routledge, 2002.

[5] Barber, Charles, Joan C. Beal, and Philip A. Shaw. The English language: A historical introduction. Cambridge University Press, 2009.

[6] Schnaars, Steven P. "How to develop and use scenarios." Long range planning 20.1 (1987): 105-114.

[7] Dahiya, Menal. "A tool of conversation: Chatbot." International Journal of Computer Sciences and Engineering 5.5 (2017): 158-161.

[8] Woods, Michael. Rural. Routledge, 2010.

[9] Belsey, Catherine. Critical practice. Routledge, 2003.

[10] White, Ellen Gould Harmon. Education. AB Publishing, 1903.

[11] Ryu, Dongwan. "Play to learn, learn to play: Language learning through gaming culture." ReCALL 25.2 (2013): 286-301.

[12] Cox, David R. "Interaction." International Statistical Review/Revue Internationale de Statistique (1984): 1-24.

[13] Hahn, Frank. "Stability." Handbook of mathematical economics 2 (1982): 745-793.

[14] Nevalainen, Sampo. "Colloquialisms in translated text. Double illusion?." Across Languages and Cultures 5.1 (2004): 67-88.

[15] March, Wayne F. "Dealing with the delay." Diabetes Technology & Therapeutics 4.1 (2002): 49-50.

A DEEP LEARNING APPROACH TO INTEGRATE HUMAN-LEVEL UNDERSTANDING IN A CHATBOT

Afia Fairoose Abedin[1], Amirul Islam Al Mamun[1],
Rownak Jahan Nowrin[1], Amitabha Chakrabarty[1],
Moin Mostakim[1] and Sudip Kumar Naskar[2]

[1]Department of Computer Science and Engineering,
Brac University, Dhaka,Bangladesh
[2]Department of Computer Science and Engineering,
Jadavpur University, Kolkata, India

ABSTRACT

In recent times, a large number of people have been involved in establishing their own businesses. Unlike humans, chatbots can serve multiple customers at a time, are available 24/7 and reply in less than a fraction of a second. Though chatbots perform well in task-oriented activities, in most cases they fail to understand personalized opinions, statements or even queries which later impact the organization for poor service management. Lack of understanding capabilities in bots disinterest humans to continue conversations with them. Usually, chatbots give absurd responses when they are unable to interpret a user's text accurately. Extracting the client reviews from conversations by using chatbots, organizations can reduce the major gap of understanding between the users and the chatbot and improve their quality of products and services.Thus, in our research we incorporated all the key elements that are necessary for a chatbot to analyse andunderstand an input text precisely and accurately. We performed sentiment analysis, emotion detection, intent classification and named-entity recognition using deep learning to develop chatbots with humanistic understanding and intelligence. The efficiency of our approach can be demonstrated accordingly by the detailed analysis.

KEYWORDS

Natural Language Processing, Humanistic, Deep learning, Sentiment analysis, Emotion detection, Intent classification, Named-entity recognition.

1. INTRODUCTION

In recent times, a large number of people are involved in establishing their own business. But it is tremendously tough to stay updated with technology and hold a place in the market full of competition. Chatbot is one of the most advanced features incorporated in most organizations or online platforms today. A customer care team has a lot of constraints to working hours, responding to multiple at the same time and also efficiency. Conversational agents definitely solve all the problems. Unlike humans, it doesn't get tired or make delays to reply. Chatbots handle similar questions easily, help firms in advertising their brands in a very cost-effective way and also help organizations to overcome their drawbacks by analysing customer feedback [1]. Not only in business, but also applications of chatbots are emerging in medical fields as well. Chatbot

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 107-122, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112309

applications are developed to imitate psychiatrist techniques to uplift a person's mood or reduce stress. Completely replacing a human might be quite challenging, but conversational agents can reduce a lot of human effort. There are many conventional AI-bots (Artificial Intelligence Bots) in the market which generate fixed responses and thus can be monotonous at times for the user asit gives redundant replies. Usually, it identifies the most similar keywords and responds to the closely matched text from its database. Eventually, the replies become inconsistent and meaningless most of the time. Even if we talk about one of the first stable virtual assistants, SIRI, which is developed by Apple, has a vast range of features helping users to manage calendars, make calls, schedule meetings and what not. However, if you have ever tried to talk with it more interactively, it often fails to understand and most frequently replies *"I am sorry, I couldn't understand. Could you try again?"*. Furthermore, using a bot in the psychological field needs to be even more accurate and human-like by giving the vibe of human-connection and empathy. That's why even today most users prefer human agents to solve their problems. For a virtual agent to respond like a human, it needs to understand the problem of the user and provide sensible replies to humans [2]. So, the aim of our research is to analyse the various features that are required by a chatbot to gain a human-like understanding of text. We are trying to develop a hybrid model that will understand the sentiment, emotion, intent and named-entities of the user's text. Once this problem is solved, it can be used in every sector for personalised conversations and this data can be very helpful to the organizations whether to help a mental health patient or to improve various services and products sold. Even universities can use it for finding out the exact problem that a student is going through by analysing the bot-human conversation. We chose particularly deep learning because DL models help to extract the meaning of mixed contrary complex sentences given by a user more accurately. The layers in DL models creates a network that helps bots to learn accurately on their own which normal ML bots fail to do [3]. Hence, we believe our research will help to improve chatbot models definitely and open the doors of vast and effective bot-applications.

2. RELATED WORK

To know about previous works on sentimental analysis, we read the research paper [4], where text mining techniques were applied to find sentiment and emotion from twitter dataset. Words from twitter were compared with the sentiment file using Bayes algorithm and given the corresponding sentiment. However, it is unable to predict sentiment of any word apart from the words stored in the database and therefore, we get a scope here to improve the drawbacks of their work. Also, an interesting Bangla intelligent social robot was introduced in the paper [5] which communicates in Bangla analysing sentiments with the help of machine learning algorithms. The authors of paper [5] have refined a huge amount of data to fit in their machine learning algorithm. Fictional conversation extracted from raw movie script were used as the metadata collection of the corpus. Naive Bayes classification algorithm was used to train the dataset. To work on context understanding of a chatbot, we went through the paper [6], which was a chatbot application based on NLP along with emotion recognition. The authors added emotion embedding vectors in the output layer of RNN. The system uses 4 levels of hierarchy to understand natural language input sentences and recognize the user's emotion. The response generation is done through collecting, analysing and integrating input dialogues consisting of text. Then document intention extraction is done from the text by neural network. The research paper [7] stated how deep learning can playa significant role in developing realistic chatbots by "mimicking human characteristics" and showed comparison between conversations by using different Neural Network models like Uni-directional LSTM, Bi-directional LSTM, Bi-directional LSTM with attention mechanism. The authors implemented a hybrid chatbot using rule-based and generative models together to makea more meaningful response. According to paper [8], to understand the user's current state of mind and wellness in depth, emotion recognition is helpful which is a detailed version of sentimental analysis. The paper further discusses the various datasets to perform emotion recognition and tabulated a list of already implemented emotion recognition and their limitations. The paper [9] detects emotion from a text using Bi-directional neural networks and tensor flow libraries. It describes how sentiment and emotion are interconnected and can be used to give betterresponses. The dataset is cleaned, processed and tokenized to fit to Bi-RNN model for training. The model had two parameters, one for sentimental analysis and another to generate dialogue. The work wasn't very efficient as they simply labelled the emotion as happy if sentiment is positive and sad if sentiment is negative. It did not classify the emotions and thus, failed to actuallysignify the improvement that could be achieved by classifying detailed emotion from the polarity of sentiments. The model played a happy song if the sentiment was positive and vice versa. However, our work completed the drawbacks of this paper by distributing emotion into different classes. Moreover, the authors of the paper [10] surveyed the effectiveness of implementing Named-Entity Recognition by deep learning. The author gave the information of named-entity datasets and where they can be used. The paper stated the different evaluation metrics of NER such as type, exact match, relaxed F1 and strict F1 metrics etc to analyse the performance and accuracy of NER. The survey analysed knowledge-based systems, unsupervised systems, feature-engineered systems, feature-inference neural network systems, word level architectures, character-level architectures, character+word level architecture and lastly, character+word+affix model. After evaluation it was noticed that neural network models outperformed all other applied models mentioned above. Furthermore, the paper [11] proposed their research on Dialogue intent classification by hierarchical LSTM which is an open-source library. Through this a model will be able to recognise the reason behind the uttered sentence. The experiment was performed on a Chinese ecommerce site. The authors compared accuracy between basic LSTM, HLSTM and their hybrid model HLSTM + Memory. By going through all the knowledgeable and qualitative research, we gathered the loopholes of all the papers that needed improvement as well we ensured that our working domain, deep learning, is the right choice for incorporating all the works we studied so far. In our literature review, we have seen the development of models by sentimental analysis or emotion or intent or named-entity. However, there hasn't been yet an analysis that clusters all the components together. Thus, we decided to observe the different performances as well as challenges while using all the features together.

3. RESEARCH METHODOLOGY

We started our research by analysing a commonly used feature - **sentiment**, which is incorporated in most customer service chatbots to help the company rate their products. It is also essential in various movie reviews or story book feedback. We implemented various deep learning models from simple neural networks to LSTM and GRU. Based on accuracy, we selected LSTM as the best model for our sentimental analysis. Next, through our study, we came up to classify sentiment into more meaningful expression that is **emotion detection**. We used 7 emotions of joy, sadness, disgust, shame, guilt, anger and fear to identify the specific mental state of the user. We applied Bi-LSTM and Bi-GRU for predicting the correct emotion. Next, we decided to bring out the **intent** of the conversation. We need to know the intent of the text in order to understand what the user wants or what the user needs. Similar to emotion detection, we used Bi-LSTM and Bi-GRU to experiment the intent classification. For getting more details about intent, we performed **named-entity recognition** as well. Named-entity recognition is important because it identifies the person, place, organization, region or even the specific date or time the user is talking about. For example- *"What is the time in Bangladesh now?"* Here *"knowing time"* is the intent and *"Bangladesh"* is the entity. Like emotion detection and intent classification, we used Bi-LSTM and Bi-GRU to recognize named-entity. As we have performed different analysis on different human understanding features it was quite impossible to find one single dataset to incorporate all the analysis. Thus, all our feature analysis was executed by independent datasets. As our domain was deep learning, we performed all our analysis within this field. Despite the unavailability of one dataset, we tried to integrate the analysed models so that all the features can be extracted froma single text. To give accurate results, Deep Learning bots need continuous training. So, it was quite obvious to get incorrect results initially. However, through our accuracy of models, estimations and various research we proposed, a text analysis structure that will give a chatbot multimodal understanding. The following Figure 1 shows the workflow of our executed analysis.

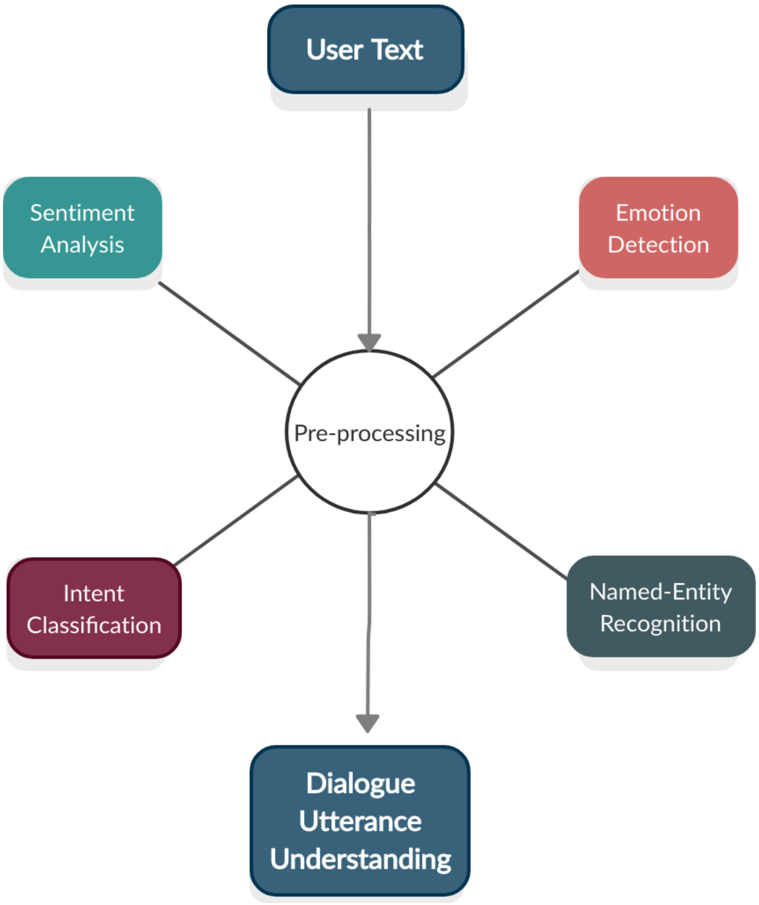


Figure 1. Analysis work flow for dialogue utterance understanding

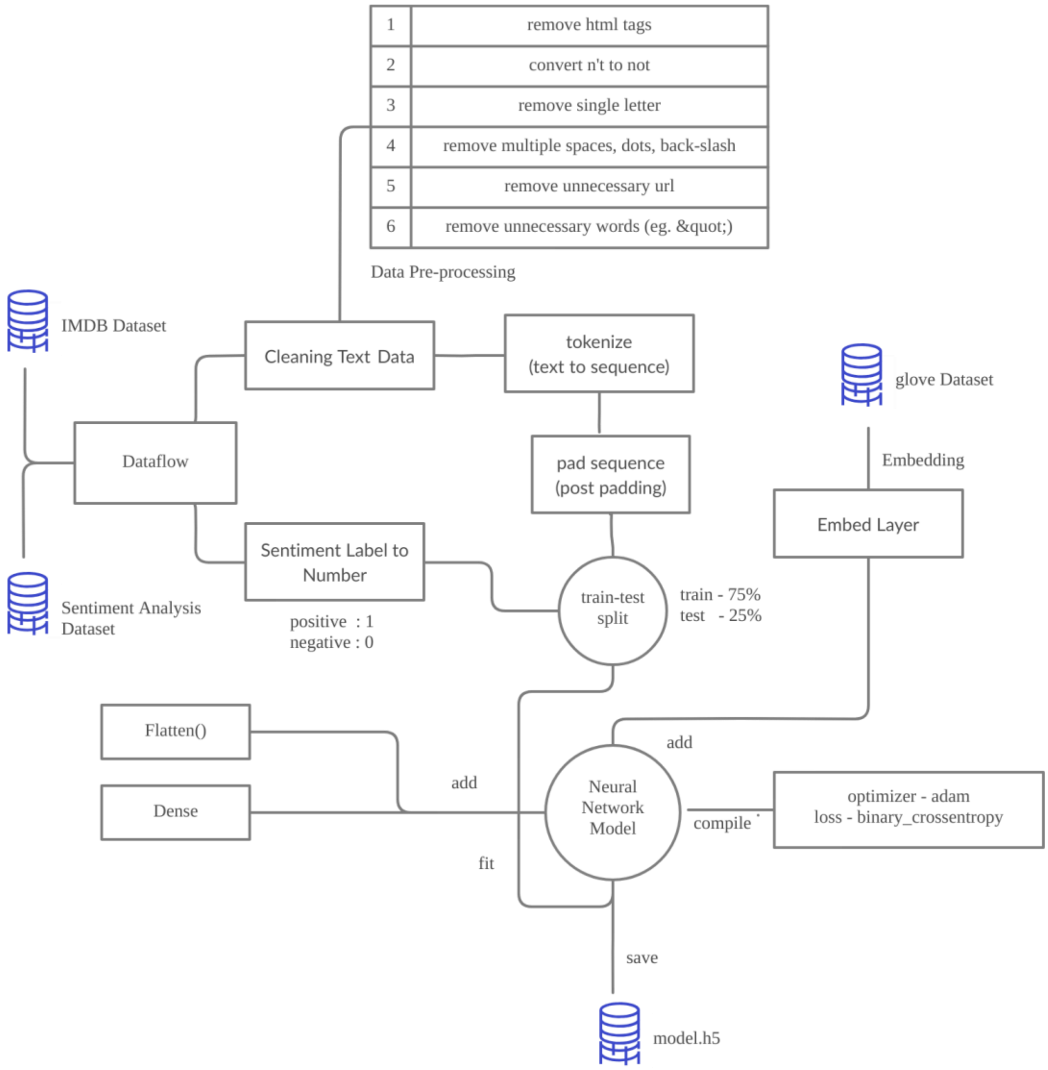
4. Experimental Analysis and Results

4.1. Sentiment Analysis

For the analysis of information in texts where opinions are highly unstructured and are either positive or negative sentiment analysis is essential [12]. For humans, it is very simple to identify positive and negative sentences. But, a chatbot can never understand the polarity of a text if it is not trained to identify it. As our goal is to create a human-like understanding of text in a chatbot,it is important to understand the sentiment of the content. We applied a simple neural network, GRU and LSTM to perform sentiment analysis on our sentiment datasets. For our work, we used two datasets for analysing our performance in different neural network models with different datasets.

4.1.1. Data Processing

The first dataset is IMDB dataset of movie review which consist of two columns [13]. The second dataset is twitter sentiment analysis dataset which has two columns as well [14]. After cleaning and refining the two datasets, we split them to test and train sets in the ratio 1:3 respectively. Then, we tokenized the datasets separately to fit into texts and further converted texts into sequence. In Figure 2, the whole workflow process of sentiment analysis is described in depth.



1	remove html tags
2	convert n't to not
3	remove single letter
4	remove multiple spaces, dots, back-slash
5	remove unnecessary url
6	remove unnecessary words (eg. &quot;)

Figure 2. Training Work Flow for Sentiment Analysis

4.1.2. Method

For capturing the context of a word in the datasets, we used Stanford's Glove word embedding which is an unsupervised learning algorithm for obtaining vector representations for words [15]. As different neural network models give different results and accuracy, we tried to implement a number of them such as Simple Neural Network, LSTM and GRU. In our models we have used sigmoid activation function and Adam optimizer. It was important to keep our batch size and epoch moderate to get rid of the overfitting problem. So, we kept the batch size at 128 and epoch to 20 for all our models. We kept the validation split to 0.2. After that we feed the two datasets into the model following the above parameters and demonstrated the different accuracy results of the two datasets.

4.1.3. Result

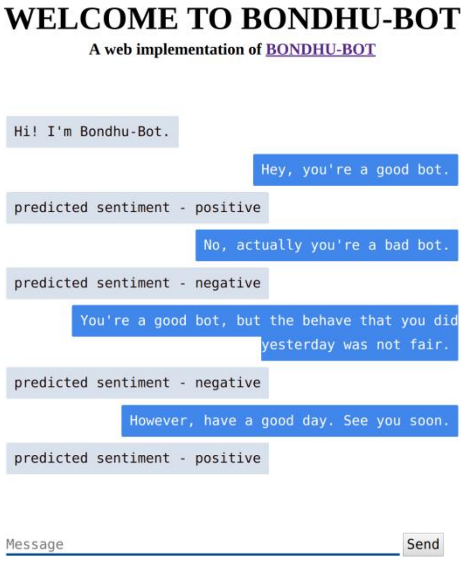
In Table 1, we showed comparison between the accuracy and loss of different DL models. From our performance analysis we found LSTM gave the highest accuracy in both test and train dataset. The train accuracy in dataset 1 and 2 are 89% and 86% respectively and test accuracy are 80% and 71% respectively. In other models the test accuracy dropped significantly compared to training accuracy.

Table 1. Accuracy for training and validation data of different Datasets and different Deep-Learning Models for Sentiment Analysis

model	**Dataset**	**Train Data**			
		loss	accuracy	loss	accuracy
Simple Neural Network	IMDB Dataset	0.37	0.86	0.82	0.68
	Twitter Sentiment	0.54	0.74	0.66	0.65
Gated Recurrent Unit	IMDB Dataset	0.32	0.91	0.95	0.79
	Twitter Sentiment	0.74	0.80	1.19	0.68
Long-Short Term Memory	IMDB Dataset	0.33	0.89	0.65	0.80
	Twitter Sentiment	0.43	0.86	0.92	0.71

4.1.4. Performance analysis

Finally in Figure 3, are some sentences given as input to make some predictions whether the sentences were classified correctly as positive or negative. We have set the range of confidence level of a sentence from 0 to 1. The sentences which have confidence levels greater than or equal to 0.5 are given a positive sentiment label and if it is less than 0.5, a negative sentiment label is given. The initial input text was *"Hey, you are a good bot"* and the bot was able to predict the sentiment 'positive' accurately. Usually, it is easier to extract sentiments from simple sentences, so we gave some complex sentences like *"You're a good bot but the behaviour that you did yesterday was not fair"*. Our bot was even successful in catching the right sentiment despite the complexity of text. It was able to consider the sentiment of both the clauses in the sentence and extract the sentiment with higher probability.



WELCOME TO BONDHU-BOT

A web implementation of BONDHU-BOT

Hi! I'm Bondhu-Bot.

Hey, you're a good bot.

predicted sentiment - positive

No, actually you're a bad bot.

predicted sentiment - negative

You're a good bot, but the behave that you did yesterday was not fair.

predicted sentiment - negative

However, have a good day. See you soon.

predicted sentiment - positive

Figure 3. Application of Sentiment Analysis

4.2. Emotion Detection

Sentiment analysis has helped us only to learn the attitude of a person from a user's input. For example- *"What a dream it was! I couldn't sleep at all"*, here by sentimental analysis we can classify this text as a negative sentence, however, this is not enough to understand a person's situation, we need to identify the exact mental state of the user. Let's analyse the above example again. If we closely analyse the text, we can see the person uttered the sentence out of some *'sadness'* and some *'fear'*. This is the exact state of mind we need to capture in our chatbot. Emotions are very subtle and ambiguous in a text which makes it really tough to detect. For that reason, we have focused on using neural network models in this paper to capture the emotion of the user.

4.2.1. Data Processing

The dataset used in this model is the International Survey on Emotion Antecedents and Reactions (ISEAR) database created by Klaus R. Scherer and Harald Wallbott which consists of 7665 sentences labelled by seven emotions like joy, sadness, fear, anger, guilt, disgust, and shame [16]. The dataset contains two columns: emotion label and text. As shown in Figure 4, we split all the phrases into smaller parts called tokens using Keras tokenizer. For stemming we used the Lancaster stemmer of NLTK (Natural language toolkit).

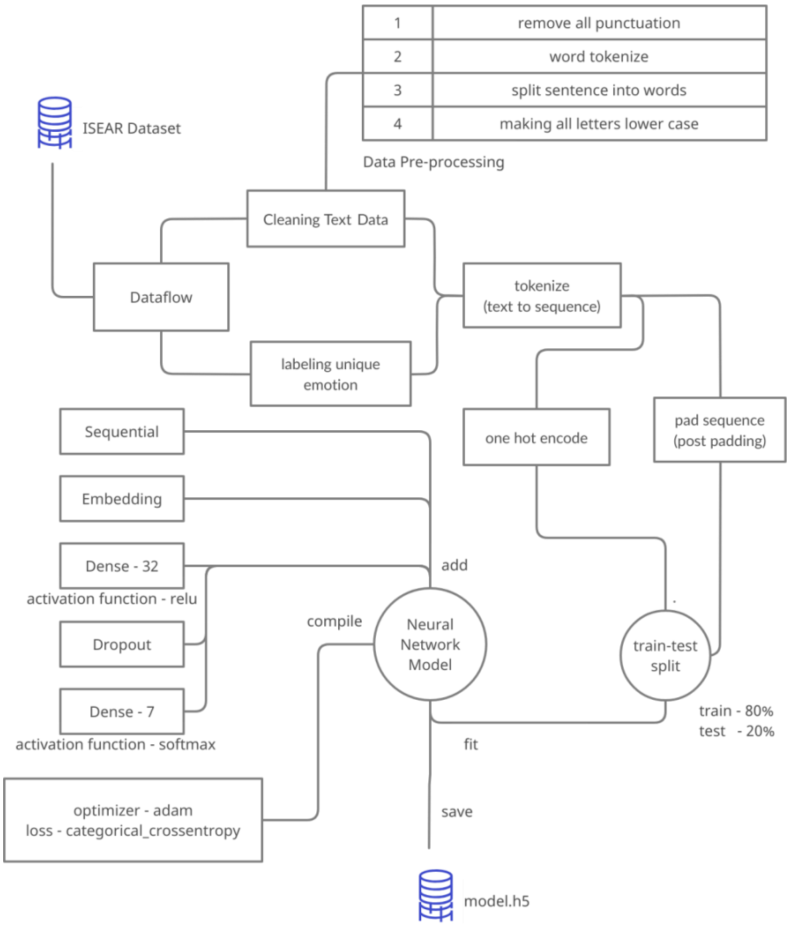


Figure 4. Training Work Flow for Emotion Detection

4.2.2. Method

For the analysis, we have used Bi-GRU and Bi-LSTM and did some comparison which works best for our analysis. We used ReLU (Rectified Linear Unit) and softmax as our activation function. Finally, we have set the optimizer to Adam for accurate results. Moreover, to save the model so that it can be loaded later we used the model checkpoint call-backs of Keras call-back library. Checkpointing the neural network model is essential as checkpoints are the weight of the model and these weights can be used to make better predictions. The liveloss plot tool was applied so that we could get the accuracy-loss plot. We took a batch size of 32 and epoch was kept at 100 for both the models so that the algorithm sees the entire dataset 100 times.

4.2.3. Result

Table 3 shows the accuracy and loss results between Bi-LSTM and Bi-GRU. Comparing the results, we can say Bi-GRU gives the highest training accuracy of 93%.

Table 3. Accuracy for training and validation data of Deep-Learning Models for EmotionDetection

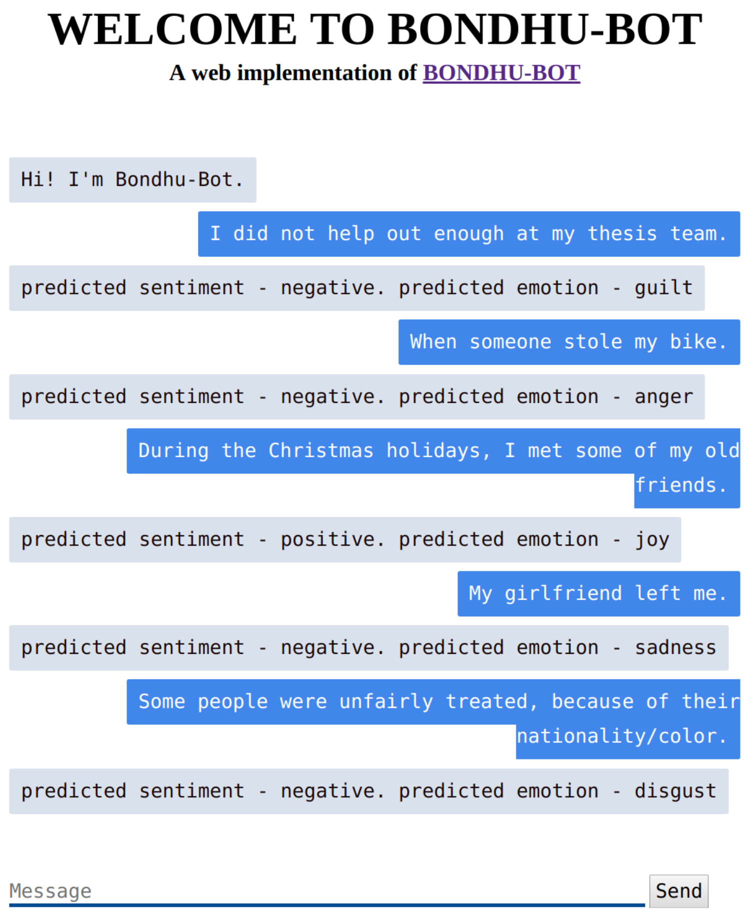
Model	**Training**		**Validation**	
	loss	accuracy	loss	accuracy
Bi-directional LSTM	0.350	0.875	1.602	0.684
Bi-directional GRU	0.173	0.936	1.556	0.437

4.2.4. Performance analysis

In Table 4, we showed the background values of how an input sentence is classified on the basis of confidence level for each emotion. The emotion with the highest confidence level will be the output. Moreover, few examples with corresponding outcomes are given in Figure 5. Despite the exact words from emotion categories are not directly mentioned in the text yet we can see our model was able to accurately identify the accurate state of mind. For example- it is a matter of “guilt” if I am not being to help my thesis team as it is my responsibility.

Table 4. Predicted confidence level of different emotions

Text: I did not help my thesis team enough.	
Emotion	**confidence**
guilt	0.49738222
shame	0.36354083
anger	0.05595107
sadness	0.037771154
fear	0.022115987
disgust	0.015596252
joy	0.0076424023
predicted result	**guilt**



WELCOME TO BONDHU-BOT

A web implementation of BONDHU-BOT

Hi! I'm Bondhu-Bot.

I did not help out enough at my thesis team.

predicted sentiment - negative. predicted emotion - guilt

When someone stole my bike.

predicted sentiment - negative. predicted emotion - anger

During the Christmas holidays, I met some of my old friends.

predicted sentiment - positive. predicted emotion - joy

My girlfriend left me.

predicted sentiment - negative. predicted emotion - sadness

Some people were unfairly treated, because of their nationality/color.

predicted sentiment - negative. predicted emotion - disgust

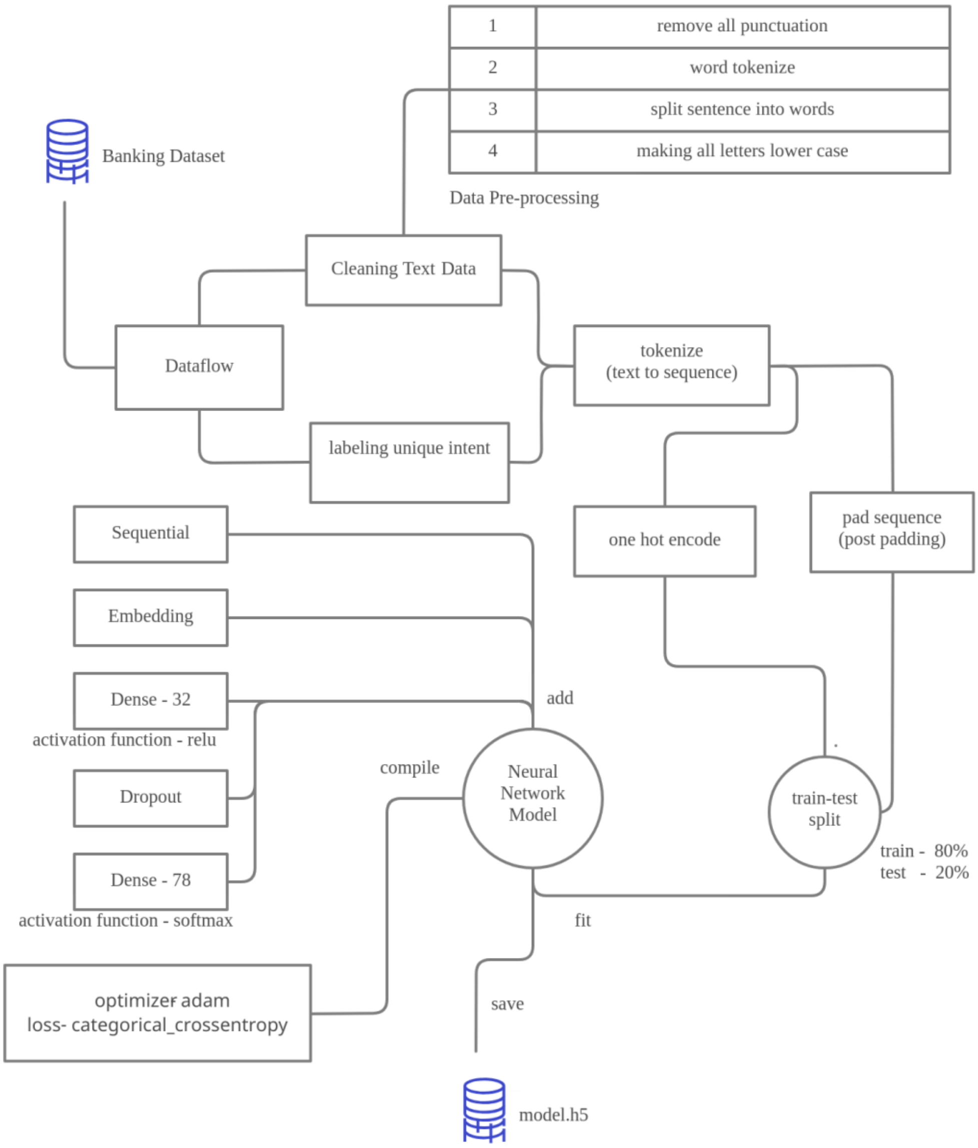
Figure 5. Application of Emotion Analysis with Sentiment

4.3. Intent Classification

Despite we got very accurate results through emotional analysis using Bi-directional GRU and Bi-directional LSTM, but for making the chatbot responses more human-like, knowing the intent of the message is essential. By emotion detection we will get the state of mind of the user, but *"why is he in that particular state?"* will be identified by knowing the intent behind it. Besides, knowing the context of a person's text helps to gather various information. As conversational agents are being used for various purposes like customer support, e-commerce and even in the field of mental health, determining the intention of a user has become very essential.

4.3.1. Data Processing

We used the dataset banking-77 which is composed of online banking queries and 78 corresponding intents [17]. After cleaning and refining the dataset, stemming was applied in the similar process like emotional analysis. Lemmatization groups all the similar words such as 'happy', 'happier', 'happiest' to get the ultimate source word even if it is written differently. Like before the One-Hot encoder was used to convert the categorical into numerical form. Similar tools used for one-hot encoder in emotion detection were also used in intent classification. Figure 6 describes the steps we followed for data processing.



1	remove all punctuation
2	word tokenize
3	split sentence into words
4	making all letters lower case

Figure 6. Training Work Flow for Intent Classification

4.3.2. Method

We split our dataset into 80% of training dataset and 20% validation. Now, our data is ready to be fed in the model. We used Bi-GRU and Bi-LSTM, as it creates a reverse copy of the sentencesand reads from both the sides to keep both past and future context of sentence into consideration. All the parameters like activation functions, optimizer, batch size and epoch were set just like our previous analysis.

4.3.3. Result

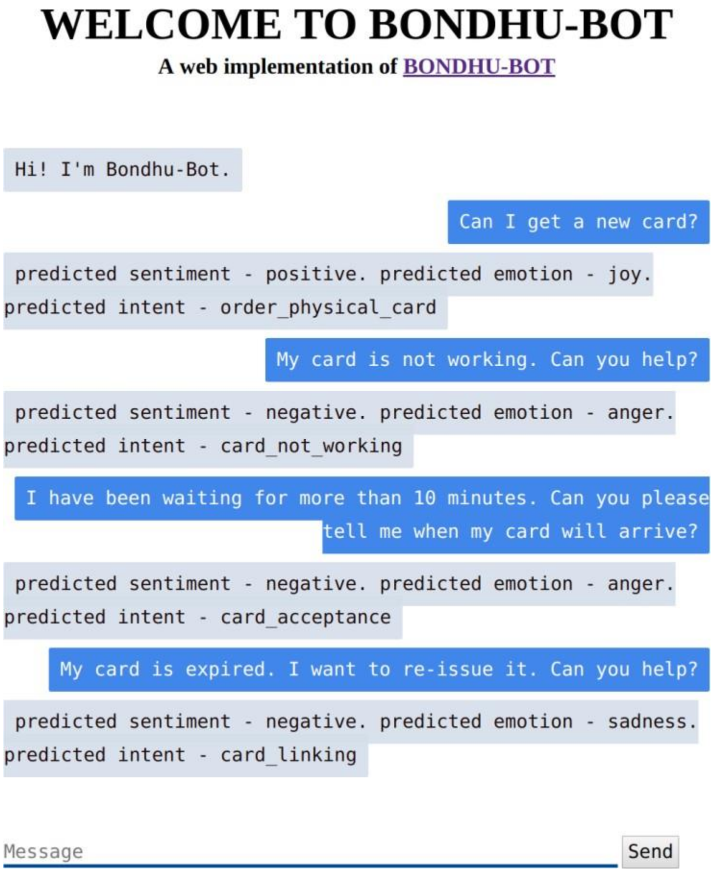
From Table 6 we got the training accuracy of around 80.7% using Bi-GRU and 77.4% accuracy using Bi-LSTM which indicates a good comparison between the two models. However, Bi-GRU gives comparatively better results for intent classification.

Table 6. Accuracy for training and validation data of different Deep-Learning Models for Intent Classification

Model	Training		Validation	
	loss	accuracy	loss	accuracy
Bi-directional LSTM	0.651	0.774	1.337	0.711
Bi-directional GRU	0.562	0.807	1.468	0.694

4.3.4. Performance analysis

Just as before, we gave some sentences as input for real time prediction. We created a get_intent class that gives the possible probabilities of similarities of the sentence with the intent categories. The intent that gives the max probability is the resultant outcome. As our intent dataset is related to bank queries, our input sentences in Figure 7 are different statements and questions regarding bank issues. In our second example we can see the user's card isn't working and our bot predicted *"card_not_working"* correctly. Similarly, the intents of the following examples were identified without any mistake.



WELCOME TO BONDHU-BOT

A web implementation of BONDHU-BOT

Hi! I'm Bondhu-Bot.

Can I get a new card?

predicted sentiment - positive. predicted emotion - joy.
predicted intent - order_physical_card

My card is not working. Can you help?

predicted sentiment - negative. predicted emotion - anger.
predicted intent - card_not_working

I have been waiting for more than 10 minutes. Can you please tell me when my card will arrive?

predicted sentiment - negative. predicted emotion - anger.
predicted intent - card_acceptance

My card is expired. I want to re-issue it. Can you help?

predicted sentiment - negative. predicted emotion - sadness.
predicted intent - card_linking

Figure 7. Application of Intent Analysis with Sentiment and Emotion

4.4. Named-Entity Recognition

Now, our next work is to get details of the intent. By *'details'* we mean about whom or where or at which date and time the context of the sentence is referring to. It is important because entities such as person, location, organization, time etc. help to gather information related to the intent of the text. This kind of task is achieved by named-entity recognition. NER is a subtask in information extraction and machine translation and also various DRNN (Deep Recurrent Neural Network) models along with word embedding are applied to perform NER [18]. With the help of NER, we can identify and categorize key entities in text which will help our chatbots to become more interactive.

4.4.1. Data Processing

The dataset we used is Annotated Corpus for NER using GMB (Groningen Meaning Bank) corpus which is tagged and built for predicting entities such as name location, time etc[19]. Next, for processing the data, we have retrieved sentences and corresponding tags. As displayed in Figure8 mappings between sentences and tags were defined by converting the words and tags to indexes.

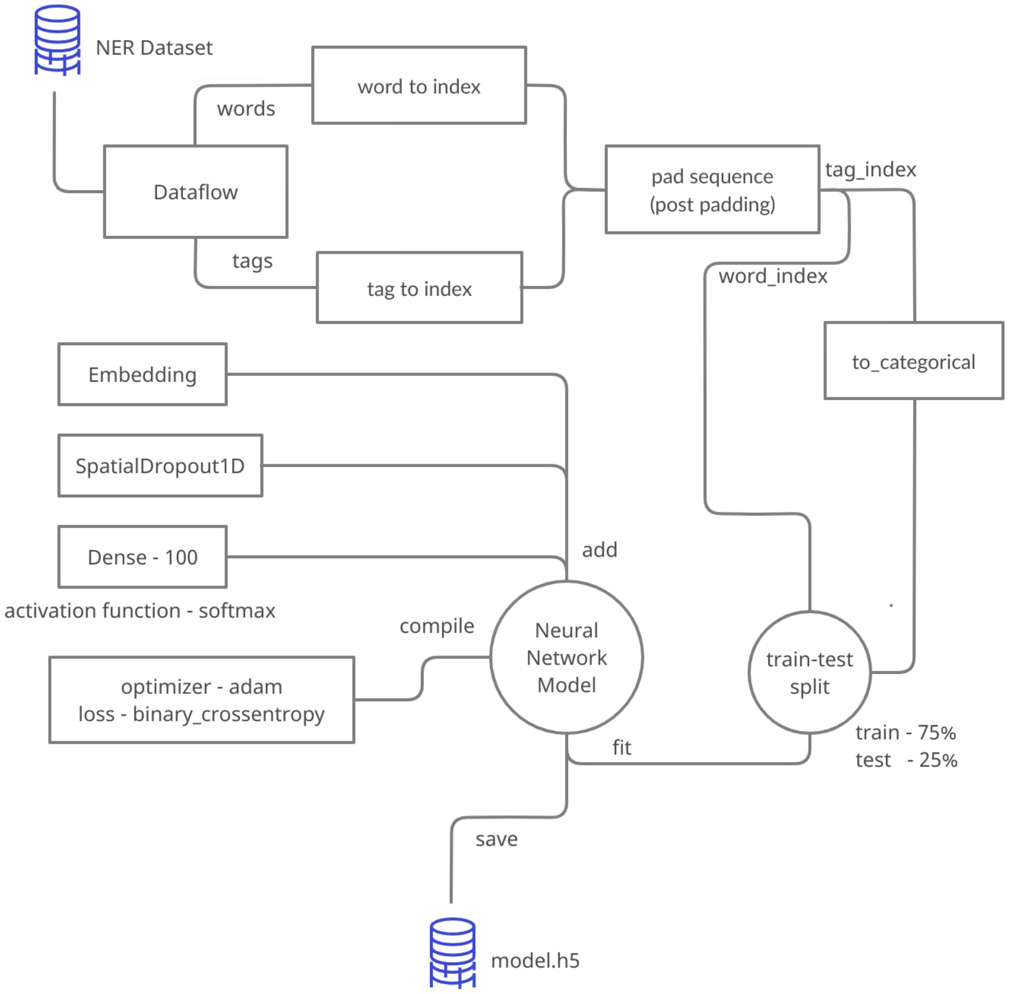


Figure 8. Training Work Flow for Named-entity Recognition

4.4.2. Method

Now, we have created train and test splits so that we can estimate the performance of algorithms when they are used to make predictions on data. The training data was split to 90% and test data was split to 10%. Our next task was to build and compile bi-directional LSTM. We have applied Spatial dropout 1D to 0.1, to drop the entire 1D feature map. We also have kept the dense to 100 in our model. For training the model properly, we have used the ModelCheckpoint and livelossplot. The batch size is set to 32 and epoch set to 3. Finally, we have done the training and evaluation of the model.

4.4.3. Result

After training the model, we got a training accuracy of 98.9% using both Bi-LSTM and Bi-GRU, given in Table 8, which plays a great role in helping chatbot to identify the entities and categorize them.

Table 8. Accuracy for training and validation data of different Deep-Learning Models forNamed-Entity Recognition

Model	**Training**		**Validation**	
	loss	accuracy	loss	accuracy
Bi-directional LSTM	0.039	0.988	0.048	0.985
Bi-directional GRU	0.034	0.989	0.045	0.986

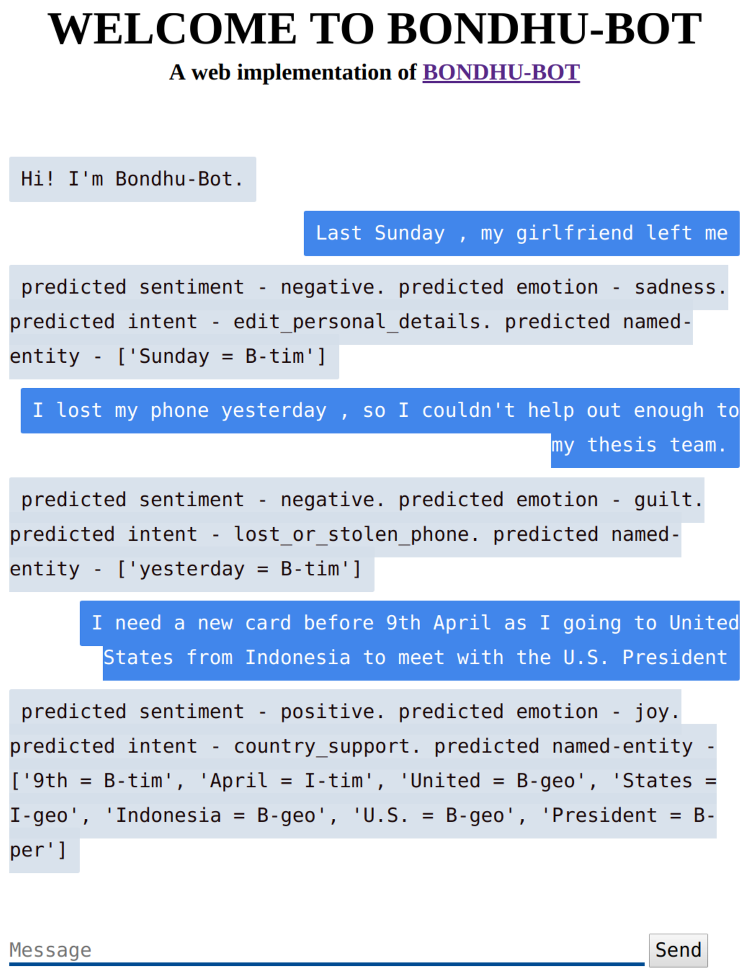
4.4.4. Performance analysis

For real time predictions, we gave some sentences as input an example is shown in Table 9. In our input sentence *'George'* is the name of the person, *'London'* and *'Indonesia'* are the places he will be travelling within and *'sunday morning'* is the date and time of travel. All these entities were identified by our chatbot accurately.

Table 9. Predicted Named-entity for a sentence

Text: George will go to London from Indonesia Sunday morning.	
Entity	**Predicted Tag**
George	I-per
London	B-geo
Indonesia	B-geo
Sunday	B-tim
Morning	I-tim

Finally, in Figure 9, we incorporated all the features analysed above. We used variations in input text which in return gave us different sentiments, emotion, intent and entity. Considering the intent dataset, we needed to set examples confined to banking. However, our first input was nowhere related to bank issues but surprisingly it gives *"edit_personal_details"* which was not as bad as we had expected for this particular text. In addition, all other components for the given text were accurately identified such as *'sad'* emotion, *'negative'* sentiment and B-time *'Sunday'*. The second input was *"I lost my phone yesterday, so I could not help out enough to my thesis team"* which is a complex sentence. But our chatbot could predict all the components of the complex sentence and predicted sentiment as *'negative'*, emotion as *'guilt'*, intent as *"lost_or_stolen phone"* and entity yesterday as *'time'*. Next, we gave a bit more complex sentence which had multiple entities and it predicted all four entities. The examples also show us that our bot is now completely prepared to understand a user's given dialogue. This can be used in different sectors for understanding queries or statements of a chatbot user simply by changing the intent dataset to the desired intent domain.



WELCOME TO BONDHU-BOT

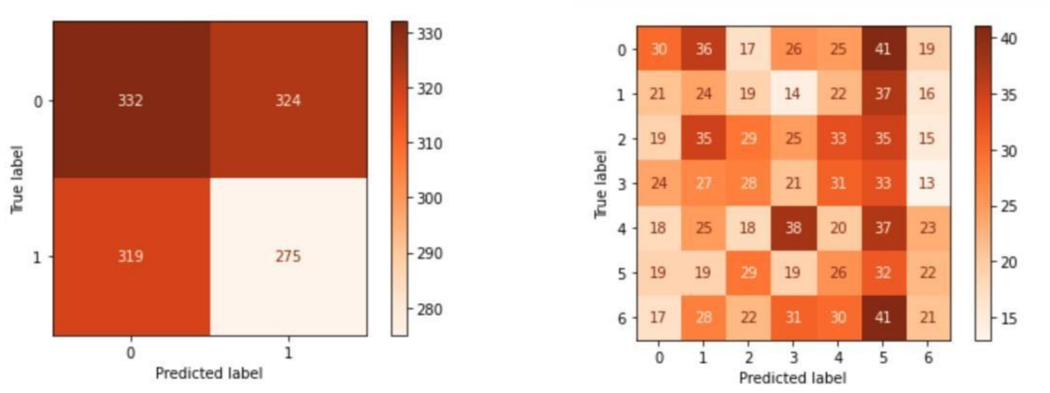
A web implementation of BONDHU-BOT

```
Hi! I'm Bondhu-Bot.

                                  Last Sunday , my girlfriend left me

predicted sentiment - negative. predicted emotion - sadness.
predicted intent - edit_personal_details. predicted named-
entity - ['Sunday = B-tim']

 I lost my phone yesterday , so I couldn't help out enough to
                                               my thesis team.

predicted sentiment - negative. predicted emotion - guilt.
predicted intent - lost_or_stolen_phone. predicted named-
entity - ['yesterday = B-tim']

      I need a new card before 9th April as I going to United
        States from Indonesia to meet with the U.S. President

predicted sentiment - positive. predicted emotion - joy.
predicted intent - country_support. predicted named-entity -
['9th = B-tim', 'April = I-tim', 'United = B-geo', 'States =
I-geo', 'Indonesia = B-geo', 'U.S. = B-geo', 'President = B-
per']
```

Figure 9. Application of Name-Entity Analysis with Sentiment, Emotion and Intent

5. Comparison

Initially, we used a machine learning algorithm - **Support Vector Machine**, to evaluate performance on sentiment analysis and emotion detection. The reason to use SVM was its mathematical foundation in statistical learning theory that made it popular in the field of machine learning and supervised classification [20]. Moreover, it was also used to recognize, interpret and process human emotion from text according to predefined emotion classes [21] Figure 10 shows results from sentiment analysis and emotional detection with SVM were not very satisfactory. It gave only 48.5% accuracy for sentiment analysis and only 14.16% for emotion detection where else all our experimental accuracy using deep learning varied between 74% to 98%. Thus, from the results we can deduce our deep learning approach outperformed ML algorithm by a significant percentage.



(a) confusion matrix for sentiment analysis

(b) confusion matrix for emotion detection

Figure 10. Performance evaluation using Support Vector Machine

6. CONCLUSION

Throughout our research, we extracted different components from texts that will help a system to have adequate information about the user's opinion or statement. This paper proposes that for a chatbot to appropriately respond to users it needs to identify the important keywords to capture the expressed sentiments, emotions as well as the intention, behind the uttered dialogue of the user in depth. We analysed all the major components through application of deep learning techniques and achieved our goals in making accurate predictions in each segment of work. However, through our work we understood that it is very difficult to build a chatbot that can mimic a human and make conversations interactive. The detailed analysis incorporating all the functions demonstrates that our proposed model will significantly help to build a human-like chatbot with in numerous applications.

7. FUTURE WORK

As now our bot is already able to acknowledge the personalised words and emotions, in future anyone including us can be able to build a complete responsive chatbot using this which can communicate in a more humanistic manner. If the chatbot is trained with a dialogue corpus dataset of any specific domain, it will be able to respond referring to all the analysed features we worked on in this research paper. All it needs is to add a dialogue generation feature in our model so thatit can generate a response automatically to the given input text by the user. To solve the unavailability of appropriate dataset and to make the chatbot more human-like, we have the plan to extend our approach to unsupervised learning in the future. And then, our bot will be able to train itself on previous history of conversations and infer the dependencies between input texts.

REFERENCES

[1] Solutions, R. I. (2018, April 13). Top 10 Reasons: Why Your Business Need A Chatbot Development.Retrieved January 4, 2021, from https://chatbotsjournal.com/top-10-reasons-why-your-business- need-a-chatbot-development-5a53760da1b6?gi=491b46b2ea69

[2] Advances in Conversational AI. (n.d.). Retrieved January 4, 2021, from https://ai.facebook.com/blog/advances-in-conversational-ai/

[3] Deep Learning vs Machine Learning: Know the Difference. Retrieved January 4, 2021, from https://www.mygreatlearning.com/blog/is-deep-learning-better-than-machine-learning

[4] Kawade, D. R., & Oza, K. S. (2017). Sentiment analysis: machine learning approach. Int J Eng Technol, 9(3), 2183-2186.

[5] Hossain, Y., Hossain, I. A., Banik, M., & Chakrabarty, A. (2018, June). Embedded system based Bangla intelligent social virtual robot with sentiment analysis. In 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR) (pp. 322-327). IEEE.

[6] Lee, D., Oh, K. J., & Choi, H. J. (2017, February). The chatbot feels you-a counseling service using emotional response generation. In 2017 IEEE international conference on big data and smart computing (BigComp) (pp. 437-440). IEEE.

[7] Bhagwat, V. A. (2018). Deep Learning for Chatbots.

[8] Acheampong, F. A., Wenyu, C., & Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. Engineering Reports, 2(7), e12189.

[9] Balaji, M., & Yuvaraj, N. (2019, November). Intelligent Chatbot Model to Enhance the Emotion Detection in social media using Bi-directional Recurrent Neural Network. In Journal of Physics: Conference Series (Vol. 1362, No. 1, p. 012039). IOP Publishing.

[10] Yadav, V., & Bethard, S. (2019). A survey on recent advances in named entity recognition from deep learning models. arXiv preprint arXiv:1910.11470.

[11] Meng, L., & Huang, M. (2017, November). Dialogue intent classification with long short-term memory networks. In National CCF Conference on Natural Language Processing and Chinese Computing (pp. 42-50). Springer, Cham.

[12] Kharde, V., & Sonawane, P. (2016). Sentiment analysis of twitter data: a survey of techniques. arXiv preprint arXiv:1601.06971.

[13] Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies (pp. 142-150).

[14] David. (2017, November 30). Twitter_sentiment [Data Set]. Retrieved from https://www.kaggle.com/ywang311/twitter-sentiment

[15] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[16] Scherer, K. R., & Wallbott, H. G. (1994). Evidence for universality and cultural variation of differential emotion response patterning. Journal of personality and social psychology, 66(2),310.

[17] Casanueva, I., Temčinas, T., Gerz, D., Henderson, M., & Vulić, I. (2020). Efficient intent detection with dual sentence encoders. arXiv preprint arXiv:2003.04807.

[18] Khan, W., Daud, A., Alotaibi, F., Aljohani, N., & Arafat, S. (2020). Deep recurrent neural networks with word embeddings for Urdu named entity recognition. ETRI Journal, 42(1), 90-100.

[19] Walia, A. (2017, September 21). Annotated Corpus for Named Entity Recognition [Data Set]. Retrieved from https://www.kaggle.com/abhinavwalia95/entity-annotated-corpus

[20] Awad, M., & Khanna, R. (2015). Support vector machines for classification. In Efficient Learning Machines (pp. 39-66). Apress, Berkeley, CA.

[21] Kirange, D. K., & Deshmukh, R. R. (2012). Emotion classification of news headlines using SVM. Asian Journal of Computer Science and Information Technology, 5(2), 104-106.

MORPHOLOGICAL ANALYSIS OF JAPANESE HIRAGANA SENTENCES USING THE BI-LSTM CRF MODEL

Jun Izutsu[1] and Kanako Komiya[2]

[1]Ibaraki University, Japan
[2]Tokyo University of Agriculture and Technology, Japan

ABSTRACT

This study proposes a method to develop neural models of the morphological analyzer for Japanese Hiragana sentences using the Bi-LSTM CRF model. Morphological analysis is a technique that divides text data into words and assigns information such as parts of speech. This technique plays an essential role in downstream applications in Japanese natural language processing systems because the Japanese language does not have word delimiters between words. Hiragana is a type of Japanese phonogramic characters, which is used for texts for children or people who cannot read Chinese characters. Morphological analysis of Hiragana sentences is more difficult than that of ordinary Japanese sentences because there is less information for dividing. For morphological analysis of Hiragana sentences, we demonstrated the effectiveness of fine-tuning using a model based on ordinary Japanese text and examined the influence of training data on texts of various genres.

KEYWORDS

Morphological analysis, Hiragana texts, Bi-LSTM CRF model, Fine-tuning, Domain adaptation.

1. INTRODUCTION

1.1. Components of the Japanese Language and the Acquisition Process

Japanese sentences contain various kinds of character, such as Kanji (Chinese character), Hiragana, Katakana, numbers, and alphabet, making it difficult to learn. Japanese speakers usually learn Hiragana first in their school days because the number of characters is much smaller than the Kanji; Hiragana has 46 characters, and Japanese use thousands of Kanji. Most Japanese sentences are composed of all kinds of characters and are called Kanji-Kana mixed sentences.

However, it is difficult for many non-Japanese speakers to learn thousands of Kanji, so children and new Japanese language learners use Hiragana.

1.2. Morphological Analysis of Hiragana Sentences

Morphological analysis is a technique that divides natural language text data into words and assign information such as parts of speech. In Japanese, morphological analysis is one of the core technologies for natural language processing because the Japanese language does not have word delimiters between words. Morphological analyzers like MeCab[1], Chasen and Juman++[2,3]

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 123-135, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112310

are now commonly used for morphological analysis. However, since the above systems target Kanji-Kana mixed sentences, it is challenging to perform morphological analysis using only Hiragana sentences.

Morphological analysis of Hiragana-only sentences is more challenging than morphological analysis of Kanji-Kana mixed sentences. If the text is a mixture of Kanji and Kana (Hiragana and Katakana), it will be divisible between Kanji, Hiragana, and Katakana. However, if the text consists only of Hiragana, there will be less information for dividing. Therefore, we propose to fine-tune the model of Kanji-Kana mixed sentences and investigated whether the accuracy of morphological analysis of Hiragana sentences can be improved by inheriting the information to be divided into words.

The Bi-LSTM CRF model was used to develop a morphological analyzer for Hiragana sentences in this paper. We used two types of training data: Wikipedia and Yahoo! Answers in the Balanced Corpus of Contemporary Written Japanese, in our studies to investigate the influence of the genre of the training data. We also fine-tune both data to examine the effect of various genres on the text. [4]

The following are the four contributions of this paper.

- Developed a morphological analyzer for Hiragana sentences using the Bi-LSTM CRF model,
- Demonstrated the effectiveness of fine-tuning using a model based on Kanji-Kana mixed text,
- Examined the influence of training data of morphological analysis on texts of various genres, and
- Demonstrated the effectiveness of fine-tuning using data from Wikipedia and Yahoo! Answers.

In this paper, we report the results of these experiments.

2. RELATED WORK

Izutsu et al. (2020) [5] converted MeCab'sipadic dictionary into Hiragana and performed morphological analysis on Hiragana sentences using a corpus consisting only of Hiragana. To our knowledge, only this work developed a morphological analyzer for only Hiragana sentences.We also developed a morphological analyzer for only Hiragana sentence, but we try to achieve this goal without any dictionary.

There are some studies focus on morphological analyzers for Hiragana-highly-mixed sentences and most of them treated a lot of Hiragana words as noises or broken Japanese. For example, Kudo et al. (2012) [6] used generative model to model the process of generating Hiragana noise-mixed sentence. They proposed using a large-scale web corpus and EM algorithm to estimate the model' s parameters to improve the analysis of Hiragana noise-mixed sentences. Osaki et al. (2016) [7] constructed a corpus for broken Japanese morphological analysis. They defined new parts of speech and used them for broken expressions. Fujita et al. (2014) [8] proposed an unsupervised domain adaptation technique that uses the existing dictionaries and labeled data to build a morphological analyzer by automatically transforming them for the features of the target domain. Hayashi and Yamamura (2017) [9] reported that adding Hiragana words to the dictionary can improve the accuracy of morphological analysis.

We used Bi-LSTM model to develop a morphological analyzer. Ma et al. (2018) [10] developed a word segmentation model for Chinese using the Bi-LSTM model.They reported that word segmentation accuracy achieved better accuracy on public datasets than the Bi-LSTM model, compared to models based on more complex neural network architectures.

Also, Thattianaphanich and Prom-on (2019) [11] developed the Bi-LSMT CRF model and performed named entity recognition extraction in Thai. In Thai, there are linguistic problems such as lack of linguistic resources and boundary indices between words, phrases, and sentences. Therefore, they prepared word representations and learned text sequences using Bi-LSTM and CRF to address these problems.

In work on Japanese morphological analyzers, Tolmachev et al. created a morphological model using neural networks and semi-supervised learning, and showed that their method performed comparably to traditional dictionary-based state-of-the-art methods, and could even outperform them when trained on a combination of human-annotated and automatically annotated data [12].

In addition, Chen et al. created an LSTM-based neural network model for Chinese word segmentation [13]. Although most of the current state-of-the-art methods for Chinese word segmentation are based on supervised learning, they report that their LSTM-based model outperforms traditional neural network models and state-of-the-art methods.

3. METHODS

In this research, we used the Bi-LSTM CRF model to generate a morphological analyzer for Hiragana sentences.We trained and compared the following five models for Hiragana sentences in our studies.

1. the Hiragana Wiki model
2. the Hiragana Yahoo! model
3. the Kanji-Kana Wiki+ Hiragana Wiki model
4. the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model
5. the Hiragana Wiki+ Hiragana Yahoo! model

The generation processes of the five models for Hiragana sentences are shown in Figure 1.

3.1. Hiragana Wiki Model

The Hiragana Wiki model is the model generated in training B (Figure 1). We used the data from Wikipedia, where Kanji-Kana mixed sentences are converted to Hiragana, for training data. Hiragana is a phonogram, and Kanji could be converted into Hiragana according to its pronunciation. We used MeCab's reading data as a pseudo-correct answer for the conversion because Wikipedia does not have Hiragana-only data. Here, UniDic was used as a dictionary of MeCab. [14]

3.2. Hiragana Yahoo! Mode

The model generated in training D (Figure 1) is the Hiragana Yahoo! model. We used reading data from Yahoo! Answers as training data. We compared its accuracy to a Hiragana Wiki + Hiragana Yahoo! model (training F in Figure 1) and Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model (training E in Figure 1).

3.3. Kanji-Kana Wiki+ Hiragana Wiki Model

In this paper, we proposed fine-tuning using a model with Kanji-Kana mixed sentences. The Kanji-Kana Wiki+ Hiragana Wiki model is generated using the original Wikipedia data and Hiragana Wikipedia data, which are the actual data converted into only Hiragana. We fine-tuned Kanji-Kana Wiki model, the model trained by the original Wikipedia data (training A in Figure 1), with Hiragana Wikipedia data, which is Wikipedia data automatically converted into Hiragana sentences (training C in Figure 1). Kanji-Kana mixed sentences contain many clues for morphological analysis, such as borderline between Kanji and Hiragana and information about Kanji. Therefore, it is expected to improve accuracy. By comparing the Hiragana Wiki model and this model, we can assess the effectiveness of fine-tuning based on Kanji-Kana mixed Wikipedia data, when we only have Wikipedia data.

3.4. Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! Model

Training E in Figure 1 generates the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model. To improve accuracy, we used both Wikipedia and Yahoo! Answers as training data.

We fine-tuned the Kanji-Kana Wiki+ Hiragana Wiki model with Hiragana Yahoo! Answers data for this model.

3.5. Hiragana Wiki+ Hiragana Yahoo! Model

The Hiragana Wiki+ Hiragana Yahoo! model is the model generated in training F in Figure 1. As training data, we used Hiragana Wikipedia data and Hiragana Yahoo! Answers data. We fine-tuned the Hiragana Wiki model with Hiragana Yahoo! Answers.

We can see the effectiveness of the Kanji-Kana Wiki+ Hiragana Wiki model, i.e., how much the Kanji-Kana mixed model affects this model, when we have both Wikipedia data and Yahoo! Answers by comparing the accuracy of this model to Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model.

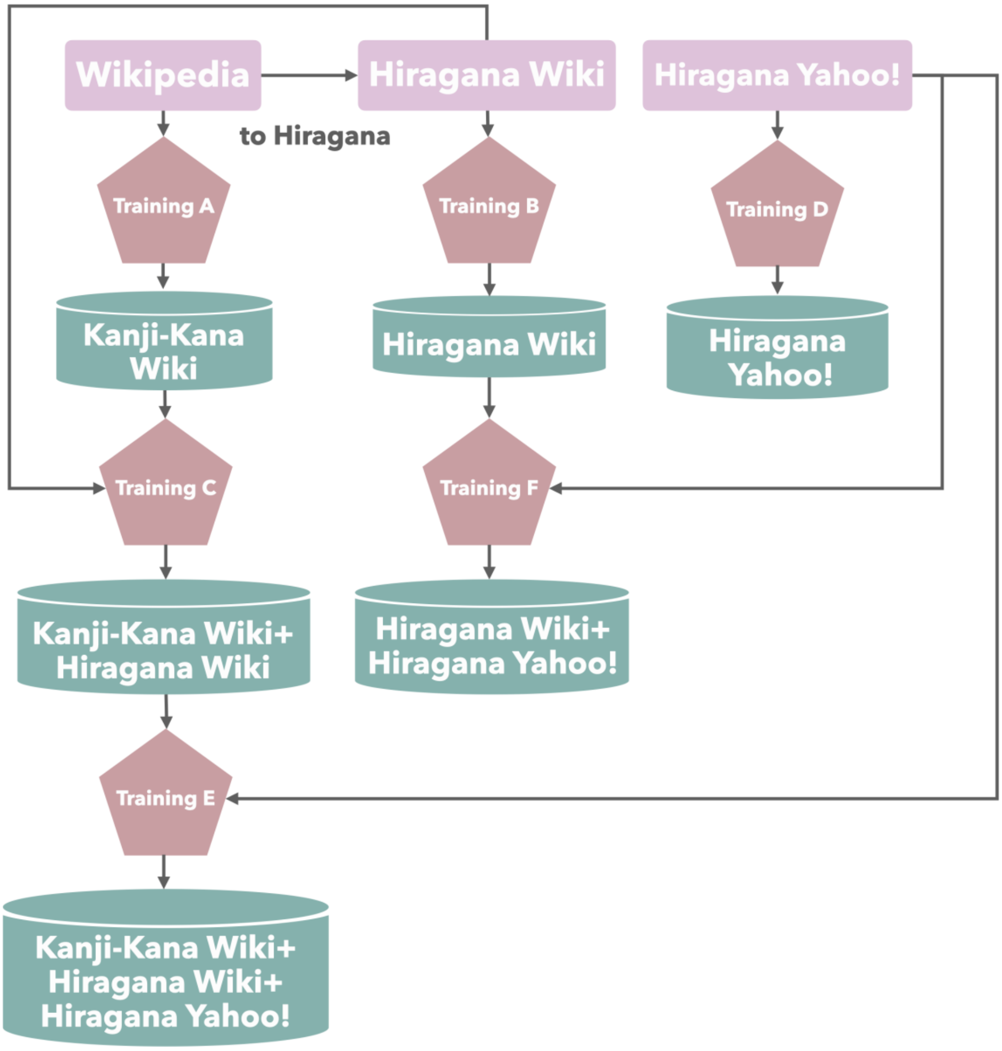


Figure 1. Model generation processes of the five models for Hiragana sentences

4. EXPERIMENT

4.1. Network Architecture

We developed Bi-LSTM CRF models using Bi-LSTM CRF module of Pytorch.

Table 1 summarizes the hyperparameters used in this experiment. The dimensions for the network architecture were determined according to the preliminary experiments.

Table 1. Hyperparameters used in the experiment

EMBEDDING_DIM	5
HIDDEN_DIM	4
Lr	0.01
Weight decay	1e-4
Optimizer	sgd
Epoch number	15

The tag size is 38 for all models and the vocabulary size varies according to the model because it is the total character type number of the training corpus. The vocabulary size of the models that use only Hiragana sentences are 192 and that of the model that uses Kanji-Kana Wiki data was 2,742.

4.2. Training Data

We used two kinds of training data in this experiment: Japanese Wikipedia data and "Yahoo Answers Data" from BCCWJ of the National Institute for Japanese Language and Linguistics. For this experiment, the data from Wikipedia were extracted from

jawiki-latest-pages-articles.xml.bz2

which is published on the website for this experiment.

We preprocessed the Wikipedia data before training to obtain Hiragana-only sentences. The preprocessing procedures are as follows. First, we conducted morphological analysis on the Wikipedia data using MeCab. The output of MeCab has features: word segmentation, part of speech, part of speech subdivision 1, part of speech subdivision 2, part of speech subdivision 3, conjugation type, conjugated and basic forms, reading, and pronunciation.

Table 2. MeCab's analysis result for "僕は君と遊ぶ"

	僕 *(Boku)*	は *(Ha)*	君 *(Kimi)*	と *(To)*	遊ぶ *(Asobu)*
Part of speech	Noun	Particle	Noun	Particle	Verb
Part-of-speech subdivision 1	Pronoun	Binding particle	pronoun	Case particle	Non-bound
Part-of-speech subdivision 2	General	*	General	General	*
Part-of-speech subdivision 3	*	*	*	*	*
Conjugation type	*	*	*	*	Godan_verb_ba_column
Conjugated form	*	*	*	*	Basic form
Basic form	僕 *(Boku)*	は *(Ha)*	君 *(Kimi)*	と *(To)*	遊ぶ *(Asobu)*
Reading	ボク *(Boku)*	ハ *(Ha)*	キミ *(Kimi)*	ト *(To)*	アソブ *(Asobu)*
Pronunciation	ボク *(Boku)*	ワ *(Wa)*	キミ *(Kimi)*	ト *(To)*	アソブ *(Asobu)*

Let us take the example of the sentence, "僕は君と遊ぶ" (*Bokuwakimi to asobu*). Table 2 shows the output result of MeCab when we input this example sentence. This is a Kanji-Kana mixed sentence, which means "I play with you." If this sentence is expressed entirely in Hiragana, that would be "ぼくはきみとあそぶ. "

We obtained Hiragana data by replacing the surface forms of words with their readings. Next, we split the Hiragana data into individual characters and assign a part-of-speech tag to each of them.

Here, B-{Part-of-Speech} is assigned to the first Hiragana character, and I-{Part-of-Speech} is assigned to the following Hiragana characters, if the Kanji consisted of more than two syllables.

By formatting this Hiragana sentence as described above, we can obtain the character data and tags corresponding to the character data with one-to-one correspondence (Table 3).

Table 3. Example of splitting "ぼくはきみとあそぶ"

ぼ *(bo)*	く *(ku)*	は *(ha)*	き *(ki)*	み *(mi)*	と *(to)*	あ *(a)*	そ *(so)*	ぶ *(bu)*
B-Noun	I-Noun	B-Particle	B-Noun	I-Noun	B-Particle	B-Verb	I- Verb	I-Verb

Please note that Japanese Kanji characters often have more than one reading. For example, "君" could be Kimi or Kun in Japanese. Therefore, sometimes this conversion makes some errors. The number of characters in the training data is 1,183,624.

We used the original Japanese Wikipedia data for the model that uses Kanji-Kana mixed sentences. Table 4 shows the characters and their tags of the Kanji-Kana mixed sentence, "僕は君と遊ぶ."

Table 4. Example of splitting "僕は君と遊ぶ"

僕 *(boku)*	は *(ha)*	君 *(kimi)*	と *(to)*	遊ぶ *(aso)*	ぶ *(bo)*
B-Noun	B-Particle	B-Noun	B-Particle	B-Verb	I-Verb

For Yahoo! Answers data, we extracted the reading data from the corpus.

Native Japanese speakers manually annotate them, but sometimes they could be different from the authors' intent. For example, "日本" (Japan in Japanese) has two readings, "Nihon" and "Nippon," and both of them are correct in most sentences. Therefore, in these cases, only the author can guarantee which one is correct. In BCCWJ, some words had a predefined reading to reduce the burden of annotators.

4.3. Test Data

We also used the BCCWJ for the test data.

The BCCWJ provides sub-corpora and we used 12 of them.

The number of characters used in this experiment for each dataset is shown in Table 5.

Table 5. Number of characters for each sub-corpus in BCCWJ

Dataset Name	**Number of characters**
Books (Library sub-corpus)	1,374,216
Bestsellers	1,093,860
Yahoo! Answers	830,960

Legal Documents	2,316,374
National Diet Minutes	2,050,400
PR Documents	2,151,126
Textbooks	956,927
Poems	466,878
Reports	2,546,307
Yahoo! Blogs	1,305,660
Books	1,281,251
Newspapers	1,301,728

The test data are tagged Hiragana characters with a one-to-one correspondence between the Hiragana character and the tag-based on reading and part-of-speech information, were created in the same way as the training data.

The data from “Yahoo! Answers" are also used as training data for creating the Hiragana Yahoo! model, the Hiragana Wiki + Hiragana Yahoo! model, and the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model, however we used different parts for the training and testing.

5. RESULT

According to genres of the test data, Tables 6 and 7 summarize the accuracies of the Hiragana Wiki model, Kanji-Kana Wiki+ Hiragana Wiki model, Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model, Hiragana Yahoo! model, and Hiragana Wiki+ Hiragana Yahoo! model. For the evaluation of each test dataset, we used macro and micro-averages of accuracy. Macro represents the macro-averaged accuracy, and micro represents the micro-averaged accuracy in Tables 6 and 7. In Table 6,blue numbers indicate that the accuracy of the Kanji-Kana Wiki+ Hiragana Wiki model or the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model was lower than that of the Hiragana Wiki model, and an asterisk indicates that this difference was significant according to a chi-square test for accuracy at the 5% significance level. In Table 7, Magenta numbers indicate that they are lower than the accuracy of the Kanji-Kana Wiki + Hiragana Wiki+ Hiragana Yahoo! model, and italics indicate that they are lower than the accuracy of the Hiragana Yahoo! model.An asterisk indicates that the difference was significant according to a chi-square test for accuracy at the 5% significance level. A plus indicates that the Hiragana Wiki+ Hiragana Yahoo! model was different from the Hiragana Yahoo! model according to a chi-square test for accuracy at the 5% significance level.The bold numbers are the best results of the models in Tables 6 and 7.

For the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model, the Hiragana Yahoo! model, and the Hiragana Wiki+ Hiragana Yahoo! model, the accuracies of Yahoo! Answers, are results of the closed test, and that is why they are written in parentheses.Also, the Yahoo! Answers evaluation data are removed from the macro and micro averages of all models. Therefore, they are average of 11 sub-corpora except for Yahoo! Answers data.

Table 6. Accuracy of Hiragana Wiki model, Kanji-Kana Wiki+ Hiragana Wiki model, and Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model according to each text genre.

	Hiragana Wiki	Kanji-Kana Wiki+ Hiragana Wiki	Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo!
Books (Library sub-corpus)	56.98	*57.15	*62.76
Bestsellers	50.14	*50.71	*60.48

Yahoo! Answers	48.32	*50.23	(*65.50)
Legal Documents	**63.39**	63.32	*60.55
National Diet Minutes	48.49	*49.59	*60.63
PR Documents	64.27	*63.93	*64.55
Textbooks	57.43	*58.01	*62.04
Poems	41.96	*42.45	*48.82
Reports	65.83	*65.57	*65.15
Yahoo! Blogs	57.12	57.10	*62.97
Books	55.73	55.77	*62.83
Newspapers	62.58	*62.30	*64.50
Macro	56.72	56.90	61.39
Micro	58.67	58.80	62.44

Blue numbers mean they are lower than the accuracy of the Hiragana Wiki model and an asterisk indicates that the model was different from the Hiragana Wiki model according to a chi-square test at the 5% level of significance. The bold numbers are the best results of the models.

Table 7. Accuracy of Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model, Hiragana Yahoo! model, and Hiragana Wiki+ Hiragana Yahoo! model according to each text genre.

	Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo!	Hiragana Yahoo!	Hiragana Wiki+ Hiragana Yahoo!
Books (Library sub-corpus)	62.76	*63.11	***+63.52**
Bestsellers	60.48	*60.16	***+61.32**
Yahoo! Answers	(65.50)	(*65.85)	***+66.38**
Legal Documents	60.55	*60.17	*+62.63
National Diet Minutes	60.63	*60.49	***+61.97**
PR Documents	64.55	***65.49**	_*+63.91_
Textbooks	62.04	61.97	***+62.66**
Poems	48.82	*47.61	***+49.55**
Reports	65.15	***66.89**	_*+64.89_
Yahoo! Blogs	62.97	***64.02**	_+62.90_
Books	62.83	*63.28	***63.36**
Newspapers	64.50	***65.54**	+64.51
Macro	61.39	61.70	61.93
Micro	62.44	62.93	63.01

Magenta numbers indicate that they are lower than the accuracy ofthe Kanji-Kana Wiki + Hiragana Wiki+ Hiragana Yahoo! model, and italics indicate that they are lower than the accuracy of the Hiragana Yahoo! model. An asterisk indicates that the model was different from the Kanji-Kana Wiki + Hiragana Wiki+ Hiragana Yahoo! model according to a chi-square test at the 5% significance level. A plus indicates that the model is different from Hiragana Yahoo! model at the 5% significance level. The bold numbers are the best results of the models.

We also evaluated the training data using MeCab. For the dictionary of MeCab, we used ipadic with the conversion of Kanji into Hiragana. The macro-averaged accuracy was 79.71%, and the micro-averaged accuracy was 80.10%.Please note that this result cannot simply be compared with results in Tables 6 and 7 because our system did not use a dictionary for the morphological analysis itself.

6. DISCUSSION

Table 6 shows that the Kanji-Kana Wiki+ Hiragana Wiki model's macro and micro-averaged accuracies (56.90% and 58.80%) are higher than those of the Hiragana Wiki model (56.72% and 58.67%). The macro-averaged accuracy of the Kanji-Kana Wiki+ Hiragana Wiki model improved by 0.18 points, while the micro-averaged accuracy by 0.13 points. This result indicates that the fine-tuning using the Kanji-Kana Wiki model is somehow effective.

Furthermore, the macro and micro-averaged accuracies of the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model (61.39% and 62.44%) are superior to those of the Kanji-Kana Wiki+ Hiragana Wiki model (56.90% and 58.80%). The Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model improved the accuracy by 4.49 points on the macro-averaged accuracy and by 3.64 points on the micro-averaged, indicating that further fine-tuning using Hiragana Yahoo! data considerably improve the permanence. Additionally, from Tables 6 and 7, we can confirm that the macro- and micro-averaged accuracies of the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo!, Hiragana Yahoo!, and Hiragana Wiki+ Hiragana Yahoo! models, the models using Yahoo! Answers as training data, are higher than those of Wikipedia and Kanji-Kana Wiki+ Hiragana Wiki models, the models using only Wikipedia as training data. In other words, the performance of the model is better when using Yahoo! Answers. We believe there could be two reasons for this result, the quality of the corpus and the similarity of the training and test data. The Hiragana Yahoo! data quality could be better than the Hiragana Wiki data because Hiragana Yahoo! data are manually annotated, whereas Hiragana Wiki data are automatically generated. Moreover, because the training and test data are both sub-corpora of BCCWJ, they can be more similar than when the training and test data are Wikipedia and BCCWJ.

Additionally, comparing the macro- and micro-averaged accuracies of the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model with those of the Hiragana Yahoo! model and the Hiragana Wiki+ Hiragana Yahoo! model in Table 7, we confirmed that the Hiragana Wiki+ Hiragana Yahoo! model is the best, and the Hiragana Yahoo! model is the second best, and the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model is the last. This result indicates that when the Hiragana Yahoo! data are available, the fine-tuning using both the Kanji-Kana Wiki model and the Hiragana Wiki model is not effective, although the fine-tuning using only the Hiragana Wiki model is useful. Notably, the accuracy of some types of test data is improved while others are decreased. The Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model was higher than the Hiragana Yahoo! model for the “Bestsellers," “Legal Documents," “National Diet Minutes," “Textbook," and “Poems", and it was higher than the Hiragana Wiki+ Hiragana Yahoo! model for “PR Documents", “Reports", and “Yahoo! Blogs" but there was no genre where the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model was the best in these three models.Therefore, we think that the reason why the Kanji-Kana Wiki model was useful when only the Hiragana Wiki data were available could be that the Hiragana Wiki data were automatically generated.

Now, let us discuss the difference among the genres of the texts. The genres where the accuracy of the Hiragana Wiki model was more than 60% were four genres: “Legal Documents," “PR Documents," “Reports," and “Newspapers."

We believe this is because the writing style of these test data is close to that of Wikipedia. Therefore, we marked these genres with underline in Tables 6 and 7. Additionally, we can see that, as for these genres, the fine-tuning is not useful. The accuracies of the Kanji-Kana Wiki+ Hiragana Wiki model decreased for these four genres, and those of the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model also decreased for two genres. As for the remaining two genres, the improvements are less than two points.

Figure 2 shows the effects or the differences of the accuracies of Hiragana Yahoo! data according to the genres of the texts. The blue line is its effects when the base model is the Kanji-Kana Wiki+ Hiragana Wiki model and the orange line is when the base model is the Hiragana Wiki model. This figure shows that the effects are the same even though the base models are different.

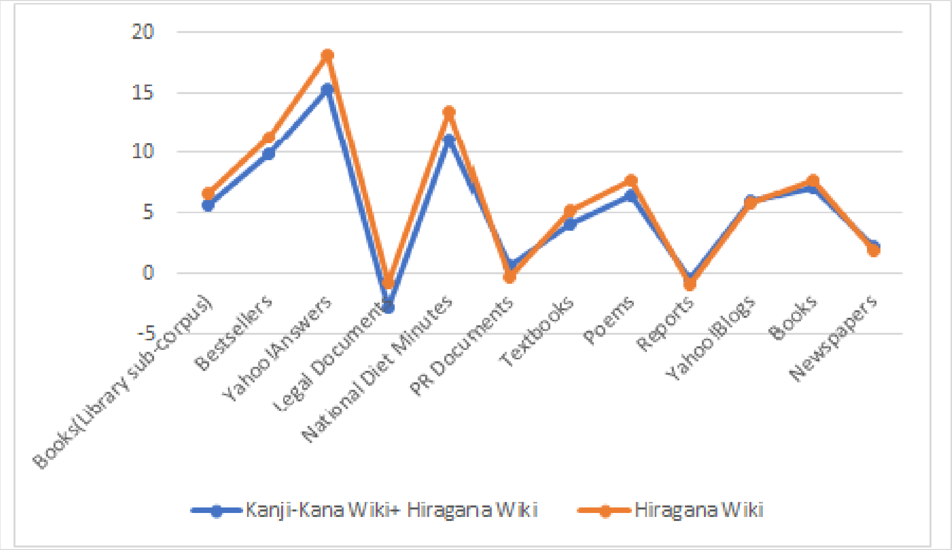


Figure 2. Effect of Hiragana Yahoo! data

Hiragana Yahoo! data was considerably effective for “Yahoo! Answers," “Books," and “National Diet Minutes." Hiragana Yahoo! data is text data of the question answering sites. Therefore, the writing style is rather like spoken language. We believe that it is effective for “Books" and “National Diet Minutes" because the Hiragana Books include dialogues and National Diet Minutes is transcription of the discussion of the Diet. In particular, Wikipedia data rarely include question sentences but question answering sites contain many. Therefore, Hiragana Yahoo! data is effective for “National Diet Minutes" that includes many questions.

Figure 3 shows the effects or the differences of the accuracies of Kanji-Kana Wiki data according to the genres of the texts. The blue line is its effects when the base model is the Hiragana Wiki model, and the orange line is when the base model is the Hiragana Wiki+ Hiragana Yahoo! model. As contrasted to Figure 2, the effects of the data are almost opposite depending on the base model. The blue line is similar to the lines in Figure 2 but the orange line is very different from those. The orange line, the effects of the Hiragana Yahoo! data when it was added to the Hiragana Wiki+ Hiragana Yahoo! model shows that the data is effective for “PR Documents,”“Reports," and“Yahoo! Blogs." The Kanji-Kana Wiki data tends to effective when the fine-tuning of the Hiragana Yahoo! is not effective. These facts indicate that the fine-tuning is effective when the original accuracy was poor, regardless of whether it used Kanji-Kana Wiki text or Hiragana Yahoo! data.

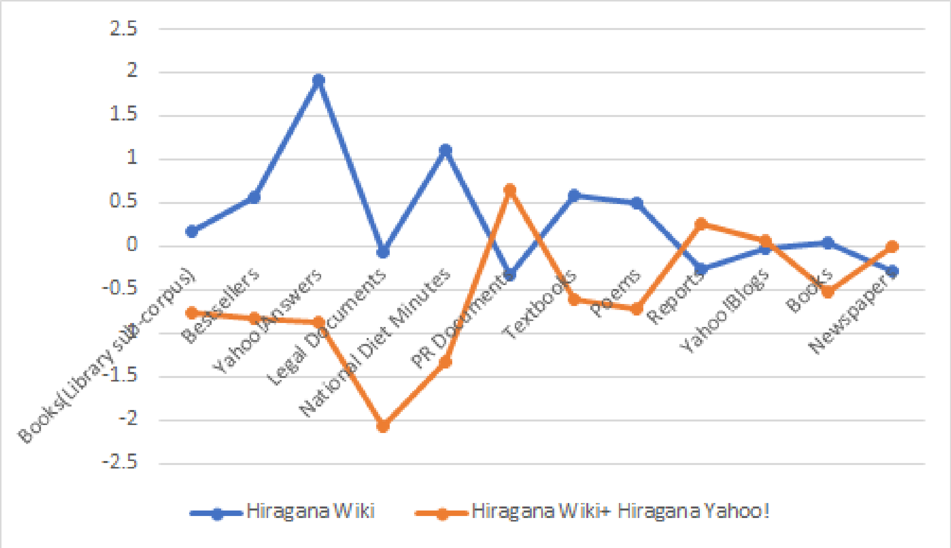


Figure 3. Effect of Kanji-Kana Wiki data

Finally, the accuracy of the Bi-LSTM CRF model still has room for improvement compared to the existing morphological analyzer for Kanji-Kana mixed text. We think that the use of dictionary and much more data should be attempted in the future.

7. CONCLUSIONS

This study developed a morphological analysis model for Japanese Hiragana sentences using the Bi-LSTM CRF model.We showed that the performance of morphological analysis of Hiragana sentences outperformed when we use in-domain data manually annotated for fine-tuning by the experiments using Hiragana data of Wikipedia and Yahoo! Answers data. We also showed that the performance of Hiragana morphological analysis is improved when we fine-tune the model of Hiragana sentences with Kanji-Kana mixed sentences. This fine-tuning is effective when the original accuracy was low. Additionally, we showed that the performance of morphological analysis varied according to the text genre. In the future, we would like to increase the number of training data to improve the accuracy of the model for analyzing Hiragana sentences. Additionally, we would like to make the system capable of outputting word reading, pronunciation, and part-of-speech classification.In this experiment, we used the Yahoo! Answers as a training example for the Kanji-Kana Wiki+ Hiragana Wiki+ Hiragana Yahoo! model and Hiragana Yahoo! models to measure the accuracy, but we would like to train the models using other data to analyze the effect of genre differences.Furthermore, we would like to investigate the effective domain adaptation method when the original accuracy is not bad.

REFERENCES

[1] Kudo, Taku., Yamamoto, K., and Matsumoto, Y. (2004). “Applying conditional random fieldsto Japanese morphological analysis.” *In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pp. 230–237.

[2] Morita, Hajime., Kawahara, D., and Kurohashi, S. (2015). “Morphological analysis for unsegmented languages using recurrent neural network language model.” *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 2292–2297.

[3] Tolmachev, Arseny, Daisuke Kawahara, and Sadao Kurohashi. "Juman++: A morphological analysis toolkit for scriptio continua." *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pages 54-59.

[4] Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso,Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. 2014. "Balanced Corpus of Contemporary Written Japanese".*Language resources and evaluation*, 48(2):345–371.

[5] Jun Izutsu, Riku Akashi, Ryo Kato, Mika Kishino, Taichiro Kobayashi, Yuta Konno, and Kanako Komiya. 2020. "Morphological analysis of hiragana-only sentences using MeCab". Inthe Proceedings of*NLP2020*, pages 65–68 (In Japanese).

[6] Taku Kudo, Hiroshi Ichikawa, David Talbot, and Hideto Kazawa. 2012. "Robust morphological analysis for hiragana sentences on the web". In the Proceedings of NLP 2012, pages 1272–1275 (In Japanese).

[7] Ayaha Osaki, Syouhei Karaguchi, Takuya Ohaku, Syunya Sasaki, Yoshiaki Kitagawa, Yuya Sakaizawa, and Mamoru Komachi. 2016. "Corpus construction for Japanese morphological analysis in twitter". In the *Proceedings of NLP 2016*, pages 16–19 (In Japanese).

[8] Sanae Fujita, Hirotoshi Taira Tessei Kobayashi, and Takaaki Tanaka. 2014. "Morphological analysis of picture book texts".*Journal of Natural Language Processing*, 21(3):515–539 (In Japanese).

[9] Masato Hayashi and Takashi Yamamura. 2017. "Considerations for the addition of hiragana words and the accuracy of morphological analysis". In *Thesis Abstract of School of Information Science and Technology*, Aich Prifectual University, pages 1–1 (In Japanese).

[10] Ji Ma, Kuzman Ganchev, and David Weiss. 2018. "State-of-the-art Chinese word segmentation with Bi-LSTM".In Proceedings of *the 2018 Conference on Empirical Methods in Natural Language Processing*, pages4902–4908. Association for Computational Linguistics.

[11] Suphanut Thattinaphanich and Santitham Prom-on.2019. "Thai named entity recognition usingBi-LSTM-CRF with word and character representation". In *2019 4th International Conference on Information Technology (InCIT)*, pages 149–154.

[12] Tolmachev, Arseny, Daisuke Kawahara, and Sadao Kurohashi. "Shrinking Japanese morphological analyzers with neural networks and semi-supervised learning." *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Volume 1 (Long and Short Papers). 2019, pages 2744–2755

[13] Chen, Xinchi, Xipeng Qiu, Chenxi Zhu, Pengfei Liu, and Xuanjing Huang. "Long short-term memory neural Networks for Chineseword segmentation." *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015, pages 1197-1206

[14] Kikuo Maekawa, Makoto Yamazaki, Takehiko Maruyama, Masaya Yamaguchi, Hideki Ogura, Wakako Kashino, Toshinobu Ogiso, HanaeKoiso, and Yasuharu Den. 2010. "Design, compilation, and preliminary analyses of Balanced Corpus ofContemporary Written Japanese". In Proceedings of *the Seventh International Conference on Language Resources and Evaluation* (LREC 2010), pages1483–1486.

FAKE OR GENUINE? CONTEXTUALISED TEXT REPRESENTATION FOR FAKE REVIEW DETECTION

Rami Mohawesh[1], Shuxiang Xu[1], Matthew Springer[1],
Muna Al-Hawawreh[2] and Sumbal Maqsood[1]

[1]School of Information and Communications Technology, University of Tasmania, Tasmania, Australia
[2]School of Engineering and Information Technology, University of New South Wales, Australian Defence Force Academy (ADFA), Canberra, Australia

ABSTRACT

Online reviews have a significant influence on customers' purchasing decisions for any products or services. However, fake reviews can mislead both consumers and companies. Several models have been developed to detect fake reviews using machine learning approaches. Many of these models have some limitations resulting in low accuracy in distinguishing between fake and genuine reviews. These models focused only on linguistic features to detect fake reviews and failed to capture the semantic meaning of the reviews. To deal with this, this paper proposes a new ensemble model that employs transformer architecture to discover the hidden patterns in a sequence of fake reviews and detect them precisely. The proposed approach combines three transformer models to improve the robustness of fake and genuine behaviour profiling and modelling to detect fake reviews. The experimental results using semi-real benchmark datasets showed the superiority of the proposed model over state-of-the-art models.

KEYWORDS

Fake review, detection, Transformer, Ensemble, Deep learning.

1. INTRODUCTION

The Internet's size and importance has exploded in recent years, and it exerts a significant and growing influence on people's daily lives. Customers usually spend a substantial amount of time online, searching for information on a variety of products, communicating with others, and reading reviews. Additionally, the Internet enables individuals to write reviews on a range of topics based on their expertise and the opinions of others who have seen their work online. These individuals may use their reviews to promote or criticise various products or services [1, 2]. Consumers tend to purchase products that have a high number of good evaluations, which can lead to increased profits for the provider [3]. At the same time, negative reviews can result in financial losses for the companies involved [4]. Since anyone can write reviews without restriction, it is possible to provide undeserved positive or negative feedback with respect to products, services, and enterprises. Hence, it is necessary to verify the truthfulness of opinions and reviews posted online to assist people to avoid being misled by false information.

Ott, et al. [5] stated that human judges have been found to be quite poor at identifying fake reviews, with an accuracy of around 57%. Additionally, it is also difficult to conduct this type of

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 137-148, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112311

identification manually. Detecting fake reviews using automatic detection based on state-of-the-art intelligent technologies has been found to be considerably faster and more accurate than using a human expert.

Many studies have been conducted into developing automated fake review detection models based on classic machine learning [6]. However, many of these models have some limitations resulting in low accuracy in distinguishing between fake and genuine reviews. These models have focused only on linguistic features to detect fake reviews and have failed to capture the semantic meaning of the reviews. Therefore, there is still a need for new models that are able to detect fake reviews efficiently. Recently, transformer or advanced pre-trained architectures have attracted considerable attention in text classification tasks and obtained superior results compared to the previous state of the art methods [7-9]. The effectiveness of these transformer architectures or techniques in constructing deep contextualized embeddings for a variety of texts is the key motivation to use them to develop a new model for detecting fake reviews. However, due the high variance and dynamic change of fake review features, using a single model could not provide a best fit for the entire training data. One transformer may be sensitive to the provided features and biased to specific features, leading to poor performance. Therefore, we present an ensemble approach that combines multiple transformer architectures. The ensemble learning has proven effective and achieved good results in text classification [10, 11]. The study reported in this Chapter was conducted to answer the following question: **Does the ensemble of transformer models perform better than state-of-the-art fake review detection methods?**As a consequence, we have proposed a new ensemble fake review detection model that combines three transformer models, namely, RoBERTa, ALBERT, and XLNet to improve the robustness of fake and genuine review profiling and modelling by handling the dependencies between input and output with attention and recurrence completely.

The main contributions of this paper are as follows:

- We investigate the performance of some transformer models in fake review detection. To the best of our knowledge, no previous study has used XLNet and ALBERT transformer models for fake reviews detection.
- We propose an ensemble model that combines three transformers called, RoBERTa, XLNet, and ALBERT to enhance the accuracy of fake review detection.
- The proposed model is compared with the state-of-the-art models and demonstrate superior performance. It significantly outperforms eight of the most recent fake review detection models.

The paper is organised as follows: Section 2 illustrates the related work done on this research topic. Section 3 provides a Preliminaries of the proposed model. Section 4 describes the proposed methodology in detail. Section 5 provides the experiments setting, the datasets and pre-processing, the evaluation metrics. Section 6 describes the results and discussion. Then, the paper is concluded in Section 7.

2. RELATED WORKS

Naive Bayes (NB) and Support Vector Machine (SVM), are examples of traditional machine learning algorithms that learn discriminant characteristics from reviews and have been used by a number of researchers to detect fake reviews [12],[6],[13]. For example, using the Linguistic Inquiry and Word Count (LIWC) tool [14], Ott, et al. [5] proposed a model to automatically identify fake reviews by combining psychological features from reviews with n-grams, which were then fed to support vector machines. This approach achieved 90% accuracy in fake review categorisation, which is substantially better than human judges were able to achieve. According to the Op-Spam dataset, human judges were only 60% accurate. Feng, et al. [15] used a combination of context-free grammar and Part-of-speech features to detect fake reviews. Their results showed that these combined features could significantly increase fake review detection performance compared to the baseline method. Later, Li, et al. [16] developed a fake review detection model, titled Sparse Additive Generative Model (SAGE) which uses topic modelling [17] with a generalised additive model [18]. The proposed model results on Op-Spam and deception datasets achieved good results with accuracy of 81.82%, 83.10%, respectively.

Cagnina and Rosso [19] combined character n-grams, LIWC, and emotion features for fake reviews detection. Then, these extracted features are fed to an SVM algorithm to classify the reviews. Xu and Zhao [20] developed a model-based deep linguistic feature for fake reviews detection. Then they used an SVM classifier to classify reviews. Fusilier, et al. [21] proposed a model based on character n-grams content features to detect fake reviews. Then NB was used to detect fake reviews. Recently, several deep learning models such as recurrent neural network (RNN), convolutional neural network (CNN), and gated neural network (GRU) have been extensively used and achieved excellent results in the natural language processing field [22] and cybersecurity field [23], [24]. These models deal with dimensional data and extract the semantic presentation. Motivated by this, Ren and Zhang [25] introduced a recurrent convolutional neural network method with an attention mechanism to learn document representation. The proposed model proved its efficiency in detecting fake reviews. Similarly, and by using the context information of the sentences, Ren and Ji [26] also used deep neural networks and proposed a hybrid fake reviews detection model (GRNN–CNN). They combined a gated recurrent neural network (GRU) and a convolutional neural network (CNN). Their proposed model tested on the deception dataset and achieved good results with an accuracy of 83.34%. Later, Zhang, et al. [27] developed a recurrent convolutional deep neural networks model (DRI-RCNN) for fake reviews detection based on word contexts. Their proposed model performed well with 82.9% and 80.8% accuracy on spam and deception datasets, respectively.

More recently, Mohawesh, et al. [6] investigated some promising deep learning and transformers models. According to their experimental results, the RoBERTa transformer exceeded the performance of the state-of-the-art methods with a 91.02% accuracy on the deception dataset. They also found that BERT, DistilBERT, and RoBERTa performed very well with a small dataset. Although various machine learning technologies have been proposed to address fake reviews detection and to aid in distinguishing between fake reviews and genuine ones, it is rarely focused on contextualised text representation models. Thus, this work proposes a new ensemble model which combines three current states of the art deep learning models that can be used with any type of neural classifier and with any type of contextualised text representation and provides a comparative analysis of the performance of several pre-trained models and neural classifiers for fake review detection.

3. PRELIMINARY

Bidirectional Encoder Representations from Transformers (BERT): A BERT-trained model is used to pre-train a deep bidirectional representation of the text that can handle the unlabelled data by focusing on both right and left context in all layers simultaneously[28]. BERT was pre-trained on English Wikipedia text of about 2.5 billion words, consisting of an 800-million-word corpus of books. The BERT model reads a complete sequence of words in parallel, giving the model the ability to understand each word's context as a result of what is located around it. Fig 1 shows the BERT-base-cased model. It consists of 12 layers, 768 hidden layers, 12 heads and 109 million parameters. As shown in Fig 1, input is provided through a CLS token followed by a sequence of words. CLS is a categorisation token in this case. The input is subsequently passed to the layers above. Each encoder applies self-attention, transfers the result through a feed forward network, and then passes the result on to the next encoder in the sequence of layers. We obtained the output from the last transformer block, as shown in Fig 1.

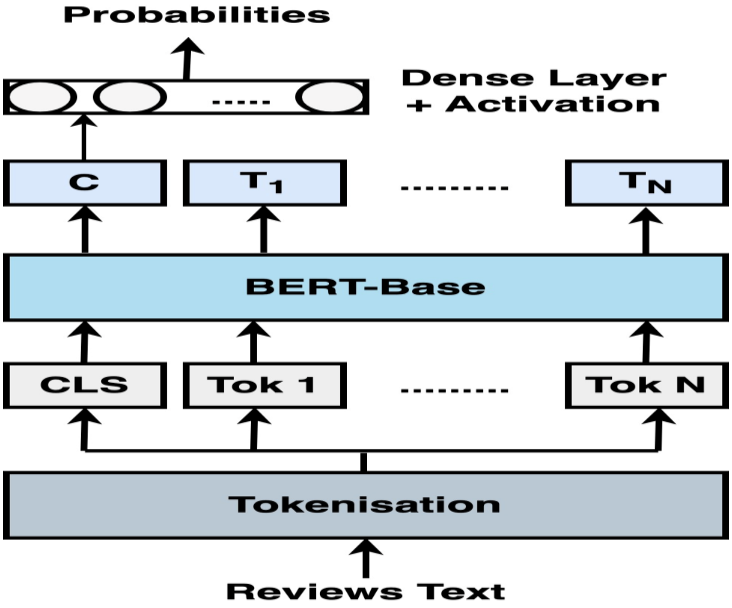


Figure 1. BERT model for text classification.

4. THE PROPOSED MODEL

As shown in Figure 2, the proposed model consists of pre-processing, combines the most recent transformer architectures and ensemble approach. The details of these stages are described as follows:

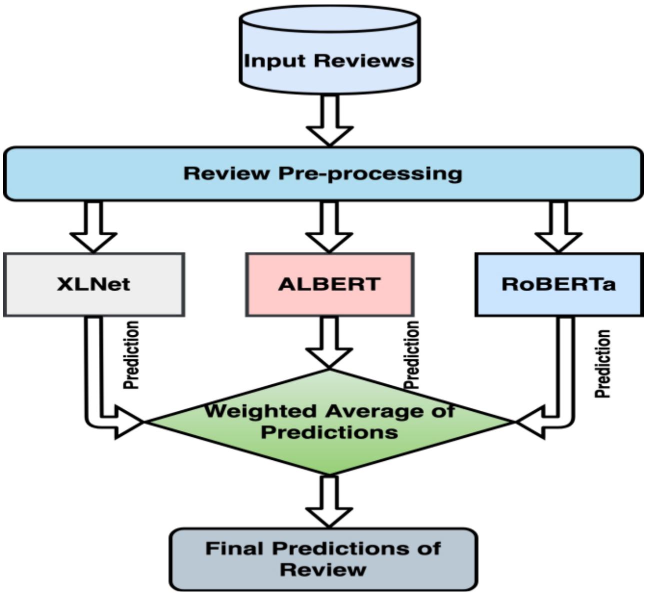


Figure 2. Ensemble of advanced pre-trained models' architecture.

4.1. Pre-Processing stage

To prepare the incoming reviews data as input for transformers, we perform many processes to clean the data and prepare it as input for transformers. We removed noise and irrelevant words, such as URLs and emojis, during this stage. Then, we split and divided each review text into a sequence of words.

4.2. Transformers: The proposed model combines the following three architectures:

Robustly optimised BERT approach (RoBERTa): an updated version of BERT that outperforms the BERT transformer [29] by training the model for a longer time on larger sequences and omitting the subsequent sentence prediction. Along with the English Wikipedia and book corpora, RoBERTa is pre-trained on the Common Crawl News dataset, comprising 63 million English-language news stories. RoBERTa base architecture consists of 768 hidden layers, 12 layers, 125 million parameters, and 12 attention heads. We used BoBERTa due to its high ability in providing dynamic masking patterns for each provided review.

XLNet [8]: It employs Transformer-XL [30] as a feature engineering model to gain a better understanding of the language context, which is an adaptation of the native Transformer. The Transformer XL model incorporates Relative Positional Encoding (RPE) and Recurrence Mechanism components into the Transformer used in BERT to manage long-term dependencies for texts that exceed the maximum permitted input length. An enormous dataset was used to train the XLNet model, which uses permutation language modelling. These permutations are one of the fundamental differences between BERT and XLNet, and they allow for the simultaneous generation of data from both sides. Thus, XLNet is able to learn the best features from the bidirectional review context which represent the fake review efficiently. The XLNet-base architecture consists of 768 hidden layers, 12 layers, 110 million parameters, and 12 attention heads.

ALBERT: is a pre-training natural language representation using modern language models that involves increasing the model size and number of parameters [9]. As a result of memory limitations and lengthier training hours, they can often be challenging to do. ALBERT (A Lite BERT) [9] employs parameter reduction strategies to boost model speed and reduce memory consumption to solve these difficulties. It shares the parameters among layers which provides a high-capacity contextual representation. This way of learning leads to capturing the meaning of words and improving the understanding of the entire text of the review. The A Lite BERT model achieved better results than DL models [31]. In our work, Albert-base-v2 architecture consists of 768 hidden layers, 12 layers, 12 attention heads, 128 embedding, and 11 million parameters.

We added a classification head with a single linear layer to each of the transformers mentioned above to distinguish between legitimate and fake reviews and label them. Then, the full architecture of each transformer and its parameters is fine-tuned to learn the review context.

4.3 Ensemble approach:

To gain the full benefits of transformers and their different perspectives on the learned features, we used the weighted average ensemble approach. In this approach, we extracted the output of the last layer of each transformer, which are logits. Then, we converted logits to probability using the SoftMax function. By using the weighted average of probabilities, we obtained a new probability for each class. The class with maximum probability is the predicted class. This approach gives the better transfer of the higher weight, improving the fake review detection rate and accuracy and preventing the bias of final decision to the less accurate model.

5. EXPERIMENT

This section presents the datasets descriptions, datasets pre-processing, the evaluation metrics and our model results and compares them to state-of-the-art approaches.

5.1. Experimental setup

We used a mini-batch size of 32 to train them on all of the datasets for 10 epochs. Overfitting was avoided by using an early stop [32]. While delta is set to zero, validation loss was used as the metric for early stopping [33]. We used AdamW optimiser [34]. Finally, we computed the loss using binary cross-entropy [35].

To evaluate the performance of the proposed model, we developed a python script and used Google CoLab interface[36]. We also used the transformer library [37], which was created by the *Huggingface* team. We run the experiments multiple times with different parameters. Table 1 presents the best hyperparameters of the three proposed models.

Table 1. The hyperparameter of the proposed models

Models	**Learning Rate**	**Batch Size**	**Type**	**Optimiser**	**Epochs**	**Max Length**
RoBERTa	2e-5	64	roberta-base	AdamW	15	256
ALBERT	2e-5	32	albert-base-v2	AdamW	20	256
XLNet	2e-5	32	xlnet-base-cased	AdamW	20	256

5.2. Dataset Description

In this study, we used two publicly available benchmark datasets. **OpSpam** [5], and **Deception dataset** [16]. **OpSpam** dataset contains 1,600 review texts for twenty hotels in the Chicago area of the United States of America, 800 of which are fake and 800 of which are genuine. A label of '1' indicates fake reviews, whereas a label of '0' indicates legitimate reviews. These reviews came from a variety of sources. The fake reviews were constructed using Amazon Mechanical Turk (AMT), and the remaining were collected from various online review sites such as Yelp, TripAdvisor, and Expedia. **The deception dataset** [16] represents a gold standard dataset containing 3,032 reviews. This dataset contains information about three distinct domains (hotels, doctors, and restaurants). Both datasets have only review text without any metadata information. In our experiments, 80% of the OpSpam and Deception datasets, were used for training, and the remaining 20% of each dataset was used to test the model. Table 2 shows The Statistical information for both datasets.

Table 2. The Statistical information of the OpSpam and Deception datasets.

Datasets	**Domain**	**Reviews Type**	**No. of reviews**	**No. of unique words**	**No. of sentences**
OpSpam[5]	Hotels reviews	Fake	800	14427	7192
		legitimate	800	14812	7963
Deception [16]	Restaurant reviews	Fake	201	5136	1827
		legitimate	201	5126	1892
	Doctor	Fake	356	5128	2369

	reviews	legitimate	200	5098	1151
	Hotel reviews	Fake	1080	16,635	8463
		legitimate	1080	17,328	9258

5.3 Evaluation metrics

In the fake review detection task, recall, precision, and the F-measure have been the most often used metrics. Since the number of reviews of both classes is equal, accuracy is also a popular metric for model evaluation. In order to evaluate the efficiency of the proposed model, we use the following metrics.

- Accuracy: full estimate of correctly classified instances and can be calculated by

$$Accuracy(Acc) = \frac{number\ of correct\ classifications}{total\ numberofdata\ samples} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- Precision: describes the proportion of successfully predicted reviews to the total number of reviews for a given class and can be calculated by

$$Precision(P) = \frac{number\ of correct\ predictions\ of\ each\ class}{total\ number\ of\ predictions\ of\ each\ class} = \frac{TP}{TP + FP} \tag{2}$$

- Recall shows the proportion of relevant reviews achieved from the total number of reviews and is calculated by

$$Recall\ (R) = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions} = \frac{TP}{TP + FN} \tag{3}$$

- F1 score shows the average of precision and recall and is calculated by

$$F - measure = \frac{2 \times recall \times precision}{recall + precision} \tag{4}$$

5. RESULTS AND DISCUSSION

As shown in Table 3t the Acc, P, R, and F1-score for RoBERTa are 94.06%, 89.38%, 98.62%, 93.77%, respectively for Op-Spam dataset, 91.02%, 92.50%, 90.00%, 90.50% respectively for deception dataset. The experiments result of the XLNet model on the OpSpam dataset are as follows: Acc, P, R, and F1-score are 92.50%,86.25%, 98.57%, 92.00%, respectively, where the Acc, P, R, and F1-score on the deception dataset are 88.56%, 77.92%, 93.97%, 85.19%, respectively. On implementing ALBERT model on the OpSpam dataset are as follows: accuracy, precision, recall, and F1-score are 93.34%, 91.25%, 95.42%, 93.29%, respectively, where the Acc, P, R, and F1-score on the deception dataset are 88.03%, 77.08%, 93.43%, 84.87%, respectively. The experiments result of the proposed model on the OpSpam dataset are as follows: accuracy, precision, recall, and F1-score are 94.37%, 89.38%, 99.31%, 94.08%, respectively, where the Acc, P, R, and F1-score on the deception dataset are 92.07%, 84.17%, 96.65%, 89.98%, respectively. It is clear from the experiments that our proposed model achieved the best performance for both OpSpam and Deception datasets due to its efficiency in capturing the robust contextualised representation of each review and the most relevant features using different models and perspectives. Furthermore, our proposed model gains the full benefit of the most accurate transformer using the weighted average in addition to the appropriate support from other transformers.

Table 3. Classification reports for OpSpam and Deception datasets.

	Datasets							
	OpSpam				**Deception**			
Model	**Acc**	**P**	**R**	**F1-score**	**Acc**	**P**	**R**	**F1-score**
RoBERTa	94.06%	89.38%	98.62%	93.77%	91.02%	**92.50%**	90.00%	**90.50%**
XLNet	92.50%	86.25%	98.57%	92.00%	88.56%	77.92%	93.97%	85.19%
ALBERT	93.43%	**91.25%**	95.42%	93.29%	88.03%	77.08%	93.43%	84.47%
Proposed model	**94.37%**	89.38%	**99.31%**	**94.08%**	**92.07%**	84.17%	**96.65%**	89.98%

6.1. Comparison with state-of-the-art models

To evaluate the effectiveness of the proposed model, our model is compared to the state of the art techniques, including SVM [5], SVM [19], SVM [15], SAGE [16], RCNN [38], GRNN–CNN [39], DRI-RCNN [27]. We use accuracy and F1-score according to the existing works [39], [27] as shown in Table 4.

- **SVM** [5]: A model of combining bigram and LIWC features using SVM as a classifier.
- **SVM** [19]: A model of a combination of four grams and LIWC features using SVM as a classifier.
- **SVM** [15]: A model of using unigram features with SVM as a classifier.
- **SAGE** [16]: The Sparse Additive Generative Model (SAGE) is a mix of topic modelling and a generalised additive model.
- **RCNN** [38] is a model of a combination of recurrent neural networks and convolutional neural networks.
- **GRNN–CNN** [39]: it is a hybrid fake reviews detection model. They combined a gated recurrent neural network (GRU) and a convolutional neural network.
- **DRI-RCNN** [27] is a recurrent convolutional deep neural networks model (DRI-RCNN) for detecting fake reviews based on word contexts.
- **BERT-Base Case** [6]: A BERT-trained model is used to pre-train a deep bidirectional representation of the text that is capable of handling unlabelled data by simultaneously focusing on right and left context in all layers.

From Table 4 results, it can be observed that the proposed model outperforms the current state-of-the-art methods on both the OpSpam and deception datasets. We can see that the ensemble of a pre-trained model achieved high accuracy with a small dataset compared to traditional machine learning and deep learning models. For example, our proposed model achieved over 90% accuracy with small training data, while deep learning could not reach 90%. Thus, we can observe that traditional and deep learning models require large datasets for training. However, conducting large datasets is not always possible. Therefore, the transformer model is a pragmatic option in the case of a small dataset.

Table 4. Comparison with the state-of-the-art methods.

Machine Learning Models	**Deception dataset**		**Op-Spam dataset**	
	Average Accuracy	**F1-score**	**Average Accuracy**	**F1-score**
SVM (bigram and LIWC features) [5]	79.33%	82.83%	82.89%	82.09%
SVM (unigram feature) [15]	83.33%	79.53%	86.09%	83.12%
SVM (four grams and LIWC features) [19]	81.67%	81.03%	84.34%	83.21%
SAGA [16]	81.82%	79.38%	83.10%	82.23%
RCNN [38]	82.16%	82.00%	83.21%	81.23%
GRNN-CNN [39]	83.34%	82.86%	84.15%	84.17%
DRI-RCNN [27]	85.24%	83.56%	87.24%	85.36%
BERT Base Case [6]	86.20%	85.50%	90.31%	89.56%
Ensemble (Ours)	**92.07%**	**89.98%**	**94.37%**	**94.08%**

In the real world, detecting fake reviews is a constant challenge, and governments are working to resolve this issue in order to mitigate its negative impacts. Additionally, detecting fake news is a difficult assignment for a machine since it must understand the difference between "legitimate reviews" and "fake reviews." However, we may use a variety of features (e.g., reviews text and emotions) to produce an effective decision about detecting fake reviews. Thus, the more features we incorporate into our models during training, the more effective the model will be at detecting false reviews. In this paper, we employed a technique that is focused on deriving textual features from reviews text dove into the analysis of the differences between fake and real reviews content, indicating that the content of fake and real reviews is significantly different, and the style of fake reviews is more similar to satire than to actual reviews. As a result, contextual features play a critical part in determining the difference between fake and legitimate reviews, as demonstrated in our experiments. Also, according to the findings, the employment of pre-trained models and the ensemble approach significantly improved the results of detecting short fake reviews text.

There are two critical aspects that influence the performance of pre-trained language models: (1) the domain and size of the training datasets and (2) the model's architecture. RoBERTa derives from a diverse variety of data sources; for example, in addition to the standard BookCorpus and Wikipedia datasets, RoBERTa was trained using CC-News [40]. However, in the ensemble approach, the RoBERTa's performance was also improved by combining it with other transformers in the ensemble approach. This combination helps handle the dynamic change of fake review features and the variance in the provided context.

7. CONCLUSION AND FUTURE WORK

In this work, we investigated one significant research question for fake review detection which is "**Does the ensemble of transformer models perform better than state-of-the-art fake review detection methods?"** In order to answer this question, this paper investigated the performance of transformers in detecting fake reviews. We proposed a new model that combines three transformer based models, namely, RoBERTa, XLNet, and ALBERT and uses the weighted average for each classifier to obtain the best result. The proposed model on two semi-real datasets has shown 92.07% and 94.37 accuracies on OpSpam and deception datasets and outperformed the state-of-the-art methods, including traditional and deep learning models. Our proposed model performed significantly better than traditional and other deep learning models using small dataset sizes,demonstrating its efficiency. In future work, we will employ many textual analyses methods in natural language processing, such as named-entity recognition, to extract additional helpful information in addition to textual content embedding to detect fake reviews.

REFERENCES

[1] S. Saumya, J. P. Singh, A. M. Baabdullah, N. P. Rana, and Y. K. Dwivedi, "Ranking online consumer reviews," *Electronic commerce research and applications,* vol. 29, pp. 78-89, 2018.

[2] J. P. Singh, S. Irani, N. P. Rana, Y. K. Dwivedi, S. Saumya, and P. K. Roy, "Predicting the “helpfulness” of online consumer reviews," *Journal of Business Research,* vol. 70, pp. 346-355, 2017.

[3] S. Saini, S. Saumya, and J. P. Singh, "Sequential purchase recommendation system for e-commerce sites," in *IFIP International Conference on Computer Information Systems and Industrial Management*, 2017: Springer, pp. 366-375.

[4] N. N. Ho-Dac, S. J. Carson, and W. L. Moore, "The effects of positive and negative online customer reviews: do brand strength and category maturity matter?," *Journal of Marketing,* vol. 77, no. 6, pp. 37-53, 2013.

[5] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, 2011: Association for Computational Linguistics, pp. 309-319.

[6] R. Mohawesh *et al.*, "Fake Reviews Detection: A Survey," *IEEE Access,* vol. 9, pp. 65771-65802, 2021.

[7] Y. Liu *et al.*, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692,* 2019.

[8] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems,* vol. 32, 2019.

[9] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942,* 2019.

[10] Y.-F. Huang and P.-H. Chen, "Fake news detection using an ensemble learning model based on self-adaptive harmony search algorithms," *Expert Systems with Applications,* vol. 159, p. 113584, 2020.

[11] L. Shi, X. Ma, L. Xi, Q. Duan, and J. Zhao, "Rough set and ensemble learning based semi-supervised algorithm for text classification," *Expert Systems with Applications,* vol. 38, no. 5, pp. 6300-6306, 2011.

[12] E. F. Cardoso, R. M. Silva, and T. A. Almeida, "Towards automatic filtering of fake reviews," *Neurocomputing,* vol. 309, pp. 106-116, 2018.

[13] R. Mohawesh, S. Tran, R. Ollington, and S. Xu, "Analysis of Concept Drift in Fake Reviews Detection," *Expert Systems with Applications,* p. 114318, 2020.

[14] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn, "The development and psychometric properties of LIWC2015," 2015.

[15] S. Feng, R. Banerjee, and Y. Choi, "Syntactic stylometry for deception detection," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2,* 2012: Association for Computational Linguistics, pp. 171-175.

[16] J. Li, M. Ott, C. Cardie, and E. Hovy, "Towards a general rule for identifying deceptive opinion spam," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 2014, pp. 1566-1576.

[17] B. M. DePaulo, J. J. Lindsay, B. E. Malone, L. Muhlenbruck, K. Charlton, and H. Cooper, "Cues to deception," *Psychological bulletin,* vol. 129, no. 1, p. 74, 2003.

[18] J. Hastie Trevor and J. Tibshirani Robert, "Generalized Additive Models. Vol. 43," ed: CRC Press, 1990.

[19] L. Cagnina and P. Rosso, "Classification of deceptive opinions using a low dimensionality representation," in *Proceedings of the 6th workshop on computational approaches to subjectivity, sentiment and social media analysis*, 2015, pp. 58-66.

[20] Q. Xu and H. Zhao, "Using deep linguistic features for finding deceptive opinion spam," in *Proceedings of COLING 2012: Posters*, 2012, pp. 1341-1350.

[21] D. H. Fusilier, M. Montes-y-Gómez, P. Rosso, and R. G. Cabrera, "Detecting positive and negative deceptive opinions using PU-learning," *Information processing & management,* vol. 51, no. 4, pp. 433-443, 2015.

[22] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55-75, 2018.

[23] M. Al-Hawawreh, E. Sitnikova, and F. den Hartog, "An efficient intrusion detection model for edge system in brownfield industrial Internet of Things," in *Proceedings of the 3rd International Conference on Big Data and Internet of Things*, 2019, pp. 83-87.

[24] M. Al-Hawawreh and E. Sitnikova, "Industrial Internet of Things based ransomware detection using stacked variational neural network," in *Proceedings of the 3rd International Conference on Big Data and Internet of Things*, 2019, pp. 126-130.

[25] Y. Ren and Y. Zhang, "Deceptive opinion spam detection using neural network," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 140-150.

[26] Y. Ren and D. Ji, "Neural networks for deceptive opinion spam detection: An empirical study," *Information Sciences*, vol. 385, pp. 213-224, 2017.

[27] W. Zhang, Y. Du, T. Yoshida, and Q. Wang, "DRI-RCNN: An approach to deceptive review identification using recurrent convolutional neural network," *Information Processing & Management,* vol. 54, no. 4, pp. 576-592, 2018.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805,* 2018.

[29] C. Alippi, G. Boracchi, and M. Roveri, "A just-in-time adaptive classification system based on the intersection of confidence intervals rule," *Neural Networks,* vol. 24, no. 8, pp. 791-800, 2011.

[30] Z. Dai *et al.*, "Attentive Language Models Beyond a Fixed-Length Context. arXiv 2019," *arXiv preprint arXiv:1901.02860*.

[31] U. Naseem, I. Razzak, M. Khushi, P. W. Eklund, and J. Kim, "Covidsenti: A large-scale benchmark Twitter data set for COVID-19 sentiment analysis," *IEEE Transactions on Computational Social Systems,* 2021.

[32] L. Prechelt, "Early stopping-but when?," in *Neural Networks: Tricks of the trade*: Springer, 1998, pp. 55-69.

[33] L. Prechelt, "Automatic early stopping using cross validation: quantifying the criteria," *Neural Networks,* vol. 11, no. 4, pp. 761-767, 1998.

[34] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101,* 2017.

[35] L. Rosasco, E. De Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?," *Neural computation,* vol. 16, no. 5, pp. 1063-1076, 2004.

[36] S. V. Halyal, "Running Google Colaboratory as a server–transferring dynamic data in and out of colabs," *International Journal of Education and Management Engineering,* vol. 9, no. 6, p. 35, 2019.

[37] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations,* 2020, pp. 38-45.

[38] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[39] Y. Ren and D. Ji, "Neural networks for deceptive opinion spam detection: an empirical study," *J Inf Sci,* vol. 385, 2017// 2017, doi: 10.1016/j.ins.2017.01.015.

[40] S. Nagel, "Cc-news (2016)," ed.

AUTHORS

RAMI MOHAWESH received the B.E and M.E in computer science and is now working towards a Ph.D degree from the University of Tasmania, Tasmania, Australia. In his PhD research, Rami is the first researcher who investigated the concept drift in fake review detection. He is a reviewer of high impact factor journals such as the Information Processing and Management Journal, Artificial Intelligence Review and Secure Computing. His research interests include Software Engineering, Cloud Computing, Natural Language Processing, Cybersecurity, and machine learning. His current work on Fake Review.

SHUXIANG XU is currently a lecturer and PhD student supervisor within the School of Information and Communication Technology, University of Tasmania, Tasmania, Australia. He has received a PhD in Computing from the University of Western Sydney, Australia, a Master of Applied Mathematics from Sichuan Normal University, China, and a Bachelor of Applied Mathematics from the University of Electronic Science and Technology of China, China. His research interests are Artificial Intelligence, Machine Learning, and Data Mining. Much of his work is focused on developing new Machine Learning algorithms and using them to solve problems in various application fields.

MATTHEW SPRINGER is a lecturer in the School of Technology, Environments and Design at the University of Tasmania. Dr Springer received his Information Systems PhD from the University of Tasmania in 2010. His major focus has been on improving teaching within the Discipline of Information and Communication Technology discipline but is also an active member of the Industry Transformation, and Games and Creative Technologies research groups.

MUNA AL-HAWAWREH received the B.E. and M.E. degrees in computer science from Mutah University, Jordan. She is currently pursuing the Ph.D. degree with the University of New South Wales (UNSW), Canberra, Australia. She works as a Research Assistant at UNSW Canberra Cyber. In her Ph.D. degree, she developed the world's first ransomware framework targeting IIoT edge gateway in the critical infrastructure. Her research interests include cloud computing, industrial control systems, the Internet of Things, cybersecurity, and deep learning. She is a program committee member and a reviewer for several cybersecurity conferences. She was awarded the First Prize for high impact publications in the School of Engineering and Information Technology (SEIT), UNSW, in 2019, and the Dr. K. W. Wang Best Paper Award (2018–2020). She is a Reviewer of high-impact factor journals, such as the IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING

SUMBAL MAQSOOD received the BS. Hons Computer Science from Punjab University College of Information Technology (PUCIT) and MS Computer Science from GC University, Lahore Pakistan. She is currently second year Ph.D student at the University of Tasmania, Tasmania, Australia. She worked as an IT Officer in one of the biggest organisations of Pakistan. Her research interests include machine learning, natural language cybernetics, bio-technologies, data science and software engineering. She is currently working on biosignals analysis using deep learning.

A PERSONALITY PREDICTION METHOD OF WEIBO USERS BASED ON PERSONALITY LEXICON

Yuanyuan Feng and Kejian Liu

Department of Computer and Software Engineering,
Xihua University, Chengdu, China

ABSTRACT

Personality is the dominant factor affecting human behavior. With the rise of social network platforms, increasing attention has been paid to predict personality traits by analyzing users' behavior information, and pay little attention to the text contents, making it insufficient to explain personality from the perspective of texts. Therefore, in this paper, we propose a personality prediction method based on personality lexicon. Firstly, we extract keywords from texts, and use word embedding techniques to construct a Chinese personality lexicon. Based on the lexicon, we analyze the correlation between personality traits and different semantic categories of words, and extract the semantic features of the texts posted by Weibo users to construct personality prediction models using classification algorithm. The final experiments shows that compared with SC-LIWC, the personality lexicon constructed in this paper can achieve a better performance.

KEYWORDS

Personality Lexicon, Machine Learning, Personality Prediction.

1. INTRODUCTION

With the rapid development of the Internet, the number of users using social network platforms is gradually increasing. As one of the largest Chinese social network platform, Weibo has a huge mount of users who will express and share their feelings, expectations and experiences on Weibo. These information not only reflects the status of users, but also may affect the spread of social public opinion.

Psychology study shows that [1] , the way an individual speaks and writes often reflects his or her personality. A person's mindset is based on behaviour, emotion, psychology and motivation, which are collectively called personality and have a great influence on individual behaviour. Weibo contains a large number of texts which can well reflect the psychological activities and personality characteristics of users. For example, people with high extroversion tend to use more words related to positive emotions, while people with high neuroticism tend to use more words with negative emotions [2]. Although there are extensive researches on personality prediction of microblog users at present, most of them are conducted on the basis of users' behaviours. There are not many researches on the construction of special microblog personality lexicon. Therefore, in this paper, the author proposes a personality prediction method based on microblog text. In order to explain the differences of different personalities, this paper adopts the personality analysis method based on lexicon. As a psychological lexicon, Linguistic Inquiry and Word Count (LIWC) [2] is a good choice, but it is not very effective in personality identification. On

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 149-159, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112312

the one hand, the lexicon is more suitable for some formal documents, perhaps because the Weibo version is short and highly colloquial. On the other hand, the lexicon lacks specificity and is not specifically used for personality calculations. Therefore, this paper proposes a method to construct personality lexicon based on “Big Five” model for microblog text. Firstly, word term frequency-inverse document frequency (TF-IDF) is used to extract keywords. Then the clustering algorithm and Word2vec were combined to divide different clusters according to semantics, and then the personality lexicon was formed by word expansion on this basis. Finally, Support Vector Machine (SVM), Random Forest (RF) and Naive Bayes (NB) algorithm are used to achieve personality prediction.

2. RELATED WORK

2.1. Evaluation Methods

There are four main methods to assess personality: questionnaire, dictionary-based and machine learning-based. Questionnaires are the most intuitive method and are designed to ask participants to rate their own behaviour using Likert scales. The most popular questionnaires include the revised NEO-Personality Scale (240 items), NEO Five-factor Scale (60 items) and BFI-44 Big Five Scale (44 items), which are designed for the Big Five personality traits[3]. Although questionnaire is a commonly used method to assess personality traits, participants may be reluctant to fill in the questionnaire or fill it out randomly because some questions involve private information. Therefore, the number of questionnaires collected by this method is usually small and the quality is difficult to guarantee. Therefore, some new methods of predicting human personality are needed. The dictionary-based approach is to detect semantic similarity in the text, that is, semantic similarity equals score. The similarity between different words can be calculated to predict personality. LIWC is a dictionary used to analyse English texts, but Chinese and English have very different grammatical rules. To solve this problem, Liu Qiu et al.[4] determined speech classes and factor structures related to personality traits by analysing the expression modes of Chinese users, and compared with LIWC, they found that in the expression of personality, There are great differences in functional words used in Chinese and English contexts. The method of personality recognition based on machine learning mainly trains the model according to the extracted personality characteristics, so as to achieve personality prediction. As most personality prediction ignores the relationship between different personality traits, individual personality prediction fails to achieve ideal results. In this regard, Gao et al.[5] took different personality traits as a quantitative whole and constructed a multi-objective regression model based on GBDT-Multitarget stacking and BP neural network. The correlation between personality traits was incorporated into the model calculation to predict the entire personality structure of users. Shu Xiaomin et al.[6] predicted Personality based on RAkEL-PA(RAkEL-Personality Analysis), an improved model of RAkEL(Multi-label integration method - random K-label set), and the results showed that there were more people with multiple Personality traits. It suggests that personality traits depend on each other. NavonilMajumder et al.[7] proposed a method of using convolutional neural network (CNN) to extract personality characteristics from articles on stream of consciousness and train different models for different personality traits, so as to achieve the effect of personality recognition.

2.2. Personality Model

Myers Briggs Type Indicator (MBIT) [8] is a popular personality model that has a large audience and is widely used in enterprise human resources. This model describes personality from four dimensions: introversion and extroversion, sense and intuition, logic and emotion, judgment and cognition. According to the scores of different dimensions, MBTI is divided into 16 personality types. However, due to the overly idealistic assumption, MBTI has long been controversial in academic circles, so there are not many in-depth studies on MBTI at present [9] .

The Big Five Model is widely used in predicting personality in the field of psychology and has academic significance [10, 11]. This Model describes personality from Five dimensions: The common cause is openness, conscientiousness, extroversion, agreeableness, neuroticism. Openness describes a person's cognitive style. Conscientiousness reflects the way in which an individual controls, manages, and regulates his or her own behaviour, assessing the organization, persistence, and motivation of an individual in goal-directed behaviour. Extroversion describes the degree of interpersonal interaction and the need for and response to external stimuli. Agreeableness assesses an individual's attitude towards others. Neuroticism reflects the individual's emotional regulation and control. Table 1 shows the performance of each personality traits in general.

Table 1. Personality traits.

Personality Traits	Scores	
	High	Low
Openness	imaginative, aesthetic, creative	pragmatic, compliant, conventional
Extroversion	enthusiastic, lively, good at socializing	implicit, euphemistic, not good at socializing
Agreeableness	trusting, direct, helpful and generous	suspicion, apathy, isolation
Conscientiousness	self-discipline, persistence, achievement, prudence, restraint	laziness, carelessness, weak willpower
Neuroticism	calm, calm, sense of security	vulnerability, depression, insecurity

The Big Five model is the most widely used personality measurement model, no matter in the study of network social behaviour and personality or the study of text and personality. Zhenkun Zhou et al.[12] found that users with lower scores of extraversions tend to post more posts with anger and fear, while users with higher scores tend to post more posts related to sadness. Qiu et al. [13] proved that linguistic features of Twitter can be used to judge agreeableness and neuroticism. Quercia et al. [14] understood the personality characteristics of Twitter users based on the number of followers and followers, and found that popular users and influential users were highly extroverted with relatively stable emotions. In general, the Big Five model is more suitable for academic research, therefore, the author in this paper based on the Big Five personality model to achieve personality prediction.

3. ESTABLISHMENT OF PERSONALITY LEXICON

The overall structure of the personality lexicon construction method for Weibo users in this paper is shown in Figure 1.

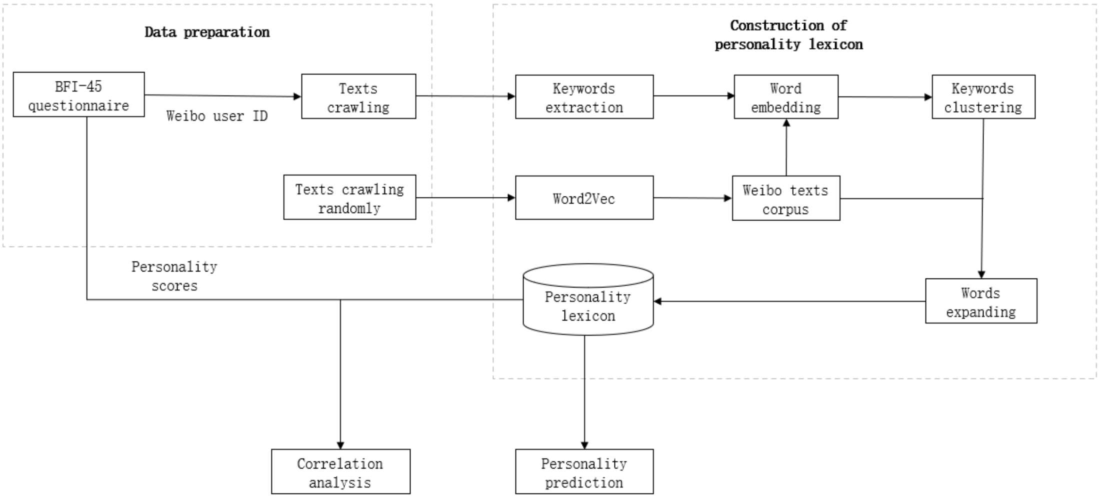


Figure 1. Personality prediction framework

3.1. Data Preparation

3.1.1. Questionnaire Processing

In order to collect users' personality data, the author takes BFI-44 [15] as a questionnaire, which contained a total of 44 questions, including 8 or 9 questions for different personality dimensions. The short questions is convenient for filling in and collecting the questionnaire. In the questionnaire, the user is required to fill in his/her Weibo ID number, so as to facilitate the subsequent retrieval of relevant Weibo text. It took about two months to publish the questionnaire on the Internet. A total of 461 questionnaires were collected, and 379 were valid. Each question was then converted into a score of 1 to 5 on a Likert scale. The personality scoresare shown in Table 2.

Table 2. Personality score statistics.

	Mean	S.D.	L(%)	M(%)	H(%)
A	32.67	4.58	20.69	58.13	21.18
C	27.79	5.08	19.89	57.34	22.77
E	22.85	5.41	22.92	57.19	19.89
N	25.90	4.62	27.83	43.02	29.15
O	30.69	4.90	19.89	57.45	22.66

The personality score obtained from the questionnaire is a continuous variable, which needs to be converted into discrete variables before it can be used in the personality classification model. In this paper, personality scores are converted according to. Table 1 shows the proportions of different personality dimensions.

3.1.2. Data Acquisition and Processing

First of all, crawl the 379 participants' Weibo texts according to their Weibo IDs filled in the questionnaires to form personality dataset. In addition, a Weibo corpus of about 1.3G is constructed by randomly crawling Weibo user texts for subsequent keyword clustering and expansion. Then, the two data sets are cleaned to remove punctuation marks, special symbols and other invalid contents. Finally, use word segmentation tools to cut words.

3.2. Construction of Personality Lexicon

In this paper, the construction of personality lexicon is mainly divided into two steps: extraction of personality keywords and construction of personality lexicon. The first step is to extract keywords related to personality traits from users' texts. The second step is to divide words into multiple clusters using k-means algorithm and Word2vec. By analysing each cluster, the semantic name is given to it. Then, the keywords extracted from each cluster are extended to obtain a personality lexicon containing words of different semantic categories.

3.2.1. Keywords Extraction

In information retrieval, TF-IDF, as the most widely used method, reflects the importance of a word to documents in a textual corpus. It assigns weights to each term in the document based on its term frequency and inverse document frequency. Items with higher weight scores were considered more important. Therefore, the author uses TD-IDF method to calculate the weight of each word in the user's texts, and extracts a certain number of high-weight words. Although TF-IDF algorithm can extract keywords, not all keywords are related to personality. Therefore, this paper uses Chi-square to select the first N keywords related to personality based on users' personality scores.

3.2.2. Construction of Personality Lexicon

In order to construct a personality lexicon, k-means clustering algorithm and Word2vec are used to put keywords with similar semantics into the same cluster. Before clustering, the appropriate number of clustering K should be determined first. For this reason, the author did a series of experiments with K values between 5 and 30, and finally selected K=18, which performed better, as shown in Figure 2. And we list some keywords of categories in appendix at the end of this paper. These keywords are words close to the cluster centre, which can well describe the overall characteristics of each category.

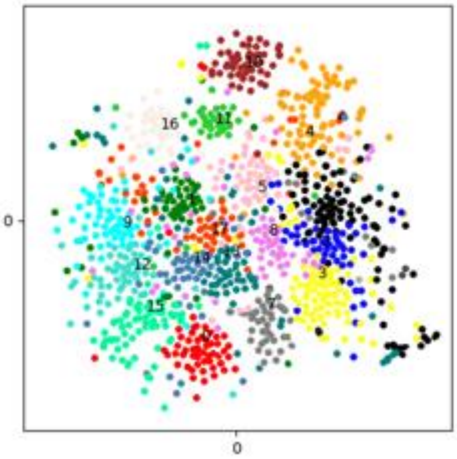


Figure 2. 18 word categories

In appendix, Categories 0 and 11 are related to evaluation, expressing attitudes towards people and things, including positive and negative evaluation; Categories 1 is related to time; Categories

2 is words related to daily life. Categories 3 is related to relationship, including family, friends and strangers; Category 4 is about to places or attractions; Categories 5 is the description related to cognitive process; Categories 6 is blessing words; Categories 7 is related to platform activities, such as Weibo lucky draw, Weibo red packets and activities sharing on other platforms; Categories 8 and 15 describe people's emotional states, including positive and negative emotions. Category 9 relates to physical health, mainly describing body parts and health conditions. Category 10 is related to social events; Category 12 is job-related; Category 13 is related to values; Category 14 relates to school life; Category 16 has to do with sports activities; Category 17 relates to food. The personality lexicon constructed in this paper not only has targeted classification, but also incorporates colloquial words and buzzwords on the Internet, which is of great help to predict the personality of Weibo users.

4. Correlation Analysis

In order to understand the correlation between text features and users' personality scores, the author extracts keywords to represents the microblogs with semantic categories or topics. With help of personality lexicon, the author calculates the number of each semantic category of keywords from users' microblogs and take them as the personality lexicon features of the user. Then Pearson correlation is performed based on the personality scores and the personality lexicon features. According to the analysis results, personality traits can be explained from the perspective of text. As can be seen from Table 3, for example, agreeableness is negatively correlated with work, and users with high agreeableness express more blessings to others; Extroversion is positively correlated with relationship, indicating that users with higher extroversion pay more attention to communication with others. Neuroticism is positively correlated with both negative and positive emotions, indicating that users with high neuroticism are emotionally unstable. Openness are positively correlated with locations, cognition and values, indicating the creativity, imagination and exploration ability of them.

Table 3. Correlation coefficient between personalities and categories

Label	Categories	A	C	E	N	O
0	Positive evaluation	0.017*	-0.019	-0.02*	0.014**	0.012
1	Time	0.02*	0.049**	0.053	-0.07	-0.041
2	Daily life	0.081	0.04	0.031	0.049*	0.096
3	relationship	-0.055	0.039	0.053**	0.012	-0.108
4	Locations	-0.07	0.021	-0.073	0.016	0.153***
5	Cognition	-0.085	0.029	-0.001	0.004*	0.052**
6	Blessing	0.081*	-0.048	-0.08	-0.078	-0.013
7	Platform activities	0.015	-0.101	0.012**	-0.012	0.084
8	Positive emotion	0.064	-0.027*	0.065	0.013*	-0.039
9	Health/Body	0.057*	0.072	-0.06*	0.06*	0.101**
10	Social event	-0.105	0.031	-0.024**	0.041*	0.078
11	Negative evaluation	-0.054**	-0.05*	-0.069	0.055**	-0.025
12	Job	-0.026***	0.077**	-0.014	-0.005	-0.08
13	Value	-0.015**	0.021	-0.031*	-0.027	0.044**
14	School life	0.035*	0.021*	0.057**	0.009	0.043
15	Negative emotion	-0.089*	0.032	-0.044	0.018***	-0.073
16	Sports activities	0.129*	-0.11	0.035	0.015	0.057
17	Food	0.041	0.055	0.034	0.029	0.069***

* Denote significance at 10% level; ** Denote significance at 5% level; *** Denote significance at 1% level.

5. Personality Prediction Model

From personality questionnaires, we get Big-five personality scores that are represented as high, medium and low grade, and make use of machine learning algorithm for training the five classifiers. Due to deep learning is based on the neural network algorithm, and require a lot of data for training so as to have a better performance. However, through the questionnaire only 379 effective questionnaires were collected, which is not enough for deep leaning training, so the traditional machine learning algorithm is a good choice. Therefore, SVM, RF and NB models are used as classifiers in this paper to make predictions based on the constructed personality lexicon and SC-LIWC [16] , simplified Chinese version of LIWC. The experimental results are shown in Table 4.

Table 4. Comparison of model accuracy.

Lexicon	Model	A	C	E	N	O	Average
Personality Lexicon	RF	**0.736**	**0.710**	**0.551**	**0.600**	**0.603**	**0.641**
	SVM	0.476	0.579	0.457	0.516	0.550	0.516
	NB	0.428	0.474	0.451	0.357	0.517	0.445
SC-LIWC	RF	0.523	0.518	0.491	0.493	0.479	0.501
	SVM	0.453	0.485	0.405	0.398	0.409	0.430
	NB	0.421	0.433	0.394	0.399	0.376	0.405

As can be seen from Table 4, the highest average accuracy of SC-LIWC is 0.501 of RF, and the lowest average accuracy of NB. The average accuracy of the personality lexicon constructed in this paper is 0.641 and 0.445, indicating that the personality lexicon is more effective for personality prediction. Compared with the different models of the two lexicons, the accuracy of RF is higher than the other two models, indicating that RF combined with the lexicons can improve the accuracy of personality prediction. In terms of personality dimension, RF model based on personality lexicon has the best performance, with relatively high accuracy of agreeableness and conscientiousness, reaching 0.736 and 0.710 respectively, which maybe because users with these two personality traits published a large number of microblog texts and learned more text features during training.

All in all, the personality lexicon proposed in this paper has a good applicability to Weibo. Although SC-LIWC is the simplified Chinese version of the lexicon, it still retains the grammatical and semantic features of English, leading to not good results in the analysis of Chinese microblog. In addition, the Weibo as Chinese social networking platform, users are usually published some texts of colloquial and network new words emerge in endlessly, SC-LIWC hasn’t these special words. And the lexicon in this paper is build based on Weibo texts, more specific and detail, achieving a better performance when predicting personality.

6. Conclusions

Based on the "Big Five" personality theory, this paper first constructed a personality lexicon for Weibo texts to explain different personality characteristics from the perspective of text. Then, machine learning algorithms RF, SVM and NB are combined to achieve personality prediction of microblog users. The final experimental results show that the combination of RF model and personality lexicon proposed in this paper can achieve good results.

On social platforms, emojis in the text posted by users will also reflect a person's personality characteristics[17] . But due to the small amount of text containing emojis, we didn't considered emojis in this paper. In the future, we will collect more texts with emojis, which can be added to establish a personality expression lexicon to improvethe accuracy of personality prediction.

APPENDIX

Table 5. Some keywords in categories.

Label	Name	Keywords
0	Positive evaluation	优秀/excellent，节奏/rhythm，性格/character，不错/good，外向/outgoing，完美/perfect，印象/impression，养成/form，表现/performance，理智/rational，不愧/worthy，成熟/mature，忍耐/patient，用功/hardworking，冷静/calm，良好/not bad，不含糊/unambiguous，大开眼界/eye-opening，肯吃苦/willing to bear hardships
1	Time	周一/monday，周末/weekend，早上/morning，第二天/next day，晚上/night，半夜/midnight，明年/next year，前些天/the day before，假期/holiday，以前/before，三天/three days，以后/after，目前/now
2	Daily life	叙旧/talk about the old days，聊天/talk，碰巧/happen to，路过/pass by，偷跑/sneak away，沏茶/make tea，过寿/celebrate the birthday，握手/shake hands，逛商场/go to the mall，购物/shopping，打折/discounts，酒友/drinking buddies，吸烟/smoking，天气/weather
3	Relation	爸爸/dad，爸比/daddy，姐妹/sisters，爸妈/dad and mom，朋友/friend，伙伴/partner，同事/colleague，对方/the opposite side，好朋友/good friend,，妈咪/mommy，姐姐/sister，爷爷奶奶/grandparents，邻居/neighbor，宝贝/baby，老公/husband，哥哥/brother
4	Locations	上海/Shanghai，南京/Nanjing，呼和浩特/Hohhot，济南/Jinan，泰国/Thailand，烟台/Yantai，列车/Train，武汉/Wuhan，城市/City，九华/Jiuhua，内蒙古/Inner Mongolia，镇江/Zhenjiang，中心/Center，成都/Chengdu，杭州/Hangzhou，北京/Beijing，美术馆/Art Museum
5	Cognitive	理解/comprehend，选择/choose，质疑/question，不解/wonder，迷惑/puzzle，搞清楚/make clear，明白/figure out，意识到/realize，渐渐/gradually，懂得/understand，熟悉/familiar，知道/know，英雄所见略同/great minds think alike
6	Blessing	祝愿/wish，生日快乐/happy birthday，迎接/to meet，鸿运/good luck，愿望/wishes，顺利/go smoothly，花好月圆/blooming flowers and full moon，家庭幸福/family happiness，牛年大吉/Happy Chinese New Year，红火，欢庆/celebrating，祝福/blessing，美好/beautiful
7	Platform activities	助力/help，投票/vote，超级/super，爱豆/idol，人气/popularity，投出，大赏/big reward，战队/team，直播/Live，盛典/grand ceremony，第一波/the first wave，喜获/congratulations to obtain，首档节目/the first TV show，红包/red packet，现金/cash，收到/received，抽奖/draw，关注/focus on，大奖/reward，惊喜/surprise，机会/opportunity，福袋/bless

		ing bag，集齐/collect all，门票/tickets，礼包/gift bag，会员/membership，免费/free，等级/level
8	Positive emotion	加油/fighting，人生/life，努力/effort，未来/future，生命/life，幸福/happiness，夏天/summer，感谢/thanks，力量/power，美好/beauty，能量/energy，青春/youth，拥有/own，感受/feelings，再见/goodbye，心中/In the heart，路上/on the road，长大/grow up，少年/teenager，感动/moved，值得/worth，期待/look forward to
9	Health/Body	闹肚子/stomach trouble，保健/health care，牙龈/gums，桑拿/sauna，按摩/massage，维生素/vitamins，布洛芬/ibuprofen，买药/buy medicine，止疼药/painkillers，皮肤科/dermatology，头孢/cephalosporin，甲沟炎/parchitis，疼死/really painful，腰疼/backache，酸痛/ache，头晕眼花/dizziness，手指/fingers，小腿/leg，腹肌/abs，脑子/brain
10	Social events	劫持人质/hostage-taking，罹难/death，染病/illness，避让/avoidance，跳窗/jumping out of a window，重创/trauma，重判/severe sentence，识破/see through，下药/drugging，恶性事件/vicious incident，急救车/ambulance，受害者/victim，性骚扰/sexual harassment，侮辱性/abuse，保姆/babysitter，罚跪/be punished to kneel，打骂/beat and scold，离婚冷静期/cooling-off period before divorce，捐献/donation，家暴/domestic violence
11	Negativeevaluation	讨厌/dislike，阴阳怪气/speak in a voice dripping with sarcasm，傻逼/sucker，恶心/disgusting，生气/angry，可恶/damn，玩笑/joke，不配/don’t deserve，卸载/uninstall，得寸进尺/The more one gets, the more one wants，自导自演/self-speech，无礼/rude，缺心眼儿/stingy
12	Job	会议/meetings，交接/handover，人力资源/human resources，小组/groups，迟到/tardiness，上班/to work，工资/salary，五险一金/five social insurance and one fund，退休/retirement，加班/work overtime，老板/boss，助手/assistant，业绩/KPI
13	Value	诚信/integrity，社会/society，为荣/honor，为耻/shame，国家/country，精神/spirit，文化/culture，权利/rights，清朗/Qinglang，文化观/cultural outlook，集体主义/collectivism，理想信念/ideals and beliefs，道德水准/moral standards，引导/guidance，纪律/discipline，守法/law-abiding
14	School life	大学生/college student，老师/teacher，学校/school，学习/study，作业/homework，论文/paper，同学/classmate，考研/prepare for postgraduate exams，开学/，上课/go to a class，毕业/graduation，考试/exam，宿舍/dormitory，公共课/optional course，考题/examination questions，没考上/failed in the exam，六级/College English Test-6
15	Negative emotion	事情/things，情绪/emotion，难过/sad，经历/experience，越来越/more and more，心情/mood，害怕/fear，想到/think of，失望/disappointment，身边/side，内心/inner，痛苦/pain，

		也许/maybe，放弃/give up，焦虑/anxious，悲伤/sorrow
16	Sports activities	国足/national football team，女篮/women's basketball team，乒乓球/table tennis，奥运会/The Olympic Games，锦标赛/championship，打球/play a ball game，跳水/diving，跑步/running，裁判/referee，场地/site，夺冠/take the crown，冠军/champion，金牌/gold medal，银牌/silver medal，口哨/whistle
17	Food	青稞/highland barley，牛奶.milk，味道/taste，鲈鱼/perch，火锅/hotpot，米饭/rice，橘子/orange，汉堡/hamburger，鸡蛋/egg，真香/really fragrant，蛋糕/cake，咖啡/coffee，好吃/delicious，海鲜/seafood，奶茶/milk tea

REFERENCES

[1] G. Stemmler and J. Wacker, "Personality, emotion, and individual differences in physiological responses," Biological Psychology, vol. 84, no. 3, pp. 541-551, 2010.

[2] J. W. Pennebaker and M. E. Francis, "Linguistic inquiry and word count: liwc {software program for text analysis}, 1999.

[3] P. John, L. P. Naumann, and C. J. Soto, "Paradigm shift to the integrative big five trait taxonomy: History, measurement, and conceptual issues", Handbook of personality: Theory and research (3rd edition), 2008.

[4] Q. Lin, L. Han, J. Ramsay, and Y. J. Fang, "You are what you tweet: Personality expression and perception on Twitter," vol. 46, no. 6, pp. 710-718, 2012.

[5] G. J. B, "Users' Personality Prediction Model Based on Multi-Target Regression %J International Journal of Computational and Engineering," vol. 4, no. 4, pp. 25-28, 2019.

[6] S.Xiaoming and M.Xiaoning, " Research on user personality Analysis Model based on microblog text " , Software Guide, vol. 19, no. 11, pp. 25-28, 2020.

[7] N. Majumder, S. Poria, A. Gelbukh, and E. Cambria, "Deep Learning-Based Document Modeling for Personality Detection from Text." IEEE Intelligent Systems, vol. 32, no. 2, pp.74-79, 2017.

[8] Myers and M. Mccaulley, Mbti Manual: A Guide to the Development and Use of the Myers - Briggs Type Indicator. Consulting Psychologists Press, 1985.

[9] D. J. Pittenger, "Cautionary comments regarding the Myers-Briggs Type Indicator," Consulting Psychology Journal, vol. 57, no. 3, pp. 210-221, 2005.

[10] L. Goldberg and R. Lewis, "The development of markers for the Big-Five factor structure," Psychological Assessment, vol. 4, no. 1, pp. 26-42, 1992.

[11] S. D. Gosling, P. J. Rentfrow, and W. B. Swann, "A very brief measure of the Big-Five personality domains," Journal of Research in Personality, vol. 37, no. 6, pp. 504-528, 2003.

[12] Z. Zhou, K. Xu, and J. Zhao, "Extroverts Tweet Differently from Introverts in Weibo," EPJ Data Science, vol. 7, no.1, pp.18, 2018.

[13] L. Qiu, J. Lu, J. Ramsay, S. Yang, W. Qu, and T. Zhu, "Personality expression in Chinese language use," Journal of Research in Personality, vol. 52, no. 6, pp. 463-472, 2017.

[14] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft, "Our Twitter Profiles, Our Selves: Predicting Personality with Twitter," in IEEE Third International Conference on Privacy, 2012.

[15] O.P. John, S. Strivastava," The Big-Five trait taxonomy: History, measurement, and theoretical perspective," in L. A. Pervin and O. P. John(Eds.), Handbook of persinality: Theory and research, vol.2, pp. 102-138, 1999.

[16] C. L. Huang, C. K. Chung, N. Hui, Y. C. Lin, and J. W. Pennebaker, "Development of the Chinese linguistic inquiry and word count dictionary," Chinese Journal of Psychology, vol.54, no. 2, pp. 185-201, 2012.

[17] D. Marengo, F. Giannotta, M. J. P. Settanni, and I. Differences, "Assessing personality using emoji: An exploratory study," Personality & Individual Differences, vol. 112, no.c, pp. 74-78, 2017.

AUTHORS

Yuanyuan Feng is in department of Computer and Software Engineering, Xihua University, Chengdu, China. She is currently working on natural language processing, particularly ontexts classification.

Kejian Liu is a professor in department of Computer and Software Engineering at Xihua University. He is working on Computer network and wireless network technology, intelligent information processing technology, high-performance computing technology.

ANEC: ARTIFICIAL NAMED ENTITY CLASSIFIER BASED ON BI-LSTM FOR AN AI-BASED BUSINESS ANALYST

Taaniya Arora, Neha Prabhugaonkar,
Ganesh Subramanian and Kathy Leake

Crux Intelligence, New York City, New York, USA

ABSTRACT

Business users across enterprises today rely on reports and dashboards created by IT organizations to understand the dynamics of their business better and get insights into the data. In many cases, these users are underserved and do not possess the technical skillset to query the data source to get the information they need. There is a need for users to access information in the most natural way possible. AI-based Business Analysts are going to change the future of business analytics and business intelligence by providing a natural language interface between the user and data. This natural language interface can understand ambiguous questions from users, the intent and convert the same into a database query. One of the important elements of an AI-based business analyst is to interpret a natural language question. It also requires identification of key business entities within the question and relationship between them to generate insights. The Artificial Named Entity Classifier (ANEC) helps us take a huge step forward in that direction by not only identifying but also classifying entities with the help of the sequence recognising prowess of BiLSTMs.

KEYWORDS

Named Entity Recognition System, Natural Language Processing , Business Analytics, Question Answering Systems, Bi-directional LSTMs

1. INTRODUCTION

At Crux Intelligence, we envisage a break-through in the analytics industry by building an AI based business analyst (ABBA) [1] that performs the functions of a business analyst. The main aim of ABBA is to create a Natural Language interface between the user and the data. This not only helps simplify data access, but also brings the user closer to the data.

Analyza [2], discusses some of the challenges faced while developing such systems. The paper also highlights why Structured Query Language despite being a widely accepted database access tool, is not user friendly and requires far too much knowledge of the physical layout of the database. Thereby it substantiates the use of Natural Language interfaces for such applications.

The most crucial role in such interfaces is played by Question Interpreter which performs the job of understanding user questions and tries to extract structured data from it.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 161-175, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112313

1.1. AI-based Business Analyst

An ABBA supports business leader(s) to make effective decisions. They know the discipline of analytics, understand the data, know how to access and absorb the data and help in decision making. A human business analyst also helps leaders take the right decisions by understanding the business problem, running relevant analysis and producing reports which are easy to consume for the user.

At Crux Intelligence, we are building an ABBA which will help in enhancing the capabilities of a human business analyst and help in making better decisions. Its key component is a question answering system which understands business queries of users and analyses enterprise data to generate appropriate answers. The input to the system is a question entered by a user in natural language. The question is analysed and processed, and the output is an answer, or a list of answers in the form of trends, bar graphs, tables, and numbers. The ABBA is capable of answering the following range of questions:

- **Data retrieval questions:** Direct questions related to entities and metrics. For example, ‘*What is the sales in New York?’*.
- **Comparative questions:** Questions involving more than two entities, time periods, etc. For example, ‘*Shipment in Jan vs Feb*’, ‘*East vs West*’.
- **Conditional questions:** Questions having conditions on entities, for example, ‘*Cities having sales > 3M and < 8M*’.
- **Questions with filters:** Question with filters like Top/Bottom, for example, *‘Top 5 stores in Texas’*.
- **Incomplete and non-elucidated questions:** For example, ‘*Sales*’, ‘*Last Month*’.
- **Questions with complex periods:** For example, ‘*MTD sales for last 3 years for East*’, ‘*Daily sales from Jan to March 15, 2021*’.

The main task of ABBA is to automatically find the right answer by identifying the entities and intents from the question. Classification of entities is a non-trivial task due to ambiguities present in a user question which may result in classifying an entity into multiple entity categories and hence may lead to different interpretations within the Question Interpreter. We describe this in detail in subsequent sections.

1.2. Named Entity Recognition

Named Entity Recognition (NER) task aims to identify entities in text and classify them into entity categories. It plays a key role in many Natural Language Processing Tasks including Question Answering. Typical examples of entities and entity categories are listed in Table 1.

Table 1. Entity Categories.

Entity Category	Entities
Location	New York, Chicago, Hong Kong
Date	Wednesday, January
Person	Albert Einstein, Mahatma Gandhi
Organization	Google, Tesla, UNESCO

Cuddle.ai [1] describes the role of Question Interpreter in the system and challenges faced during interpretation. One of the tasks within the interpreter is entity extraction, where, for a question, "*How many cars were sold in New York?*", '*cars*' is an entity of category '*Product*' and '*New York*' is an entity of category '*Region*'. It also encounters ambiguities in user questions due to closed domain terminologies where an entity can be classified into multiple entity categories in different contexts. The ability to find correct and relevant answers relies heavily on the Named Entity Recognition task performed on the users' questions.

Several high quality NERs such as those by Stanford, [3] and Spacy, [4] are available. Since these models are targeted at an open domain, they could not be used to meet the special needs of a closed domain system. A major limitation of using such NERs is that they typically classify proper nouns and sometimes numbers or alphanumeric entities like dates as entities. Business entities like '*sales*' or '*number of cars sold*' will remain unrecognized while using such NERs. One of the major reasons of this limitation is that entities in closed domain are not always proper nouns. They can be verbs or even adjectives.

Another approach for identifying entities is by using part of speech tags (POS). In such a scenario, the accuracy of the question interpreter is largely dictated by the accuracy of the POS-Tagger, which is sensitive to case types and the domain on which it was trained. For example, for question, "*What is the sales of Greater Cincy East?*", where '*Greater Cincy East*' is a location. POS Taggers would easily identify '*Greater*', '*Cincy*' and '*East*' as three proper nouns. However, for question, "*What is the sales of greater cincy east?*" we would find that the token '*east*' has been marked as an adjective. Such cases are important to handle in closed domain system where the question is either typed or converted from speech.

Therefore, it is important to have an intelligent domain agnostic model which can also support the specific scenarios discussed earlier. We used BiLSTM architecture to identify entities and classify them into entity categories. The detailed architecture is described in the subsequent sections.

Entity disambiguation becomes even more challenging when the user questions are shorter in length and when an entity gets mapped to multiple categories due to insufficient context in the question. A system to use external knowledge was proposed by Feng [5], where they used a knowledge enhanced Named Entity Disambiguation model which involved using a factual and a conceptual knowledge graph to improve named entity disambiguation for short and noisy texts. We have also used knowledge to further improve our disambiguation performance in the form of custom knowledge dictionary with a different approach which we will describe in section 3.

1.3. LSTMs for Named Entity Recognition

LSTMs [6] are exceptionally capable of learning sequences. Their sequence learning capability find extensive use in NLP. A bidirectional LSTM [7] is even more potent as it makes two passes of the same sequence. Therefore, while tagging an element in a sequence, a BiLSTM not only keeps in mind the past elements but also the elements ahead of it. More advanced neural models have been created for open domain systems using a combination of BiLSTMs with CNNs [8] and CNN along with CRF [9].We chose to use just the BiLSTM model for our named entity classification task as our system deals with a closed domain. The actual meaning of the token has a lower importance in our system in comparison to the sequence it is a part of.

The rest of the paper is organized as follows. We describe various entity categories in section 2, data preparation steps in section 3 and describe the system architecture in section 4. The evaluation procedure is described in Section 5 and the results and error analysis is detailed in Section 6 followed by conclusion.

2. ATTRIBUTES

The process of extraction of structured data from a user question requires us to have certain structured headers under which we categorize the data. We use the term Attribute to refer to an entity identified in a question and attribute class to refer to its category. We will also use these terms in subsequent references. The term Entity signifies a different meaning in ANEC which will be discussed in this section.

The five important attributes considered for named entity classification task are as follows:

- **Entity:** Examples include IDs of *Regions*, *Stores*, *Brands* and actual names like *New York*, *Delhi*, *Texas*.
- **Entity Type:** This refers to the type of entities. For example, *New York* has entity type *City* as well as *State*, *Delhi* has an entity type of *Region*. *Coca Cola* is a brand whereas *Diet Coke* is of type sub-brand.
- **Metric:** A metric is a countable concept as captured in the enterprise database. The derived word '*sold*' corresponds to *Sales* metric.
- **Temporals:** Temporals refer to time and period values. For example, *this week*, *July*, *from Jan 31 2020 to Dec 31 2021* and *January 2018*. It also includes business specific temporals and its abbreviations such as *YTD* (Year to Date), *Q1* (Quarter1), *JFM* (JanFeb-Mar) and *MTD* (Month to Date).
- **Conditions and Filters:** Conditions and filters include words like *highest*, *top-K* and any other conditions that the user wants to apply on the attributes of the question.

3. DATA PREPARATION

The input to the Question Interpreter (QI) is a user question. Two modules within the QI, namely Period Identifier and Condition and Filter identifier, identify Temporals and Condition and Filter attributes from the question respectively. These attributes along with the question are sent as input to ANEC which further identifies and classifies other attributes in the question using knowledge dictionary.

3.1. Knowledge Dictionary Creation

We created three knowledge dictionaries, one each for Entity, Entity Type and Metric attributes. Each dictionary contains words corresponding to its attribute type. Another separate dictionary contains words that occurred across multiple dictionaries. For example, '*Customer Segment Sales*' where '*Customer Segment*' was an Entity Type, '*Customer 01*' an Entity and '*Sales*' as a Metric.

3.2. Data Augmentation

Historical dump of the question database was taken and a total of 1,442 questions were retrieved. The dump consisted of questions, Entity, Entity Type and Metric tokens present in each question. Templating was performed to generate more questions. Each question in the question dump was taken and its attributes were replaced with their respective placeholders. A few complex templates were also generated synthetically. The method used for templating is illustrated in Figure 1.

A few examples of questions generated via templating are illustrated in Table 2. While using the actual IDs in the placeholder for the Entity, the Entity Type was mentioned along with it.

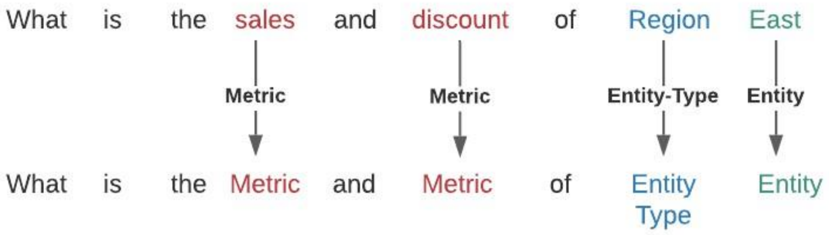

What is the sales and discount of Region East
Metric
Metric
Entity-Type
Entity
What is the Metric and Metric of Entity Type Entity

Figure 1. Templating of questions

Table 2. Replacing placeholders with corresponding attributes

Template	What is the METRIC and METRIC of Entity Type - Entity
Metric 1	Sales
Metric 2	Target
Entity	90
Entity Type	Store
New Question	What is the sales and Target of Store 90

Template	What is the METRIC and METRIC of Entity Type - Entity
Metric 1	Sales
Metric 2	Discount
Entity	East
Entity Type	Region
New Question	What is the Sales and Discount of East

Template	METRIC of (Entity Type – Entity) vs (Entity Type - Entity)
Entity 1	Region West
Entity 2	Region
Entity	9
Entity Type	Region
Metric	Sales
New Question	Sales of Region 9 vs Region West

However, in case of the actual names of entities, the Entity Type placeholder was dropped. The Entity and Metric dictionaries were iterated over, and the placeholders were replaced with tokens from respective dictionaries. In total 11,27,571 questions were generated using templating approach.

4. SYSTEM ARCHITECTURE

The overall architecture of ABBA is represented in Figure 2. In QI, the user question is first passed through Period Identifier, then Condition Identifier and Filter Identifier modules which identify Temporal, Conditions and Filter attributes from it respectively. A detailed architecture of QI including ANEC is illustrated in Figure 3. The question from QI is then tagged with POS tags using Stanford POS tagger [10]. The POS tagged question is converted into a feature matrix which is then sent as an input to the BiLSTM model as highlighted in Figure 4.

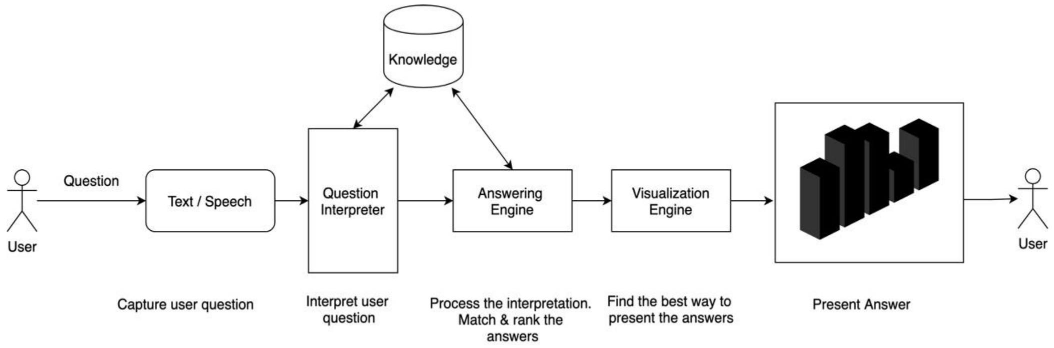


Figure 2. AI-based business analyst

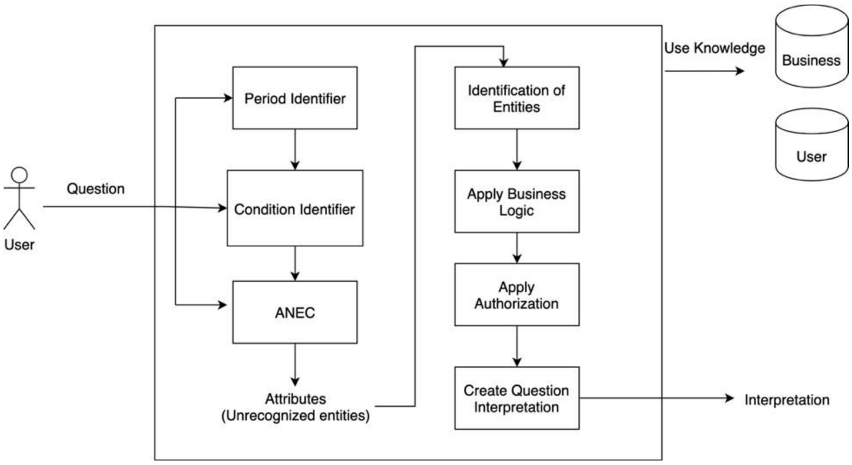


Figure 3. Question Interpreter

4.1. Feature Vector

The default tagset used for Stanford's English POS tagger is Penn Treebank Tagset [11] for POS tagging. These tags were grouped into 8 classes. In addition to these, 4 more classes were defined based on the knowledge dictionary in which each word or its lemma is found in.

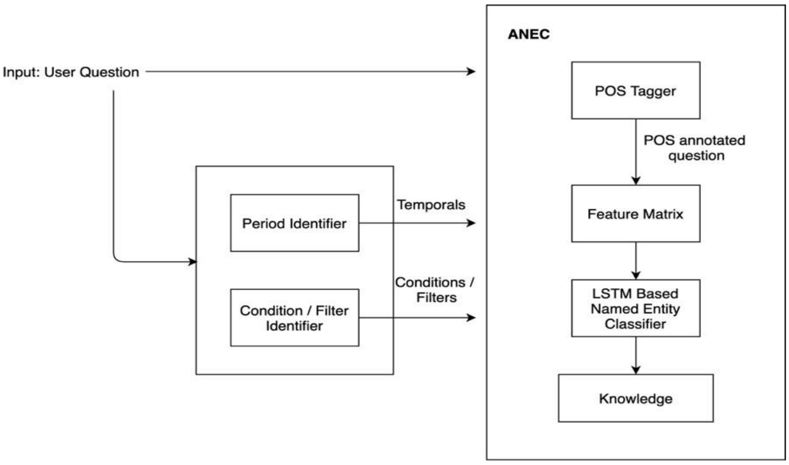


Figure 4. ANEC System Architecture

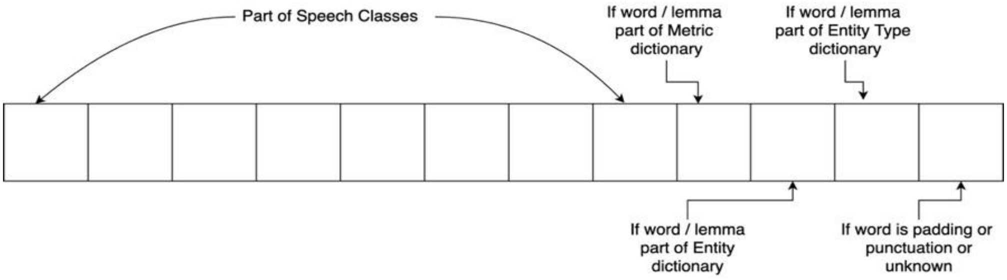

Part of Speech Classes
If word / lemma part of Metric dictionary
If word / lemma part of Entity Type dictionary
If word / lemma part of Entity dictionary
If word is padding or punctuation or unknown

Figure 5. Feature Vector of a word

Classes 9 to 11 are based on the 3 knowledge dictionaries namely - Entity, Entity Type and Metric. Class 12 indicates whether the word is a padding, punctuation or an unknown input. The 12 classes are listed in Table 3 and together they form a feature vector for each word as illustrated in Figure 5.

Table 3. Feature Vector class and their corresponding POS/Dictionary Tags

Feature Vector class	**POS / Dictionary Tag**
Class 1	NNP (Proper noun, singular), NNPS (Proper noun, plural)
Class 2	NN (Noun, singular or mass), NNS (Noun, plural)
Class 3	VB (Verb, base form), VBD (Verb, past tense), VBG (Verb, gerund or present participle), VBN (Verb, past participle), VBP (Verb, non-3rd person singular present), VBZ (Verb, 3rd person singular present)
Class 4	JJ (Adjective), JJR (Adjective, comparative), JJS (Adjective, superlative)
Class 5	CC (Coordinating conjunction), DT (Determiner), EX (Existential *there*), FW (Foreign word), IN (Preposition or subordinating conjunction), PDT (Predeterminer), POS (Possessive ending), PRP (Personal pronoun), RB (Adverb), RBR (Adverb, comparative), RBS (Adverb, superlative), RP (Particle), TO (to) , UH (Interjection)
Class 6	Alphanumeric and CD (Cardinal number)
Class 7	SYM (Symbol), LS (List item marker)
Class 8	WP (Wh-pronoun), WP$ (Possessive wh-pronoun), WRB (Whadverb), MD (Modal), WDT (Wh-determiner)
Class 9	Entity
Class 10	Entity Type
Class 11	Metric
Class 12	Padding, Unknown, Punctuation

4.2. Output

The output for each token from the BiLSTM model is a vector having 6 classes which is reflected in Figure 6. Each of these classes reflect the probability of a word belonging to a particular category with respect to the named entity classification task. The word is tagged with the class having the maximum probability.

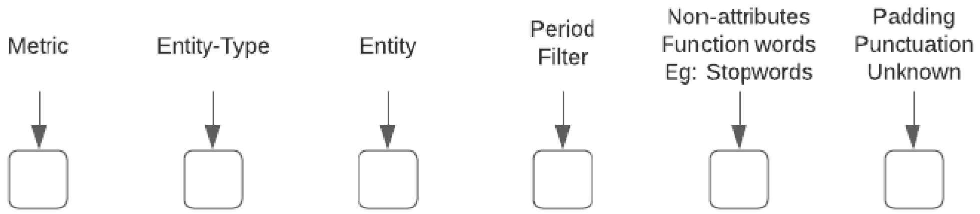


Figure 6. Output Vector of a word

4.3. Model Details

A Bi-Directional Recurrent Neural Network with Long Short-Term Memory units is used to predict the named-entity classes from the feature vector. The output of each network for each token passes through a softmax layer to give a probability for each named-entity class. An overview of the flow of data is highlighted in Figure 7.

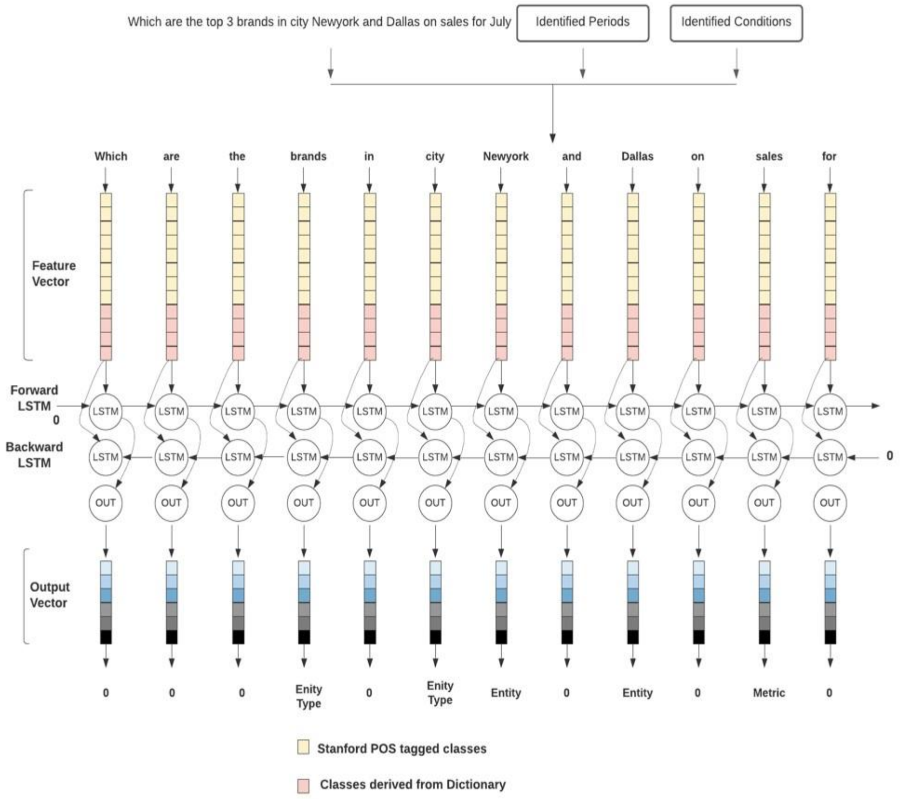


Figure 7. An overview of a question going through BiLSTM Model

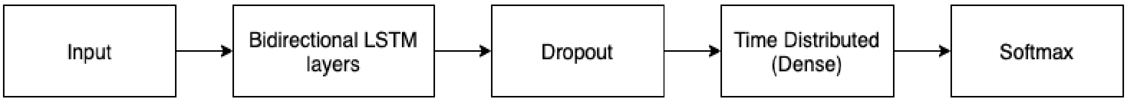


Figure 8. Network Model

The model was implemented in Keras [12] with a TensorFlow [13] backend. The network model is highlighted in Figure 8. The training and hyper-parameters are highlighted in Table 4.

4.4. Knowledge Query

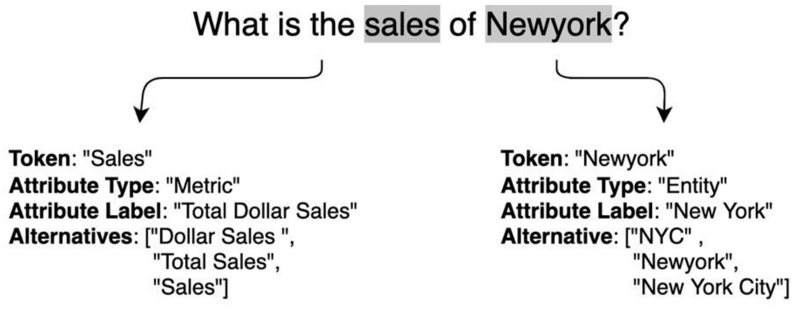
Knowledge Query refers to a query made to the Knowledge Dictionary for linking attribute tokens with their labelled entries in the database.

As mentioned, the output from the BiLSTM model is a vector having 6 classes, each of which reflects the probability of a word belonging to a particular category with respect to the named entity classification task. It is difficult to form a relation between collocated attribute words classified by the BiLSTM model as they have no significant meaning of their own in the absence of any link with the knowledge base. This can be explained with two following scenarios:

Table 4. Model Hyper-Parameters

Parameter	**Value**
Learning rate	0.001
Epoch	40
Batch Size	32
Dropout	0.2
LSTM Units	128
LSTM layers	1

- **Scenario 1:** For multiple consecutive words ‘*United*’, ‘*States*’ and ‘America’, individually identified as attribute of type Entity, the challenge is to determine that the three Entities occur together as a phrase and refer to a nation.
- **Scenario 2:** In some cases, the same word may refer to different attributes in the knowledge dictionary. For example, the word ‘*sales*’ can refer to a metric ‘*Items Sold*’ as well another metric *‘Unit Sales’*. Hence, for disambiguation and to establish a relationship with other attributes, we need to perform a Knowledge Query.



What is the sales of Newyork?

Token: "Sales"
Attribute Type: "Metric"
Attribute Label: "Total Dollar Sales"
Alternatives: ["Dollar Sales ",
"Total Sales",
"Sales"]

Token: "Newyork"
Attribute Type: "Entity"
Attribute Label: "New York"
Alternative: ["NYC" ,
"Newyork",
"New York City"]

Figure 9. Identifying actual entities in the database from Attribute tokens

An example of the Knowledge Query process is highlighted in Figure 9. The highlighted words refer to the named entities identified by the BiLSTM model as belonging to an attribute class. The 3 steps involved in this process are:

- Grouping
- Disambiguation
- Query

In **grouping**, the tokens belonging to the same attribute category are grouped together as one and successive knowledge queries are performed to extract entity names for the group of tokens. Primary assumptions in grouping process are:

- Successive words labelled with the same attribute class are grouped together. For example, ‘*Greater*’, ‘*Cincy*’, and ‘*East*’ are all entities. If they occur consecutively as ‘*Greater Cincy East’* in a sentence, they would be grouped together.

- Successive words of the same attribute class, when separated by a single non-named entity, will also be assigned the same group. For example, ‘*Portland, Oregon*’ (Entity) and ‘*Number of Cars Serviced*’ (Metric). This helps account for presence of punctuation marks and stop-words in labels of attributes. One limitation of this assumption is that instances of two distinct attributes occurring together with a coordinating conjunction or a comma might end up being grouped together. For example, ‘*New York City and Dallas*’ and ‘*New York City, Dallas*’. Here, ‘*New York City*’ and ‘*Dallas*’ refers to the names of two different cities and yet they get grouped together due to this assumption. Such instances are handled by a separate disambiguation algorithm discussed next.

Disambiguation is performed using repeated calls to the Knowledge Dictionary. The Disambiguation Algorithm works as shown in Listing 1. It is also illustrated in Figure 10. The example illustrated is that of the phrase ‘*Chicago, Dallas, Texas and San Francisco, California*’. The phrase contains three Entities, ‘*Chicago*’, ‘*Dallas, Texas*’ and ‘*San Francisco, California*’ First, all forms of stopwords and punctuation marks are removed. Then, a **query** is made for words starting from the end of the string, one by one into the knowledge dictionary. A successful response means that a match for a particular phrase exists in the Knowledge Dictionary. An empty response indicates that no match was found. In the example, first query is made for the word ‘*California*’. After receiving a successful response for it, a subsequent query is made for this word along with the word preceding it. Thus, after querying ‘*Francisco California*’ a successful response is received again and next query is made for ‘*Francisco California*’ by preceding it with ‘*San*’ for which a successful response is received as well. Next, on querying ‘*Texas San Francisco California*’ an empty response is received, indicating that this search phrase does not exist in the Knowledge Dictionary. Now, previously stored response of ‘*San Francisco California*’ (highlighted in Magenta) is saved and the remaining words are sent back for disambiguation. The function starts querying again from ‘*Texas*’, and moves on to ‘*Dallas Texas*’ before receiving an empty response at ‘*Chicago Dallas Texas*’.

Listing 1: Disambiguation Algorithm

~~~
def disambiguate(tokens):
```Disambiguate algorithm for attribute type entity```
#marker: stores position from end of string corresponding to last positive response
#out: stores output response from the present knowledge query marker = 1 result = [] for i in
range(1, len(tokens) + 1):
   out = knowledgeCall(tokens[-i: ], “entity”)     if out != []:        marker = i        results = out if
marker == len(tokens):
   return([[tokens[-marker: ], result]]) else:
   return(disambiguate(tokens[ :-marker]) + [[tokens[-marker: ],
result]] )
~~~

It saves the stored response for ‘*Dallas Texas*’ (highlighted in Red) and moves on to query

'*Chicago*'. The recursive function finally returns all the valid word groups along with their actual labelled entries in the knowledge dictionary.

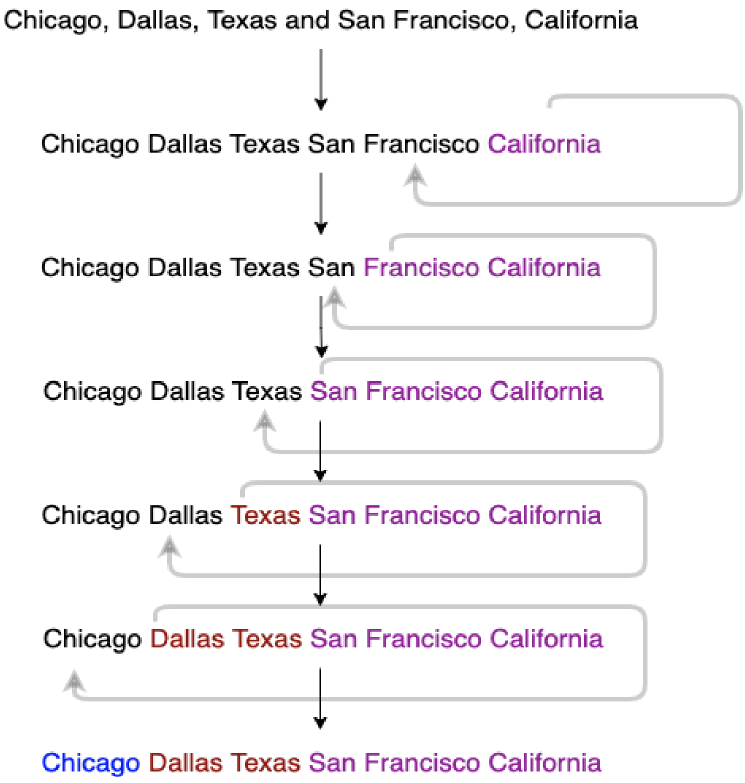

Chicago, Dallas, Texas and San Francisco, California
Chicago Dallas Texas San Francisco California
Chicago Dallas Texas San Francisco California
Chicago Dallas Texas San Francisco California
Chicago Dallas Texas San Francisco California
Chicago Dallas Texas San Francisco California
Chicago Dallas Texas San Francisco California

Figure 10. Illustration of Disambiguation Process

5. EVALUATION

The training, validation and test dataset followed a 60:20:20 split on the questions generated by templating and the system was evaluated on gold dataset of 120 instances with many simple to complex cases created by business analysts. The metrics were calculated at question-level followed by calculation at dataset-level. Each question was evaluated based on the entities (Metric, Entity, Entity Type, Temporal etc) and sibling-relations (Metric-Condition, Filter-Entity Type etc) identified. For each question, the following 3 lists were captured based on MUC evaluation metrics described by [14].

- **Matches:** The entities and sibling-relations that are matched perfectly from both the predicted list and the ground-truth list
- **Spurious:** The entities and sibling-relations that are present in the predicted list, but not in the ground-truth list
- **Missing:** The entities and sibling-relations that are present in the ground-truth list, but not in the predicted list

The evaluation is done using the following 3 metrics using the above lists:

- Precision: |Matches|/(|Matches| + |Spurious|)
- Recall: |Matches|/(|Matches| + |Missing|)

- F1-score: $\frac{2 \times Precision \times Recall}{Precision + Recall}$

The dataset-level metrics are computed from aggregating question-level metrics by computing their micro-averages and finally the accuracy for the dataset is computed as follows:

$$Accuracy = \sum_i \frac{I_A F_i}{n}, \text{ where}$$

1. F(i) is the F1-score of the ith interpretation
2. I(A) is an indicator function that returns 1 if F(i) = 1, else 0
3. n is the total number of samples in the dataset.

Table 5. Performance scores on Gold Dataset

Metric	**Partial Comparison**	**Strict Comparison**
Precision	0.979	0.979
Recall	0.979	0.979
F1-score	0.979	0.979
Accuracy	0.987	0.987

The evaluation is done based of 2 types of comparisons between the predictions and ground truth. The 2 comparators used for this purpose are:

- **Strict comparator:** All properties of the entities and sibling-relations must be equal for two entities and sibling-relations to be considered equal.
- **Partial comparator:** Even if the span (start and end indices) of an entity or siblingrelation is incorrect, as long as the other properties of the entity or sibling-relation is correct, they are considered to be equal.

The accuracy was measured in terms of number of questions with all the attributes classified correctly. Instances of questions where entities were classified partially, were treated as an incorrect classification. The result of the experiment is reported in Table 5.

We have not published the performance of other popular Named Entity Recognizers in comparison against our system. Firstly, this is because ANEC was built to classify attributes in a closed business domain whereas other NERs were built for more general open domain tasks. Secondly, by classifying attributes with ANEC helps us save a large number of calls to the Knowledge Dictionary. In case of a standard NER, we have to make a large number calls to the Knowledge Dictionary just to determine which class of Attributes a phrase belongs to. Hence, comparing two systems aimed at different domains would not be a fair comparison and the results would be highly biased towards ANEC.

6. RESULTS AND ANALYSIS

We have divided the results in the following three categories:

- Attributes present in single knowledge dictionary
- Attributes present in multiple knowledge dictionaries
- Attributes not present in any knowledge dictionary

Following subsections present the results and examples from each category.

6.1. Attributes Present in Single Knowledge Dictionary

The system was able to identify and classify all attributes which were present in a single knowledge dictionary. For example - *Sales* and *Sales Achievement* were present only in the Metric dictionary. Similarly, words like *Region* and *Store* were present only in Entity Type knowledge dictionary.

6.2. Attributes Present in Multiple Knowledge Dictionaries

For attributes present in multiple dictionaries, the system was able to classify attributes correctly when the attribute word was present with other words as an attribute. For example, the word *Customer* was present in the Entity Type dictionary as *Customer Segment*, whereas it was present in the Entity dictionary as *Customer 01*. For ambiguous cases like *Customer,* it was easy to both identify as well as classify them with the help of adjacent words.

In case of words wholly occurring in multiple dictionaries, it becomes difficult to classify them. For example, *Target* occurs by itself, both as a Metric as well as an Entity. Another example is the word *Store*, which is used both as a Metric as well as an Entity Type. In such cases, the system is able to identify named attributes but is unable to classify them with sufficient confidence. A few such ambiguous questions are highlighted below.

1. What is the **Target** and **Sales** for West Region?
2. What is the **Sales** for **Target** for West Region?

In the first example, the word *Target* is a Metric along with the word *Sales*. In case of second example, the word *Target* is an Entity whereas the word *Sales* remains a Metric. In such cases, our system fails to classify the target with sufficient confidence.

6.3. Attributes not Present in Any Knowledge Dictionary

For words which do not occur in any of the attribute dictionaries, it is crucial that we identify them even though there is no way to classify them correctly. A few examples of such tokens are *Performance* and *Productivity* which were not present in any knowledge dictionary but had some significance in the business sense. Our system was able to identify them, but, since they were not found in the knowledge, they were marked as unrecognized attributes.

7. CONCLUSION

In this paper, we utilize our Recurrent Neural Network with BiLSTM units to identify and classify named entities in natural language questions. We have also provided an overview of the techniques employed to develop a Neural Network based NER in context of an AI Based

Business Analyst. Availability of a large collection of annotated data was very important to train a deep learning model, so this paper also discussed about templating approach which was used to create a large training dataset from a small sample set of 1,442 questions.

While our BiLSTM model is effective in identifying and classifying a large majority of questions, it falls a tad bit short when it comes to identifying context in very complex cases as highlighted in section 6.2. The resolution of such ambiguities needs extensive research into business attributes and how they are linked together by various stop-words and function-words. An extension of the ANEC pipeline can be inclusion of a spell-checker. Attributes with spelling mistakes usually get added to the list of unrecognized entities. A spell-checker module can help us identify attributes present in the knowledge dictionary from the list of unrecognized attributes. Another addition can be usage of word embeddings as features for better classification of attributes that are not present in any knowledge dictionary.

ACKNOWLEDGEMENTS

We would like to thank all the members of Crux Intelligence for their support and contribution. We would also like to thank Samriddhi Sinha for his valuable contribution in building the system.

REFERENCES

[1] Joshi, A.; Prabhugaonkar, N.; Hadaye, R.; Unnikrishnan, S.; Das, S.; Gala, N.; Bari, P.; and Mall, N. 2019. 'I Under- stand What You Asked': Question Interpreter for an AI- Based Business Analyst. In Arai, K.; Kapoor, S.; and Bhatia, R., eds., Intelligent Systems and Applications, 1282–1288. Cham: Springer International Publishing. ISBN 978-3-030- 01057-7.

[2] Dhamdhere, K.; McCurley, K.; Nahmias, R.; Sundararajan, M.; and Yan, Q. 2017. Analyza: Exploring Data with Conversation. *Proceedings of the 22nd International Conference on Intelligent User Interfaces.*

[3] Finkel, J. R.; Grenager, T.; and Manning, C. 2005. Incorporating Non-Local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, 363–370. USA: Association for Computational Linguistics.

[4] Honnibal, M.; and Montani, I. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

[5] Feng, Z.; Wang, Q.; Jiang, W.; Lyu, Y.; and Zhu, Y. 2020. Knowledge-Enhanced Named Entity Disambiguation for Short Text. In *Proceedings of the 1st Conference of the Asia- Pacific Chapter of the Association for Computational Lin- guistics and the 10th International Joint Conference on Nat- ural Language Processing*, 735–744. Suzhou, China: Asso- ciation for Computational Linguistics.

[6] Hochreiter, S.; and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*, 9: 1735–1780.

[7] Schuster, M.; and Paliwal, K. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, 45: 2673– 2681.

[8] Chiu, J. P. C.; and Nichols, E. 2016. Named Entity Recognition with Bidirectional LSTM CNNs. *Transactions of the Association for Computational Linguistics*, 4: 357–370.

[9] Ma, X.; and Hovy, E. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1064–1074. Berlin, Germany: Association for Computational Linguistics.

[10] Manning, C.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S.; and McClosky, D. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 55–60. Baltimore, Maryland: Association for Computational Linguistics

[11] Marcus, M.; Kim, G.; Marcinkiewicz, M. A.; MacIntyre, R.; Bies, A.; Ferguson, M.; Katz, K.; and Schasberger, B. 1994. The Penn Treebank: Annotating Predicate Argument Structure. In*Proceedings of the Workshop on Human Language Technology*, HLT '94, 114–119. USA: Association for Computational Linguistics. ISBN 1558603573.

[12] Chollet, F.; et al. 2015. Keras. https://github.com/fchollet/ keras.

[13] Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Leven- berg, J.; Mane´, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar,K.;Tucker,P.;Vanhoucke,V.;Vasudevan,V.;Vie´gas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; and Zheng, X. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

[14] Chinchor, N.; and Sundheim, B. 1993. MUC-5 Evalua- tion Metrics. In *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993.*

ONE-CLASS MODEL FOR FABRIC DEFECT DETECTION

Hao Zhou, Yixin Chen, David Troendle, Byunghyun Jang

Computer Information and Science, University of Mississippi, University, USA

ABSTRACT

An automated and accurate fabric defect inspection system is in high demand as a replacement for slow, inconsistent, error-prone, and expensive human operators in the textile industry. Previous efforts focused on certain types of fabrics or defects, which is not an ideal solution. In this paper, we propose a novel one-class model that is capable of detecting various defects on different fabric types. Our model takes advantage of a well designed Gabor filter bank to analyze fabric texture. We then leverage an advanced deep learning algorithm, autoencoder, to learn general feature representations from the outputs of the Gabor filter bank. Lastly, we develop a nearest neighbor density estimator to locate potential defects and draw them on the fabric images. We demonstrate the effectiveness and robustness of the proposed model by testing it on various types of fabrics such as plain, patterned, and rotated fabrics. Our model also achieves a true positive rate (a.k.a recall) value of 0.895 with no false alarms on our dataset based upon the Standard Fabric Defect Glossary.

KEYWORDS

Fabric defect detection, One-class classification, Gabor filter bank.

1. INTRODUCTION

Fabric inspection plays a critical quality control role in the textile industry. Trained human inspectors conduct quality inspections to find any potential fabric defects. However, due to the inherent limitations of human labor such as eye fatigue and distraction, the process is considered time-consuming, inconsistent, error-prone, and expensive. Thus, in order to improve this important process, an automated and accurate inspection system is highly desired. However, developing such an automated inspection system is technically challenging mainly due to the following two challenges.

Challenge 1: Defects in fabrics vary in their size and shape. Further, fabric defects are usually caused by machine malfunctions such as weaving and painting issues, and they are relatively rare in practice, implying that the number of error-free fabrics (referring as non-defective, positive, or normal) is significantly larger than the number of defective fabrics (referring as defective or negative).

To deal with many different types of defects, researchers have built detection systems based on multi-class classification [1] [2]. However, such systems do not work well beyond the pre-defined classes and suffer from inherent unbalanced dataset issues (e.g., the number of instances of some defects is larger than that of other defects).

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 177-189, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112314

Challenge 2: In addition to the variations in defects, background textures also vary in their patterns, based on different weaving methods and painting patterns (e.g., plain, patterned, etc). They are categorized into over 70 kinds [3]. Building a system that analyzes all kinds of fabrics is a challenging task.

Some researchers took advantage of Gabor filters to analyze specific types of fabrics with optimized filter parameters [4] [5]. However, different optimization approaches and objectives yield different results, making it difficult to apply in the real world. More importantly, an optimized Gabor filter only works for one or a few types of fabrics. In other words, covering a wide range of fabrics requires an optimized Gabor filter bank.

To address the aforementioned challenges, this paper proposes a novel one-class model for fabric inspection. The model is based on one-class classification (OCC) and Gabor filters. For challenge 1, we leverage one-class classification, which is helpful when negative samples are absent or not well-defined. In OCC, an efficient classifier is able to define a classification boundary with only the knowledge of the positive class [6]. Therefore, OCC simplifies the problem to focusing only on positive fabrics. This eliminates the need to collect, deal with various fabric defects and worry about imbalanced datasets or lack of negative samples. For challenge 2, we carefully design a Gabor filter bank with different orientations and bandwidths to analyze different fabric textures. By doing so, we ease the development of an optimized Gabor filter design for each fabric texture. In order to define a good classification boundary, we design an autoencoder to maximize representative features from the output of the Gabor filter bank. Finally, we leverage a nearest neighbor density estimator to locate defects and draw detective areas on fabric images. To validate our proposed one-class model, we extensively test the model from different aspects. We focus on the effectiveness of the model on plain, patterned, and rotated fabric images, and also the advantages of the autoencoder as a feature selector compared against a hand-crafted method, a Principal Component Analysis (PCA) method, and another advanced deep learning algorithm. We also investigate the model's performance when parameters of our model change. The experiments show that our proposed model effectively detects defects on fabric images with various background textures and achieves a true positive rate value of 0.895 with no false alarms on a selected dataset from the Standard Fabric Defect Glossary [7].

The rest of the paper is organized as follows. Section 2 summarizes efforts on the problem, Section 3 details the overall design of our model, Section 4 shows our experiments from different aspects, and Section 5 concludes our paper.

2. RELATED WORKS

Researchers have proposed numerous systems and algorithms to automate the fabric defect detection problem. These methods can be generally classified into four types of approaches - statistical, structural, model-based, and spectral. We briefly introduce these approaches with more focus on relevant spectral approaches. We also introduce several advanced deep learning algorithms for the problem toward the end of this section.

Statistical approaches [8] [9] usually leverage first- (e.g., mean and standard deviation) or second-order statistics (e.g., correlation method). These statistics are used to represent the color information of fabrics. However, statistical information extracted only from raw images (color or grey-level images) is not enough to fully represent fabric features, not being able to distinguish between fabric features and defects, especially in patterned cases. Secondly, the structural approach [10] works well only on plain fabrics as the patterns of plain fabrics are easier to analyze and retrieve. When encountering complex patterns, the structural approach is not a viable solution. Thirdly, model-based approaches (e.g., Markov random field) usually explore the relation among pixels of fabric images. As with the statistical approaches, model-based approaches tend to ignore small defects, which degrades the applicability of the approach. Lastly, spectral approaches based on Fourier, wavelet, or Gabor transforms are superior to other approaches as they can extract texture features by analyzing fabrics in the frequency domain, which in return, are less sensitive to noise compared to ones in the spatial domain (e.g., statistical approach).

Gabor filters have gained popularity for texture analysis due to their ability to model the human visual cortex cells. In the literature, Gabor filter-based approaches for fabric defect detection are based either on optimized Gabor filters or on a set of Gabor filters called a Gabor filter bank. Tong et al. [4] optimize a few Gabor filters by utilizing composite differential evolution. In return, the algorithm can successfully segment the defects from the fabric images. Mak et al. [5] also optimize Gabor filters by a genetic algorithm. It should be noted, however, optimized filters work only on certain types of fabrics. Kumar et al. [11] propose a Gabor filter bank with different scales and orientations. The Gabor filter bank covers multi-resolution and can detect the features of defects by fusing outputs of Gabor filters. Even though a Gabor filter bank deals with different types of fabrics, simply fusing the outputs of filters may intensify noise in normal fabrics since each filter extracts features from one pair of scales and orientations. Unfortunately, the outputs of a Gabor filter bank are significantly bigger (e.g., the dimension of feature vectors is large). Bissi et al. [12] perform Principal Component Analysis (PCA) on the outputs of a Gabor filter bank which greatly reduces the dimension of feature vectors. However, during the process, information loss can occur if the number of principal components is not selected carefully.

Researchers have also tried to solve the fabric defect detection problem by utilizing advanced deep learning algorithms. Some treat the problem as an object detection problem. Zhou et al. [2] build a system by incorporating several techniques to the vanilla Faster RCNN to locate and classify defects. Zhang et al. [1] build a detection system based on YOLO by extensively comparing among different YOLO frameworks. These algorithms are supervised, which implies that in order to obtain a well-trained model, a balanced and large dataset is a must. However, this is difficult to achieve in practice. Moreover, defect variations in terms of size and shape, and texture background make these algorithms vulnerable.

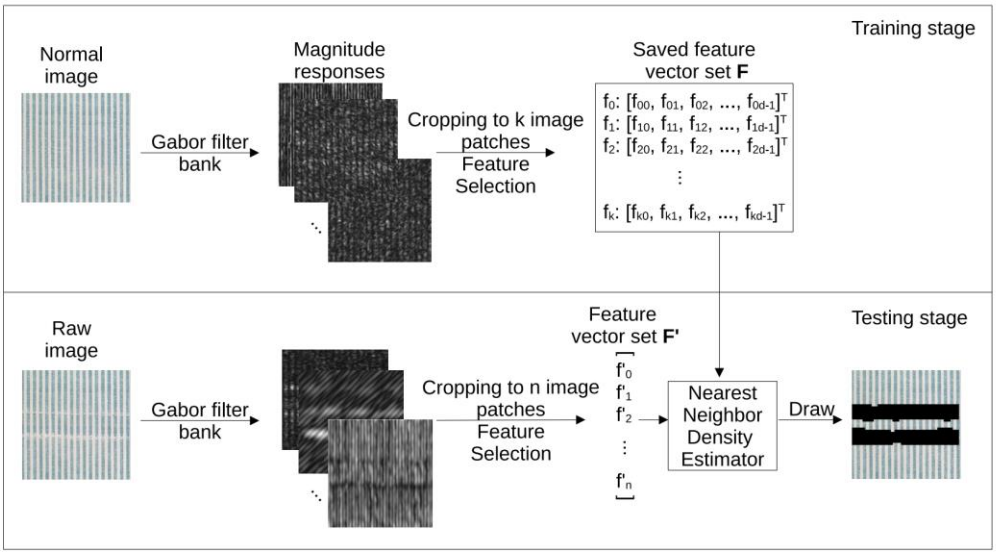

Training stage
Normal image
Gabor filter bank
Magnitude responses
Cropping to k image patches
Feature Selection
Saved feature vector set **F**
f_0: [f_{00}, f_{01}, f_{02}, ..., f_{0d-1}]^T
f_1: [f_{10}, f_{11}, f_{12}, ..., f_{1d-1}]^T
f_2: [f_{20}, f_{21}, f_{22}, ..., f_{2d-1}]^T
f_k: [f_{k0}, f_{k1}, f_{k2}, ..., f_{kd-1}]^T
Testing stage
Raw image
Gabor filter bank
Cropping to n image patches
Feature Selection
Feature vector set **F'**
f'_0
f'_1
f'_2
f'_n
Nearest Neighbor Density Estimator
Draw

Fig. 1. Overall model design.

3. DESIGN

In this section, we overview the design of our proposed model for the fabric defect detection problem. As shown in Fig. 1, the model has two stages - training and testing stages. The model in the training stage takes normal images (i.e., images without defective areas) as input, and applies a Gabor filter bank with different orientations and bandwidths. The filtered images then are cropped into small image patches (e.g., k image patches) and a technique of feature selection is applied on these images patches. Each feature vector is D dimension and the generated feature vectors are saved for the testing stage. In the testing stage, the model takes a raw image (either a normal or a defective image) as input, applies the same Gabor filter bank to the image, and crops the filtered images into image patches. After the same feature selection, a density estimator takes the saved feature vector and the new feature vector to determine an image patch is either normal or defective. Lastly, the determined defective image patches are painted on the raw image. In the rest of this section, we detail our Gabor filter bank, the feature section technique, and our density estimator.

3.1. Gabor Filter Bank

Gabor filter is a good model of simple cells in the human visual cortex and is capable of representing textures very well [13]. A Gabor filter is the product of a sinusoid and a Gaussian:

$$g(x, y; \lambda, \theta, \phi, \gamma) = \exp\left(- \frac{x'^2 + \gamma^2 y'^2}{2\sigma^2} \right) \cos\left(2\pi \frac{x'}{\lambda} + \phi \right),$$

where λ is the wavelength of the Gabor filter (e.g., the number of cycles/pixel), θ is the orientation of the Gabor filter (e.g., the angle of the normal to the sinusoid), ϕ is the phase (e.g., the offset of the sinusoid), γ represents aspect ratio, σ is the spatial envelope of the Gaussian and is controlled by the bandwidth (e.g., $\sigma = 0.56\lambda$), $x' = x \cos \theta + y \sin \theta$, and $y' = -x \sin \theta + y \cos \theta$.

In the fabric defect detection problem, the wavelength λ and orientation θ are more important as fabrics can be orientated differently and the width of yarns varies. So, the design challenge is to find a good pair of λ and θ. However, due to the varieties of fabrics, it is not possible to find a perfect pair of λ and θ that works well for all types of fabrics. Recently, researchers tried to optimize Gabor filters for each type of fabrics [4] [14], but it is unrealistic when considering a real-world situation where a type of unseen fabric continuously presents. Without losing generality, we propose a Gabor filter bank with different wavelengths and bandwidths that can cover most fabrics. We have a set of wavelengths $\{\lambda: \lambda = 2m \text{ where } 2 \leq m \leq 6\}$ and a set of orientations $\{\theta: \theta = 10n, \text{ where } 0 \leq n \leq 18\}$. Thus, our Gabor filter bank G consists of a total of 90 Gabor filters (e.g., $g_0, g_1, ..., g_{89}$). When filtering the input image I, with the Gabor filter bank G, the magnitude responses G′ can be represented as the following.

$$\{G': G'_i = I \circledast g_i\},$$

where $0 \leq i \leq 89$. Note that the filtered images G′ have the same size as the image I (512×512 in our dataset).

3.2. Feature Selection

Magnitude responses G′ are the features filtered by the Gabor filter bank and their sizes are large (e.g., 512×512) since we propose a Gabor filter bank to increase the generality. It is possible that some filters have the same or similar effects when filtering the input image. In this case, the magnitude responses would present redundant features, which decreases the performance and increases testing time in our model. So in order to speed up the model, especially the density estimation process (e.g., determining defective image patches), more general and representative features are needed. To this end, we propose an autoencoder feature learning neural networks method, inspired by the development of deep learning algorithms [15].

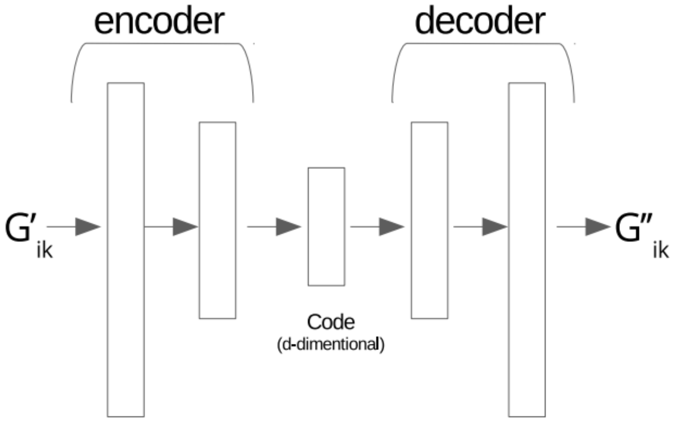


Fig. 2. Structure of autoencoder.

In our model, instead of manually determining what statistical data we use or what features are more general, we rely on autoencoder neural networks to find the more representative features from the magnitude responses. We found that an autoencoder with a simple structure is capable of finding the representative features from the magnitude responses. The structure is shown in Fig. 2. By trying to reconstruct G'_{ik} and minimizing the distance between G'_{ik} and G''_{ik} over and over again, the autoencoder neural network is able to learn the best code to represent the input response. Note that we train the autoencoder neural network only with the responses from normal fabric images. After training, whenever input is fed to the autoencoder, the code is the representative feature vector of the input.

One may question why we don’t apply autoencoders on fabric images directly, or why we don’t take advantage of transfer learning [16]. One reason we don’t directly train autoencoders on fabric images is that we don’t have a large and balanced dataset. If we train an autoencoder directly on fabric images, the convergence of our autoencoder is an issue. Transfer learning might be a good solution since the previous study [17] shows that the weights of the first layer in convolutional neural networks are indeed quite similar to Gabor filters. However, one reason we don’t transfer weights from well-trained models is that the choice of models is limited and some requirements (e.g., input size, number of layers, etc.) must be met. Moreover, to our best knowledge, there are no pre-trained models for the problem of fabric defect detection so that transferring weights from quite different datasets might hurt the performance of our design as well.

3.3. Nearest Neighbor Density Estimator

By carefully observing fabric images, one can easily find that most defects are different from the fabric's background. This provides us a good opportunity to classify fabric image patches into either defective or normal ones. To this end, we propose a Nearest Neighbor Density Estimator (NNDE). NNDE is able to identify defective fabric patches at the testing stage based on the saved feature vectors from the training stage as in Fig. 1. Given a feature vector f'_i (note that a feature vector represents an image patch) and the saved feature vector set *F*, the estimator can be mathematically defined as the following.

$$s_i = \frac{Dist(f'_i, F)}{Dist(f_j, F)}, f'_i \in F', f_j \in F,$$

where *Dist(·)* returns the distance between the feature vector and its nearest neighbor in *F* and s_i is proportional to the likelihood ratio. Furthermore, a feature vector f'_i is classified into either the normal or the defective class based on a threshold τ as the following.

$$f'_i = \begin{cases} Defective & \text{if } s_i > \tau \\ Normal & \text{otherwise} \end{cases}$$

Note that τ must be equal to or greater than one. Having a bigger value indicates the model allows more defective patches to be classified as normal patches. Empirically, we set τ to 1.05.

4. RESULTS AND EVALUATION

We conducted experiments to validate the effectiveness of the proposed model. First, we introduce our experiment environment, dataset, and metrics we used for quantitative analysis. Then, we present accuracy results along with output visualization. Next, we present the effectiveness of our model on two different choices of feature selection - hand-crafted [8], PCA [18], and residual neural networks [19] methods. We also explore the impact of two important parameters - image patch size and Gabor filter bank parameters on speed and accuracy (conditioned true positive rate). Last, we demonstrate the robustness of our proposed model by testing rotated fabric images.

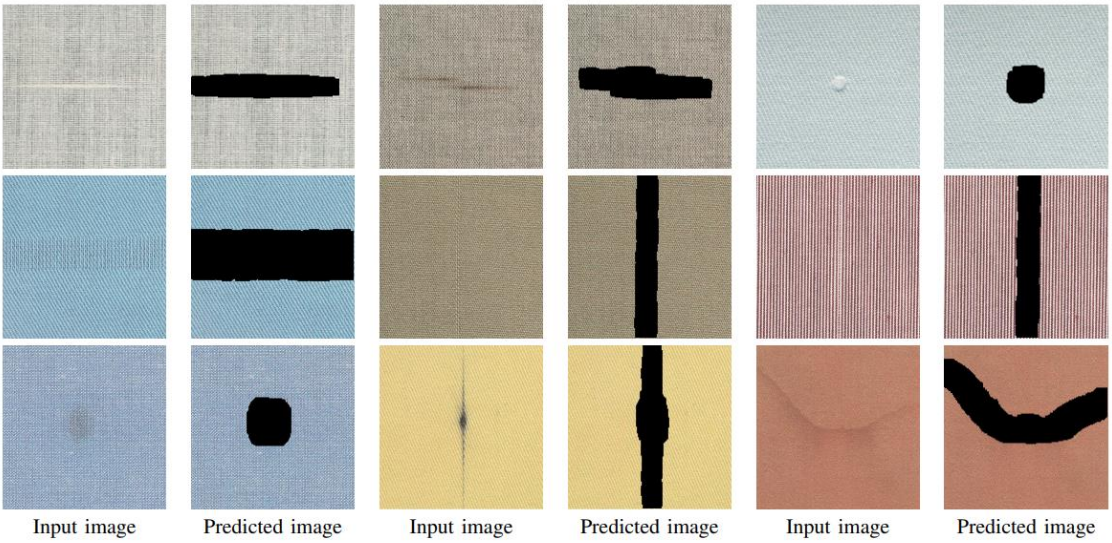


Fig. 3. Visual outputs of plain fabric defect detection (1st, 3rd, 5th columns present input images and 2nd, 4th, 6th columns show defects detected in black).

4.1. Experiment environment, dataset, and metrics

We implement the model in Python. The testing environment is configured with AMD A10-6800K APU with a 2.5 GHz frequency. Our dataset contains fabric images that are collected from the Standard Fabric Defect Glossary (SFDG) [7] dataset that consists of more complex fabric images than other simple datasets such as the TILDA Textile Texture Database [20]. Because the SFDG dataset contains a lot of duplicated fabric images, we then select 31 representatives that cover most of cases in the SFDG dataset (i.e., every 512×512 fabric image differs in fabric backgrounds, defects, colours, etc). We hope that our proposed model can solve the defect detection problem on these images so that we show the generosity of our model when compared with other models that work for certain types of fabrics. In the experiments, we use several metrics to quantify the performance of our model, such as true positive rate (TPR), false positive rate (FPR), and receiver operating characteristic curve (ROC curve). TPR is defined as TP/(TP + FN) and FPR is defined as FP/(TP + FN) where TP, TN, FP, FN indicate true positive (defect samples classified as defects), true negative (normal samples classified as normal), false positive (normal samples classified as defects), and false negative (defect samples classified as normal), respectively. By these metrics, we are able to measure the model's ability to distinguish between the classes. In particular, we focus on TPR (recall) when FPR (ratio of false alarm) is equal to 0. We call this metric a conditioned TPR (cTPR). We value this metric since it can represent the real-world situation where a small portion of false alarms results in significant economic losses.

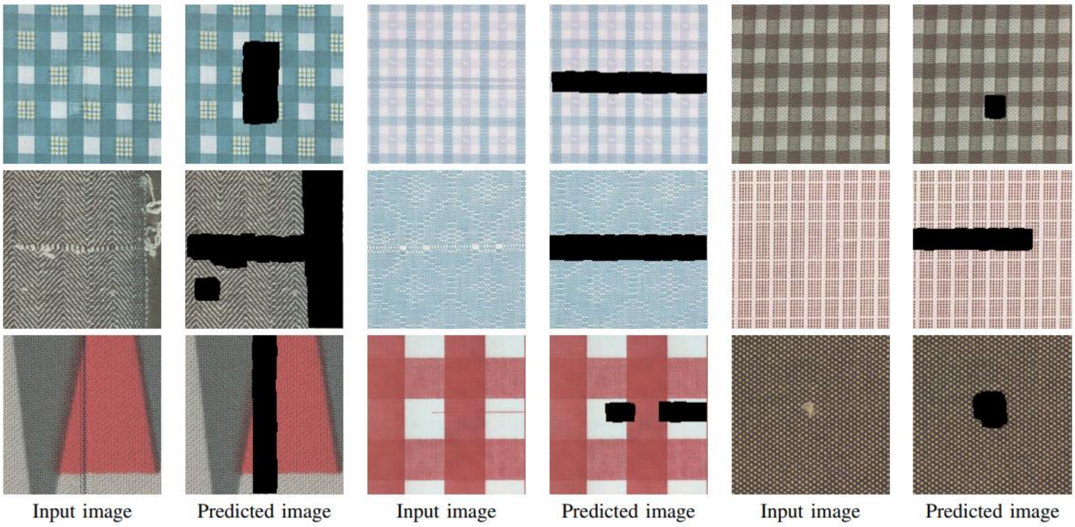


Fig. 4. Visual outputs of patterned fabric defect detection (1st, 3rd, 5th columns present input images and 2nd, 4th, 6th columns show defects detected in black).

4.2. Visualized Results

We visualize the defect detection results of plain and patterned fabric images selected from the dataset in Fig. 3 and Fig. 4 respectively. Our proposed one-class model successfully detects all defects successfully regardless of different fabric textures and shapes/sizes of defects. Fig. 4 shows our model works for patterned fabric as well as plain fabric. We analyze the reasons as follows. First, our Gabor filter bank is capable of extract texture features from different dimensions, which makes the model less insensitive to the changes in motif size. Second, our feature learning autoencoder makes the features extracted from the Gabor filter bank outputs more versatile. It makes the model effective in locating different kinds of defects.

Note that the defects detected are slightly larger in size than actual defects. This is because our proposed model predicts defects on the basis of image patches instead of pixels. Therefore, as the patches are larger in size, the defects marked tend to get larger. We don’t see this level of discrepancy as a problem in real world as defects would be cut off with some margins for safe quality control when found.

4.3. Comparison of different feature learning techniques

We compare the performance of our proposed autoencoder with the other two traditional feature selection methods, the hand-crafted [8] and PCA [18], to demonstrate the effectiveness of our proposed autoencoder. Regardless of the kinds of fabrics, our model achieves a higher value of cTPR and AUC (area under ROC curve) as shown in Table 1. Specifically, our autoencoder improves overall cTPR and AUC by 12.9% and 2.9% respectively. This suggests that by utilizing an autoencoder, our model can learn more representative and general features from the magnitude responses of the designed Gabor filter bank. The defective feature vectors, therefore, are more easily detected by the nearest neighbor density estimator. One may notice that AUCs don’t change much. Indeed, AUC measures the performance of the model across all different FPR values. Stable AUCs suggest that our model improves cTPR a lot by allowing some positive samples to be predicted as negative samples (FPR only increases a bit). This also implies some defects are very close to defective fabrics, making them very difficult to classify correctly (e.g., the fabric located at the 2nd row and the 5th column in Fig. 3). Our proposed model can still predict defects very well on all kinds of fabrics when there are no any false alarms arise.

Table 1. Comparison between different feature selections. Note that the numbers are averaged over all cTPRs and AUCs of 31 fabric images respectively.

	Hand-crafted [8]	PCA [18]	Ours
cTPR for plain fabrics	0.807	0.803	**0.907**
AUC for plain fabrics	0.953	0.951	**0.980**
cTPR for patterned fabrics	0.779	0.774	**0.882**
AUC for patterned fabrics	0.945	0.944	**0.974**
cTPR for al fabrics	0.793	0.788	**0.895**
AUC for all fabrics	0.949	0.947	**0.977**

4.4. Comparison against deep learning algorithms

Besides traditional feature learning methods, we also compare our autoencoder with broadly-used residual neural networks (ResNet) [19]. By end-to-end training a ResNet and a binary classifier using all 31 fabrics in our dataset, the ResNet model is able to perform fabric defect detection on all 31 fabrics with a 93.5% recall and an 83.6% precision. However, when coming to cTPR (i.e., no false alarms exist), the cTPR of the ResNet model is only about 0.217, which is far worse than ours (0.895). We conclude three reasons for it. Firstly, it is no doubt that the ResNet and its variants have gained lots of success in all kinds of computer vision tasks due to the powerful representation learning. It is also a fact that the bedrock for these successes is datasets where the number of samples and accurate labels are superior to our dataset (hundreds of thousands vs. 31). The quality of representation learning, thus, cannot be ensured. For example, unlike our Gabor filter bank and simple autoencoder, the convolutional filters of ResNet are learnable and these large number of parameters are unlikely to learn well without sufficient samples. Secondly, the sizes of defects are usually small compared with 512×512 images. Without having special attention to defects, ResNet models could just learn the representations of fabrics instead of defects. Also, defects have different types and to our best knowledge, there is no concrete number that says how many types of fabric defects exist. This makes training a deep learning classifier hard since current models are still learning from existing data and cannot infer well outside of the scope. However, our one-class model shows its strength by only learning from fabrics patterns. Lastly, the number of normal samples is way larger than that of defective samples, an unbalanced dataset makes training even harder as trained models can be misleading by just predicting all samples to be normal, which is also the main reason the cTPR is low.

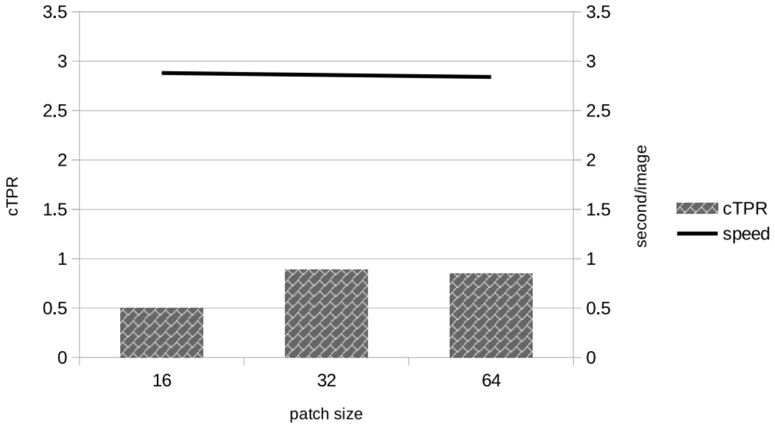
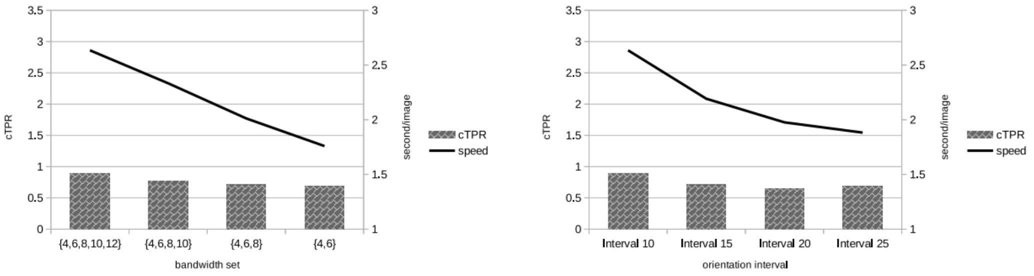


Fig. 5. The performance impact of patch size on cTPR and speed.

4.5. Impact on parameters

1) **Image patch size:** Image patch size is important in our design. Bigger image patches contain more information than smaller image patches. To find an appropriate size, we test our model on various image patch sizes to see how it affects cTPR and speed. Fig. 5 shows that cTPR increases as the patch size increases then decreases back. We believe that bigger image patch sizes may have more information which simply gets redundant after a certain size. Our autoencoder is able to process redundant information fairly well. Therefore, there is little gain overall unless it is too small. Fig. 5 also shows the speed decreases by a very limited amount as the patch size increases. If we take the speed of patch size 32 as a baseline, the difference is only about 0.6% (existing when cropping an image to patches and compressing by the autoencoder). So we claim that the speed is little affected by the patch size. Also bigger patch sizes make defects visualized larger than actual size. Therefore, we choose 32 as our patch size in the following experiments. However, one should know that the speed can be greatly improved if the number of patches per image decreases.



(a) Bandwidth

(b) Orientation

Fig. 6. The performance impact of parameters of Gabor filter bank on cTPR and speed.

2) **Orientations and bandwidths of Gabor filter bank:** One common issue of applying the Gabor filter bank is speed. While more orientations and bandwidths introduce more outputs of the Gabor filter bank, the processing speed increases. In order to make our model applicable and practical, we extensively tested our model on various orientations and

bandwidths. In Fig. 6a, we decrease the number of bandwidths by one so the total number of filters in our Gabor bank is decreased by 18 (e.g., 18 orientations in total). We also decrease the number of orientations by setting an interval from 10 to 25 as in Fig. 6b. One can see that when having more bandwidths and orientations, the metric, cTPR, increases but the processing time also increases. Since accuracy has the top priority in the problem, we keep all 90 filters in the experiments.

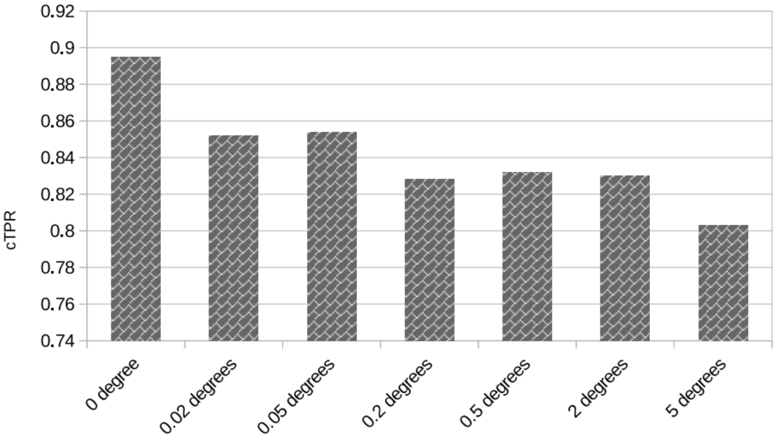


Fig. 7. cTPR impact of rotated fabric images at different angles. Note that the images are rotated in counter-clock wise.

4.6. Tests on fabric distortion (rotation)

Previous research tests proposed models only on fabric images without any distortion. However, in reality, fabrics are easily distorted while traveling from one process to another (In fact, distortion detection/correction machines are placed between some processes in factory). To simulate it, we test our model on manually rotated fabric images at different angles (e.g., 0.02, 0.05, 0.2, 2, and 5 degrees). Fig. 7 shows that the cTPR decreases as the rotation angle grows. The performance decreases since when classifying a rotated image patch, the classification boundary is not as clear as the one without rotations. Even with the decrease in performance, we think our model still holds its robustness very well as one can see from the visualized results of Fig. 5. We believe the robustness of our model not only credits to the Gabor filter bank but also the autoencoder. The Gabor filter bank provides magnitude responses from different orientations and then the autoencoder is able to learn more general features.

5. CONCLUSIONS

In this paper, we propose a one-class model with a carefully designed Gabor filter bank, an autoencoder as a general feature learner, and the nearest neighbor density estimator, to solve the fabrics defect detection problem. Our proposed model does not need defective samples, which is a significant practical advantage. The Gabor filter bank contains 90 Gabor filters at numerous scales and orientations, which saves our efforts on optimizing Gabor filters for any specific fabric, and also show the advantage when compared with a ResNet model that was trained on a limited dataset. In order to increase the applicability of our model, we design an autoencoder to better learn general features from the outputs of the Gabor filter bank. The experiments show the proposed autoencoder improves the performance by over 12.9% when compared against the hand-crafted and PCA feature selection methods. We also extensively test our model at different parameter settings. Moreover, we demonstrate applicability by showing our model can work well on plain, patterned, and rotated fabric images.

REFERENCES

[1] H.-w. Zhang, L.-j. Zhang, P.-f. Li, and D. Gu, "Yarndyed fabric defect detection with yolov2 based on deep convolution neural networks," in 2018 IEEE 7th data driven control and learning systems conference (DDCLS). IEEE, 2018, pp. 170–174.

[2] H. Zhou, B. Jang, Y. Chen, and D. Troendle, "Exploring faster rcnn for fabric defect detection," in 2020 Third International Conference on Artificial Intelligence for Industries (AI4I). IEEE, 2020, pp. 52–55.

[3] H. Y. Ngan, G. K. Pang, and N. H. Yung, "Automated fabric defect detection - a review," Image and vision computing, vol. 29, no. 7, pp. 442–458, 2011.

[4] L. Tong, W. K. Wong, and C. K. Kwong, "Differential evolution-based optimal gabor filter model for fabric inspection," Neurocomputing, vol. 173, pp. 1386–1401, 2016.

[5] K.-L. Mak, P. Peng, and K.-F. C. Yiu, "Fabric defect detection using multi-level tuned-matched gabor filters," Journal of Industrial & Management Optimization, vol. 8, no. 2, p. 325, 2012.

[6] S. S. Khan and M. G. Madden, "One-class classification: taxonomy of study and review of techniques," The Knowledge Engineering Review, vol. 29, no. 3, p. 345–374, 2014.

[7] Standard fabric defect glossary. [Online]. Available: https://www.cottoninc.com/quality-products/textileresources/fabric-defect-glossary/

[8] D. Chetverikov and A. Hanbury, "Finding defects in texture using regularity and local orientation," Pattern Recognition, vol. 35, no. 10, pp. 2165–2180, 2002.

[9] A. Latif-Amet, A. Ertuzun, and A. Ercil, "An efficient ¨ method for texture defect detection: sub-band domain co-occurrence matrices," Image and Vision computing, vol. 18, no. 6-7, pp. 543–553, 2000.

[10] D. Chetverikov, "Pattern regularity as a visual key," Image and Vision computing, vol. 18, no. 12, pp. 975– 985, 2000.

[11] A. Kumar and G. K. Pang, "Defect detection in textured materials using gabor filters," IEEE Transactions on industry applications, vol. 38, no. 2, pp. 425–440, 2002.

[12] L. Bissi, G. Baruffa, P. Placidi, E. Ricci, A. Scorzoni, and P. Valigi, "Automated defect detection in uniform and structured fabrics using gabor filters and pca," Journal of Visual Communication and Image Representation, vol. 24, no. 7, pp. 838–845, 2013.

[13] J. P. Jones and L. A. Palmer, "An evaluation of the two dimensional gabor filter model of simple receptive fields in cat striate cortex," Journal of neurophysiology, vol. 58, no. 6, pp. 1233–1258, 1987.

[14] Y. Li, H. Luo, M. Yu, G. Jiang, and H. Cong, "Fabric defect detection algorithm using rdpso-based optimal gabor filter," The Journal of The Textile Institute, vol. 110, no. 4, pp. 487–495, 2019.

[15] S. Mei, Y. Wang, and G. Wen, "Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model," Sensors, vol. 18, no. 4, p. 1064, 2018.

[16] L. Torrey and J. Shavlik, "Transfer learning," in Handbook of research on machine learning applications and trends: algorithms, methods, and techniques. IGI global, 2010, pp. 242–264.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, vol. 25, pp. 1097–1105, 2012.

[18] C. L. Beirao and M. A. Figueiredo, "Defect detection ˜ in textile images using gabor filters," in International Conference Image Analysis and Recognition. Springer, 2004, pp. 841–848.

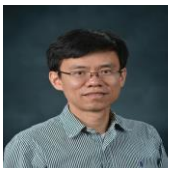
[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[20] Tilda textile texture database. [Online]. Available: https://lmb.informatik.unifreiburg.de/resources/datasets/tilda.en.htm

AUTHORS

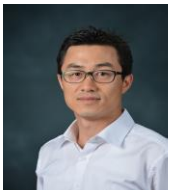
Hao Zhou: Mr. Zhou holds MS degree in Computer Science from University of Mississippi. His research area includes Machine Learning, Artificial Intelligence, GPU computing, and parallel computing. He received BS degree from the University of Mississippi and is currently a doctoral student at Pennsylvania State University.

Yixin Chen: Dr. Yixin Chen is currently a professor in Computer and Information Science department at the University of Mississippi. He received PhD degree in Computer Science from Pennsylvania State University. His research area includes Artificial Intelligence, Machine Learning, Computer Vision, and Image processing.



David Troendle: Dr. Troendle is currently a research professor in Computer and Information Science department at the University of Mississippi. He received PhD degree in Computer Science from the University of Mississippi. His research area includes GPU Computing, High Performance Parallel Computing, and Computer Architecture.

Byunghyun Jang: Dr. Jang is currently an associate professor in Computer and Information Science department at the University of Mississippi. He received Ph.D. degree in Computer Engineering from Northeastern University in 2010. His research area includes GPU Computing, High Performance Parallel Computing, and Computer Architecture.



GAN-BASED DATA AUGMENTATION AND ANONYMIZATION FOR MASK CLASSIFICATION

Mustafa Çelik[1, 2], Ahmet HaydarÖrnek[1, 3]

[1]Huawei Turkey R&D Center, Istanbul, Turkey
[2]Department of Computer Engineering Faculty of Computer and Informatics,
Istanbul Technical University, Istanbul, Turkey
[3]Department of Electrical and Electronics Engineering, Konya Technical
University, Konya, Turkey

ABSTRACT

Deep learning methods, especially convolutional neural networks (CNNs), have made a major contribution to computer vision. However, deep learning classifiers need large-scale annotated datasets to be trained without over-fitting. Also, in high-data diversity, trained models generalize better. However, collecting such a large-scale dataset remains challenging. Furthermore, it is invaluable for researchers to protect the subjects' confidentiality when using their personal data such as face images. In this paper, we propose a deep learning Generative Adversarial Networks (GANs) which generates synthetic samples for our mask classification model. Our contributions in this work are two-fold that the synthetics images provide. First, GANs' models can be used as an anonymization tool when the subjects' confidentiality is matters. Second, the generated masked/unmasked face images boost the performance of the mask classification model by using the synthetic images as a form of data augmentation. In our work, the classification accuracy using only traditional data augmentations is 93.71 %. By using both synthetic data and original data with traditional data augmentations the result is 95.50 %. It is shown that the GAN-generated synthetic data boosts the performance of deep learning classifiers.

Keywords

Convolutional Neural Network, Data Anonymization, Data Augmentation, Generative Adversarial Network, Mask classification

1. INTRODUCTION

In the last few years, wearing a mask had vital importance because of the pandemic that affects the whole people. It has become mandatory to wear a mask to avoid the disease. However, people are not sensitive to wearing a mask. For this reason, mask classifiers detect whether a person wears a mask is necessary for a public area to warn people and provide a healthy environment. Although the traditional machine learning methods are used in image classification work, it requires a huge amount of preprocessing and feature extraction steps. Over the last decade with the popularity of deep learning, image classification models have produced remarkable performance without spending effort on preprocessing and feature extraction.

One of the crucial issues in deep learning-based classifiers is the dataset size. Small-sized datasets may cause over-fitting and produce poor generalization on the test set. In many realistic cases like mask classification, we have limited datasets because obtaining labeled data is costly and time-consuming. Moreover, collecting personal data (e.g., face and face with mask) is more challenging due to subjects' confidentiality.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 191-199, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112315

To solve the overfitting problem, dropout [7], layer normalization [1], batch normalization [9] methods have been implemented. However, the models underperform in testing data because of the small-sized dataset. Researchers try to overcome this problem by using traditional data augmentation techniques. These techniques which include operations such as zooming, cropping, rotating, flipping, and scaling, are the standard methods that improve the performance of the classifiers (e.g., mask classifier). However, the images that are proliferated by using traditional augmentation techniques are fundamentally highly correlated and have a similar distribution to the original images.

In this work, we propose to use the generative adversarial network (GAN) [5] model to proliferate synthetic masked and unmasked face images which provide an additional form of data augmentation and an effective tool for data anonymization. The model consists of two networks (e.g., generator, discriminator), one network generates synthetic images and the other one discriminates between original and synthetic images.

The contributions of this work are the following:

- Generating synthetic masked and unmasked face images using GANs to boost the accuracy of the mask classification model.
- Anonymization of the real-world images to protect subjects' confidentiality.

2. RELATED WORK

Although there are plenty of studies [6, 8, 16, 18] that use deep learning-based approaches, these studies are based on clinical data and aim to diagnose the COVID-19 after the subjects have been already infected. The advantage of the deep learning approach can be used to develop a prevention system against the pandemic such as a model that detects whether people wear a mask or not.

Authors in [13] developed a combined deep learning and machine learning model which used deep learning to extract features and support vector machines to classify the samples whether wear a mask or not. In [19], researchers proposed a deep learning-based face mask-wearing condition identification method that consists of pre-processing, face detection crop, super-resolution, and mask-wearing identification steps. On common camera devices, [11] proposed an edge computing-based mask detection model which provides a real-time performance.

GANs are a promising approach that syntheses images [5]. Over the last decades, GANs have gained an extreme reputation in computer vision. Different types of GANs have been proposed to generate quite realistic natural images [4, 10, 14, 17, 20, 21]. Also, GAN-based models have been used to generate synthetic samples, especially in medical imaging [2, 3]. To the best of our knowledge, there is no existing literature on GAN-based synthetic masked face images generation as a form of data augmentation and anonymization.

3. MATERIALS AND METHOD

3.1. Materials

The images used in this study were taken by Huawei M2150 Camera which can send images via FTP protocol. By creating a monitoring setup at the Huawei entrance a dataset including real-world images was obtained. To train and test our deep learning model, 18400 images and 1178 images were used, respectively. The summary of the material can be seen in Table 1.

Table 1. The Properties of The Camera and Dataset

Effective pixels	2560 (H) x 1920 (V)
CPU	Hi3516D
Effective pixels	2560 (H) x 1920 (V)
CPU	Hi3516D
Frame Rate	30 FPS
Computing Power	1 TOP
Video Encoding Format	H.265/H.264/MJPEG
Intelligent Analysis	Face and Person Detection
Dataset	19578 images

3.2. Generating Synthetic Images

The key point of training a CNN network (e.g., mask classification) is the size of the training dataset. A large-sized dataset boosts the accuracy of the model. To enlarge the dataset we augmented the dataset in two different approaches: 1) classic augmentation methods which include operations such as zooming, cropping, rotating images, etc. 2) generating synthetic masked and unmasked face images with the help of generative models which use the data samples.

3.2.1. Classic Data Augmentation

The CNN model which has hundreds of parameters need a large-sized dataset to be trained. When building models with such networks that have multiple layers and there is a limited number of data, it is possible to face an over-fitting problem. The basic approach to solve the over-fitting is the classic data augmentation techniques [12]. These techniques include image transformations such as zooming, cropping, rotating, translation, scaling, flipping, and so on.

3.2.2. Generative Adversarial Network for Image Synthesis

Recently, GANs are popular frameworks that generate synthetic samples. It aims to learn the distribution of a dataset (e.g., masked/unmasked face images) in order to generate new images based on the learned distribution.

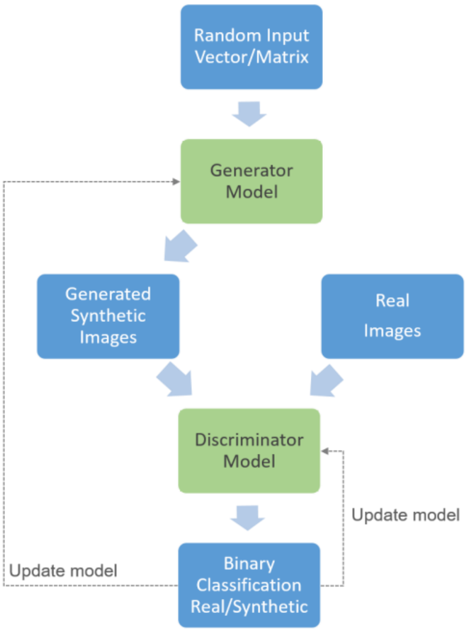

Random Input Vector/Matrix
Generator Model
Generated Synthetic Images
Real Images
Discriminator Model
Update model
Binary Classification Real/Synthetic
Update model

Figure 1. GAN Architecture Overview

While there exist many different types of approaches used in generative modeling, a GAN uses the approach shown in Fig. 1. There are two different neural networks called Generator (G) and Discriminator (D). The generator model generates synthetic samples (e.g., images) by using the given random input vector. The discriminator model tries to detect whether a given sample is real or synthetic. The training process follows each other, the discriminator model is trained a few epochs, then the generator is trained a few epochs, the process repeats till both the generator and discriminator model get better.

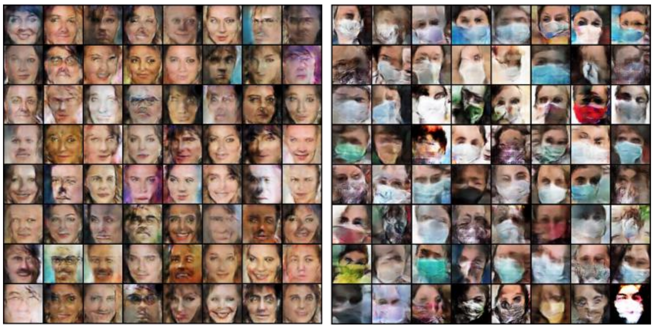


Figure 2. Masked and Unmasked Synthetic Images Generated by GAN Model

GANs are highly sensitive to hyperparameters, activation functions, and regularization. Table 2 shows the parameters of the models.

Table 2. Hyperparameters Used For Training The Model

Parameter Name	Value
Image size	64
Batch size	128
Learning rate	0.0001
Epochs	100
The activation function of discriminator's output-layer	sigmoid
The activation function of discriminator's middle-layers	Leaky ReLu
The activation function of the generator's output-layer	hyperbolic tangent
The activation function of the generator's middle-layer	ReLu

3.2.2.1. Generator Network

Generator network uses a random number vector or matrix as which is used as a seed to generate synthetic samples (e.g., image). It takes a 128x1x1 shaped tensor and converts it to a 3x64x64 images. In order to do the conversion, a deep convolutional GAN architecture is used (Fig. 3).

The middle layer of the architecture uses the ReLu Activation function [15]. In the output layer of the activation function, the hyperbolic tangent function (tanh) is used.

Generator Training Steps:

- The generator generates a batch of synthetic images.
- The synthetic images are given to the discriminator model.
- The discriminator model calculates the loss for the synthetic images.
- The weights of the generator model are adjusted with the help of the loss value which is calculated by the discriminator model.

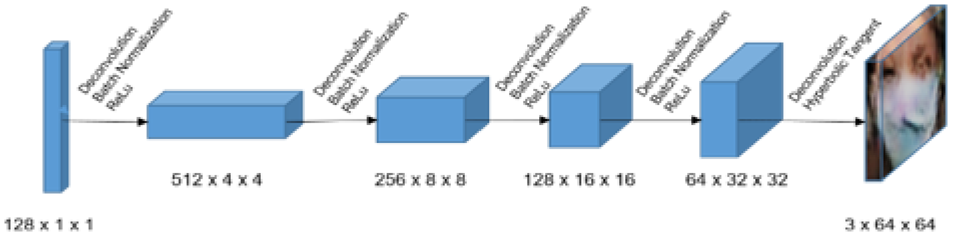


Figure 3. Generator Architecture

3.2.2.2. Discriminator Network

The Discriminator network uses CNN. It takes an image as input and classifies it as a real or synthetic image. As an input 3x64x64 image is given to the network. Discriminator gives an output of a single number between 0 and 1 which is a probability of the image being real. The architecture of the discriminator is shown in Fig. 4. After each layer Leaky ReLu activation is used, except the output layer uses the Sigmoid function. Batch normalization is applied after each middle layer conversion. Also, the discriminator model which is a binary classification model can use binary cross-entropy loss function for evaluation.

Discriminator Training Steps:

- It is expected that the discriminator model gives 0 if the given input image is generated by the generator model. If the output is 1, it means that the given image is from the real dataset which means the image is not generated by the generator model.
- A batch of real images is given to the discriminator model with the label of 1. Discriminator calculates the loss for real images.
- A batch of synthetic images is given to the discriminator model with the label of 0. Discriminator calculates the loss for synthetic images.
- The loss values of the synthetic and real images are added, and an overall loss value is calculated.
- The weight of the discriminator model is updated by using the overall loss of the whole input images.

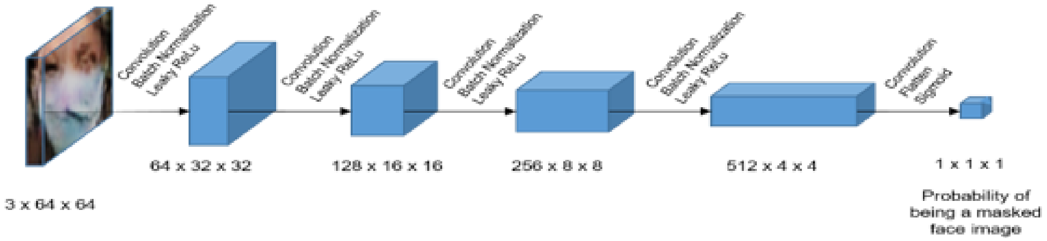


Figure 4. Discriminator Architecture

4. Experiments

To generate new face images with and without a mask, a GAN model was used and 1000 pieces images with mask 1000 pieces images without mask were generated. Two different scenarios were created to compare the effects of GAN-generated images on the training performance (i) training set was used to train the model and tested (Fig. 5) (ii) training set with GAN-generated images was used to train the model and tested (Fig. 6).

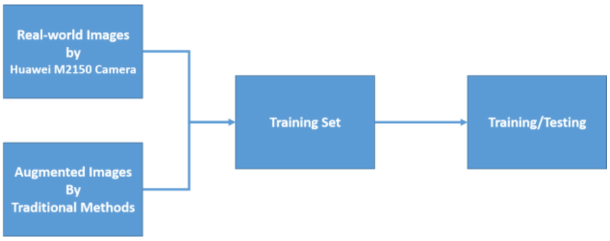


Figure 5. Scenario 1. The real-world images and images augmented by traditional methods are used to train the ResNet18 model.

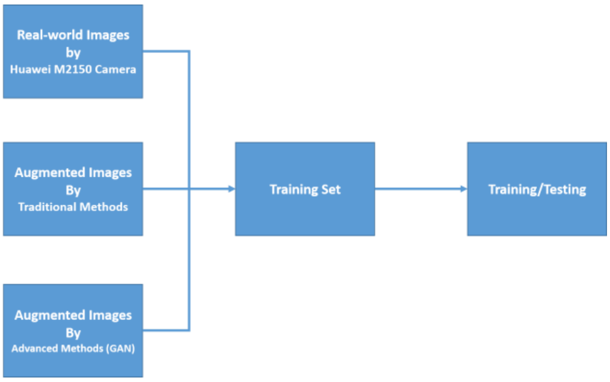


Figure 6. Scenario 2. The real-world images , images augmented by traditional methods, and images augmented by the GAN are used to train the ResNet18 model.

The ResNet18 pre-trained model was trained with a training set for the first scenario. For the second scenario, 1000 pieces images with mask 1000 pieces without masks were generated, and the ResNet18 pre-trained model was trained. The data information can be seen in Table 3}.

Table 3. Scenarios - Training Dataset

Training Dataset	**Scenario 1**	**Scenario 2**
With mask	9200	9200
Without mask	9200	9200
GAN-generated with mask	-	1000
GAN-generated without mask	-	1000
Total training data	18400	20400

5. RESULTS

After training parts were completed, trained 2 models were evaluated with the same testing dataset including 950 images with mask, and 228 images without a mask. The results can be seen in Table 4.

Table 4. All Results

All Results	**Scenario 1 (%)**	**Scenario 2 (%)**
Sensitivity	75.00	88.15
Specificity	98.21	97.28
Accuracy	93.71	95.50

According to the first scenario, 933 of 950 images with masks and 171 of 228 images without masks were correctly classified by the ResNet18 model. Therefore, the model achieved 98.21% specificity, 75% sensitivity, and 93.71% accuracy.

According to the second scenario, 924 of 950 images with masks and 201 of 228 images without masks were correctly classified by the ResNet18 model. Therefore, the model achieved 97.28% specificity, 88.15% sensitivity, and 95.50% accuracy.

6. DISCUSSION

Training a deep learning model without data augmentation generally causes overfitting because small datasets cannot be generalized by deep learning models. Data augmentation methods are categorized into traditional and advanced methods. Traditional data augmentation methods are already implemented by deep learning frameworks such as Tensorflow and Pytorch. Although traditional methods increase model performances advanced methods help models to achieve more performances.

In this study, we show how advanced methods increase the model performance by creating 2 different scenarios. Whereas, only traditional methods were used and achieved 93.71% accuracy in the first scenario, traditional and advanced (GAN-generated) methods were used together and achieved 95.50% accuracy in the second scenario.

It is shown that the GAN-generated synthetic data boosts the performance of the classifier. In future studies, we will be working on generating different images to train deep learning models.

7. CONCLUSION

In this work, we propose a GAN model to generate synthetic masked and unmasked images to increase classification performance and provide data anonymization.

By creating 2 different scenarios, how advanced methods increase the model performance was shown. While traditional methods achieved 93.71% accuracy, traditional and GAN methods together achieved 95.50% accuracy.

With the development of new GAN models, we will generate new images and extend our dataset to train our models without overfitting and provide more privacy.

ACKNOWLEDGMENT

This study was supported by the ”Epidemic Prevention System” project of Huawei Turkey R&D Center.

REFERENCES

[1] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

[2] A. Chartsias, T. Joyce, M. V. Giuffrida, and S. A. Tsaftaris. Multimodal mr synthesis via modality-invariant latent representation. IEEE transactions on medical imaging, 37(3):803–814, 2017.

[3] P. Costa, A. Galdran, M. I. Meyer, M. Niemeijer, M. Abra`moff, A. M. Mendonc¸a, and A. Campilho. End-to-end adversarial retinal image synthesis. IEEE transactions on medical imaging, 37(3):781–791, 2017.

[4] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. arXiv preprint arXiv:1506.05751, 2015.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.

[6] E. E.-D. Hemdan, M. A. Shouman, and M. E. Karar. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. arXiv preprint arXiv:2003.11055, 2020.

[7] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.

[8] L. Huang, R. Han, T. Ai, P. Yu, H. Kang, Q. Tao, and L. Xia. Serial quantitative chest ct assessment of covid-19: a deep learning approach. Radiology: Cardiothoracic Imaging, 2(2):e200075, 2020.

[9] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR, 2015.

[10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1125–1134, 2017.

[11] X. Kong, K. Wang, S. Wang, X. Wang, X. Jiang, Y. Guo, G. Shen, X. Chen, and Q. Ni. Real-time mask identification for covid-19: an edge computing-based deep learning framework. IEEE Internet of Things Journal, 2021.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25:1097–1105, 2012.

[13] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the covid-19 pandemic. Measurement, 167:108288, 2021.

[14] M. Mirza and S. Osindero. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.

[15] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In Icml, 2010.

[16] Q. Ni, Z. Y. Sun, L. Qi, W. Chen, Y. Yang, L. Wang, X. Zhang, L. Yang, Y. Fang, Z. Xing, et al. A deep learning approach to characterize 2019 coronavirus disease (covid-19) pneumonia in chest ct images. European radiology, 30(12):6517–6527, 2020.

[17] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans. In International conference on machine learning, pages 2642–2651. PMLR, 2017.

[18] Y. Oh, S. Park, and J. C. Ye. Deep learning covid-19 features on cxr using limited training data sets. IEEE transactions on medical imaging, 39(8):2688–2700, 2020.

[19] B. Qin and D. Li. Identifying facemask-wearing condition using image super-resolution with classification network to prevent covid-19. Sensors, 20(18):5236, 2020.

[20] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015.

[21] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image in painting with deep generative models. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5485–5493, 2017.

PROPOSAL FOR A METHODOLOGY TO CREATE AN ARTIFICIAL PANCREAS FOR THE CONTROL OF TYPE 1 DIABETES USING MACHINE LEARNING AND AUTOMATED INSULIN DOSING SYSTEMS (AID)

Rodrigo Vilanova[1] and Anderson Jefferson Cerqueira[2]

[1]Avantsec, Brasilia, Brazil
[2]Computer Science Department, University of Brasília (UnB), Brasilia, Brazil

ABSTRACT

The number of children, adolescents, and adults living with diabetes increases annually due to the lack of physical activity, poor diet habits, stress, and genetic factors, and there are greater numbers in low-income countries. Therefore, the aim of this article is to present a proposal for a methodology for developing a pancreas using artificial intelligence to control the required doses of insulin for a patient with type 1 diabetes (T1D), according to data received from monitoring sensors. The information collected can be used by physicians to make medication changes and improve patients' glucose control using insulin pumps for optimum performance. Therefore, using the model proposed in this work, the patient is offered a gain in glucose control and, therefore, an improvement in quality of life, as well as in the costs related to hospitalization.

KEYWORDS

Machine Learning, Artificial Intelligence, Data Mining, Diabetes, T1D, AID, iCGM, Clustering, Regression, Decision-making, Time series, Cybersecurity, Methodology.

1. INTRODUCTION

The number of diabetic people in the world has increased over the years due to lack of physical exercise, poor diet habits, stress, and genetic factors. It is estimated that the population of patients with diabetes will double in 25 years. The expected number of patients by 2034 will be approximately 44 million individuals, and the costs related to treating the complications of the disease are US$700 billion yearly. Thus, this paper seeks to bring intelligence to improve blood glucose control, using machine learning, reducing the ills of the disease and the costs related to disease treatments [24] [65].

To control diabetics, studies have been developed so that it is possible to administer medications with greater precision to patients. However, there is an excess of data being generated by multiple devices, based on the two main categories:

Blood glucose sensors - integrated continuous glucose monitoring (iCGM), such as the Freestyle Libre [8] [59] [14] and Dexcom [39].

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 201-215, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112316

Insulin pumps - automated insulin dosing systems (AID), such as the Tandem Diabetes Care t:Slim X2 insulin pump and the Ominipod.

According to Gordon Moore's observation [40], soon, several more wearables and IoTs, with even more technological capabilities should be available, further increasing the availability of data related to glucose control in patients with diabetes.

As the data volume tends to increase, it is inferred that an interoperable [34] system for [53] integration, indexing and data analysis, can benefit patients and healthcare systems through the use of a decision-making engine [37], using machine learning techniques.

This scenario may reduce the time to market [10] of other players in the glucose control scenario, increasing competitiveness and benefiting scientific production, in addition to reducing the costs of treating patients with acute diabetes.

Obviously, the provision of an artificial intelligence service to support the cloud decision-making algorithm [29], increases the attack surface for potential offenders [38]. Therefore, it is essential that advanced systems for cyberattacks early detection [41] are used to monitoring the intelligence service engine.

The Federal District Government, capital of Brazil, has a free iCGM program, via the Brazil's Unified Health System (SUS) [62], providing measurement sensors for patients with HbA1c up to 7 [64]. However, to get access to the sensors, the patient must fill out a document and then deliver it to a health centre in their region. Unfortunately, Brazil still has a great deficiency in the digital [36] and social [52] inclusion in a large part of the population. This fact makes it difficult to complete the aforementioned document, which can cause problems in accessing the blood glucose measuring device.

In this research, we used a project called AvantData for Good - AVD4G, by the Brazilian company Avantsec [5], where the goal is to collect data on diabetes control from FreeStyle Libre users [55], and the automatic completion of the report to ease the availability of the device to the disadvantaged population. The project also has, as a secondary goal, the storage, and processing of blood glucose data to support physicians and patients in the decision-making process of blood glucose control, respecting the General Data Protection Regulation [22] and Exchange of Supplementary Health Information (TISS) [20].

With the use and dissemination of AvantData for Good project [3] through the AvantAPI [4], it is expected to obtain a significant volume of data from diabetes patients, for clustering models algorithm [2], in order to determine possible groups of individuals and their relationships with factors influencing as: age, gender, race, volume of physical activity and eating habits of the population with diabetes.

2. BACKGROUND AND RELATED WORKS

According to the International Diabetes Federation (IDF) [26], the number of children and teenagers living with diabetes increases annually. In 2019, more than one million children and adolescents had type 1 diabetes (T1D) and 86,000 children (increasing by 2 and 3% per year) are expected to develop T1D each year. There is currently no cure for T1D, but people with T1D with adequate daily insulin treatment, regular blood glucose monitoring, education and support can live healthy lives and delay or prevent many of the complications related to diabetes, therefore it is mandatory to keep glucose within the target range in patients with T1D. Although pharmacotherapeutic advances have improved in recent years, many patients are still not adept at keeping their blood glucose levels within suggested limits. Blood Glucose Self-Monitoring (SMBG) helps improve blood glucose control and empowerment of people with diabetes; mainly useful for people with diabetes who are using insulin as it facilitates insulin delivery and detection of hypoglycemia.

In September 2016, the United States Food and Drug Administration (FDA) approved the Freestyle Libre Pro Glucose Monitoring System as a continuous glucose monitoring system (iCGM) for blind study in research. In September 2017 [19], the FDA approved the Freestyle Libre for personal use by patients, in which the sensor is disposable and applied to the back of the patient's arm, allowing its use for up to 14 days. Using this technology, physicians and patients can use a handheld device to download the blood glucose information stored in the sensor [8].

The "FreeStyle Libre" continuous glucose monitoring system [32] consists of a sensor and a handheld reader. A sensor unit is 3.5 cm in diameter and 0.5 cm thick. A thin filament is located in the centre underneath, which is coated with an adhesive. A handheld reader measures at a distance of 1 to 4 cm from the sensor. In addition to the glucose level, the device's touch-sensitive display indicates data curves as well as the trend of glucose values in arrow form. The sensor saves up to 8 hours of results. The handheld reader can be connected to a computer to create additional diagrams and analysis. A sensor measurement range from 2.2–27.8 mmol up to l (40–500 mg/dl). According to the manufacturer's specifications for use in humans, the sensor's lifetime is up to 14 days, before needing to be replaced [14].

Abbot [1] is the US company responsible for FreeStyle Libre product. The manufacturer makes a website available to its users [33] with the ability to collect and process information from (iCGM). Furthermore, the sensor works as an Analog Digital Converter (ADC)[47], therefore it allows storing and exported data.

To drug administration, it is possible to use insulin pump therapy, also known as automated insulin dosing systems (AID), which is an evolution application form, that has been shown to be highly effective in maintaining blood glucose levels and providing flexibility to patients' lives. It works by providing the patient with a continuous subcutaneous infusion of fast-acting insulin and allows the patient to administer bolus throughout the day, for feeding and correction of elevated glucose levels.

(AID) use is approved in patients with type 1 and selected patients with type 2 diabetes; however, it is important to select the correct patients for pump therapy. Insulin pump technology continues to evolve rapidly, and many options are now on the market, including those that are used in conjunction with continuous glucose monitoring (iCGM). However, (AID) is not suitable for all patients with diabetes who require the use of insulin [57].

Generally, AID is a treatment option for adults with type 1 diabetes who are motivated to improve glycemic control after a trial of multiple daily insulin injection (MDI) therapy and who can show the level of self-care required for adherence [23].

Artificial intelligence in medical applications using machine learning directly to the patient has been increasingly used to perform medical evaluation diagnoses [6]. Machine learning can make use of previously processed data to make objective and informed recommendations and can help ensure that decisions are made as assertively as possible, assisting healthcare professionals in decision-making.

The Algorithmic decision-making systems (ADMS) represent a holistic and interdisciplinary phenomenon and have some distinct interpretations [35]. Computer scientists focus on technical aspects of algorithms, implicitly privilege commercial interests, and celebrate the power of the artifact, but there is some research to suggest that algorithmic systems create economic benefits that substantially exceed possible ethical concerns and implications [27, 48]. The other interpretation concerns sociologists and attorneys who are pessimistic and defend the prohibition of most ADMS because they are unfair, having unethical or inappropriate decisions can be triggered in society [54, 61].

Hence, with developing a pancreas that makes use of artificial intelligence, the development of an ADMS is intrinsic to decide from collected data, that is from the data collected from the iCGM, the decision-making system decision can trigger the AID to administer the amount of insulin needed on a historical basis.

3. Methodology for an Artificial Intelligence Pancreas

To aid in the process of collecting glucose level data from patients, FreeStyle Libre was created as a less invasive technique that avoids the need for the patient to take a glucose measurement with multiple daily bites. After analysing the data, administration of the required daily insulin can be carried out after analysis by a physician using the insulin pump.

However, according to the FDA report [18], the FreeStyle Libre device may incorrectly indicate hypoglycemia, as after clinical studies it was verified that the device indicated 40% of the time that the user's sensor glucose values were at or below 60 mg/dL, where the user's glucose values were actually in the range of 81 to 160 mg/dL. For this reason, it is recommended that FreeStyle Libre glucose monitoring be based on trends and patterns analysed over time, not isolated data.

The scenario proposed in this paper for the control of diabetes in T1D patients is to use data mining and machine learning techniques to collect data from patients and, through clustering, provide enough information for decision-making algorithms with the use of artificial intelligence provides physicians with accurate information on administering the optimal amount of insulin according to the patient's identified glucose level. To achieve this goal, we suggest using the Methodology for an Artificial Intelligence Pancreas (MAIP) presented in Figure 1.

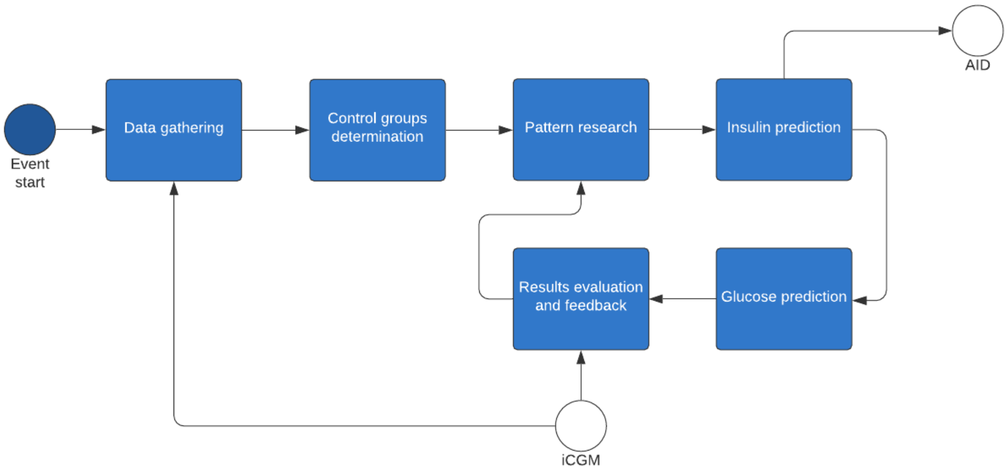


Figure 1. Methodology for an Artificial Intelligence Pancreas (MAIP)

The flow shown in Figure 1 is a continuous 6-step methodological process: Data collection, Determination of control groups, pattern search, insulin administration, blood glucose prediction, analysis of results and feedback.

3.1. Data gathering

Objective: Data receipt in AvantData for Good for dataset creation.

Processing: The website of Abbot, the company that created FreeStyle Libre, provides a converter for text spreadsheet, where all the readings sent can be consolidated for possible use by the user.

Expected outcome: It is suggested to export the data from the ADC sensor to a text file to load this information into AvantData for Good [4], which will be responsible for processing, filtering and indexing the data to create the dataset [30].

In the collection phase, there will be a short form that must be filled in by the patient with personal information. Data that allow the detection of whom the patient is will not be collected, that is, the objective is to have the minimum and necessary parameters for the creation of intelligence engines, but the individual's identity will be preserved and safeguarded.

3.2. Control groups determination

Objective: Detection of possible groups of patients with common characteristics (e.g., age, sex, cardiorespiratory capacity, physical activity, eating habits, among others).

Processing: Creation of a clustering model [63] in which the dataset must be prepared so that it can be processed. There are several pre-processing techniques available [25].

Dataset processing must be done using linear algebra [16], with mathematical operations between matrices and vectors. Thus, in order for the dataset to be properly processed, the categorical variables must be worked on to become their numerical equivalents [66].

After processing the dataset, it will be necessary to adjust the mathematical scales through feature scaling [51]. The use of min–max normalization is sufficient for this model, as the technique will allow the mitigation of possible mathematical dominance of a variable in relation to others in the model.

As we are dealing with an unsupervised model for this phase, it is not yet necessary to use the y vector. In this way, the dataset is ready for processing.

After pre-processing the data, the choice of classification models and the evaluation of the efficiency of each model begins, as well as the number of clusters that will be made available for the next phase. Each cluster will represent a group of individuals that will be studied in the control phase.

Due to the high accuracy demonstrated in other experiments [11], the K-Mean and SVM algorithms can be used to process the data in the clusters.

Once the model has been processed, the stage of adjusting the number of ideal clusters to define the control groups begins, as well as their related characteristics of associated factors determined in the "Data Collection" phase.

Expected outcome: Control groups correlated to factors associated with diabetes.

3.3. Pattern research

Objective: Establish a function of the relationship of the independent variables with the estimated probability for ideal insulin dosage [42].

Processing: Apply regression models to the control groups, using a physician-assisted insulin delivery process and with control of the glycemic response through the FreeStyle Libre of individuals belonging to the control groups [60, 9].

In this step, a mathematical function must be identified that will represent a method of estimating the volume of insulin for the patient. For this, the system will use non-linear regression algorithms, due to its satisfactory ability to reduce the risk of adverse situations [43].

Expected outcome: Analysis of data previously determined in the previous phase and creation of an insulin administration mechanism correlated with the individual's metabolic response and the expected blood glucose after a certain period of time. [21].

For each control group, it is suggested the administration of the medicine, via (AID), as prescribed by the physician and the sensing of the blood glucose level response via (iCGM). Thus, a new dataset will be created and can be represented by an X matrix, with the following independent variables: Control group, Amount of Insulin administered, Time of medicine administration, Current Insulin Level.

The y vector with the expected result of insulin in the next few minutes will serve as a basis for feedback to the system in later phases, as well as the measurement of performance in relation to the suggested dose [28].

Therefore, the objective of this step is to create models with high capacity to estimate the ideal insulin dose according to dataset data. This model will be evaluated and compared with the time series data presented in the next phases.

3.4. Insulin Prediction

Objective: Create a model that protects the patient against incorrect insulin administration. Obviously, insulin administration must be supervised by endocrinologists to ensure patient safety. At this stage, the objective is to determine a model to support decision-making, in order to protect the patient against any anomaly in the prediction stage.

Processing: With the result of the previous step, create a decision support algorithm [7] with the estimated amount of insulin for each individual [44], respecting their personal characteristics and receiving feedback in the closed-loop [56] of the FreeStyle Libre sensor (iCGM).

Based on the prediction of the amount of insulin administered via (AID) and the result obtained in several time intervals (iCGM), it is possible to obtain the creation of a new model where the blood glucose of a time interval (iCGM) will be analysed, related to the micro-dose of insulin (AID) applied in the immediately previous interval, that is, the blood glucose obtained (iCGM) after a certain period of time after the application of insulin via (AID).

With this information, the model must generate a classifier, which will be compared the estimated blood glucose, derived from the regression model, related to blood glucose achieved, data obtained via (iCGM). The model will sort in a boolean way whether the administration has reached an expected level or not.

Thus, the system begins to have the fundamental building blocks for decision-making on whether to continue with the administration of the next micro-dose of insulin via (AID).

Expected outcome: The system must have the decision-making capacity to apply or not the next insulin micro-dose, and each insulin administration must be reported to the system, as well as the decision taken at that moment. Thus, the intelligence engine has enough data to refine the models and necessary improvements in order to increase the accuracy and mitigate the risks of inappropriately injecting rapid insulin into the patient's body.

3.5. Glucose Prediction

Objective: The objective of this step is to create an estimate of blood glucose, distributed in time intervals in the future. In other words, predict how much blood glucose will be in the future, if this insulin is applied at the expected initial time.

Processing: Use a time series mechanism to predict the patient's blood glucose after insulin administration to provide feedback to the machine learning system and check whether the dose applied in the previous phase reached the expected result. Send feedback to the system with the expected blood glucose and the result achieved, measured via (iCGM).

With the proposed methodology, the effectiveness of the model must always be revised. To ensure that there is an efficient review process, another machine learning tool, based on time series, should be used to predict blood glucose in a short period of time [50].

Expected outcome: The time series will be based on the measured samples of the (iCGM), as well as the administration of rapid insulin (AID) and the date/time of release of the homonym. In other words, we are talking about a multivariate time series system for glucose forecasting [58]. This supports the main objective of the project to provide an intelligence engine that allows to simulate an artificial pancreas [12].

3.6. Results evaluation and Feedback

Objective: The objective is to compare what was estimated with the individual's response and refine the model, seeking greater assertiveness in relation to the current medicine administration process.

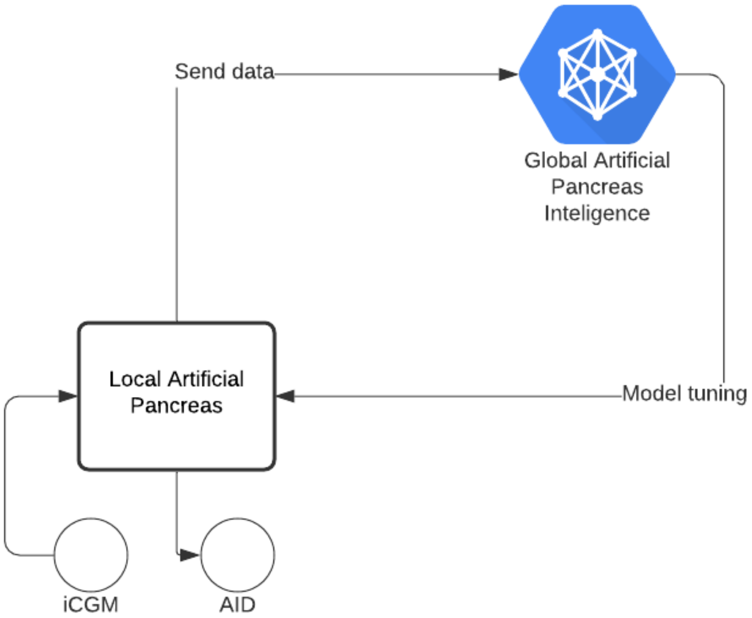

Send data
Global Artificial Pancreas Inteligence
Local Artificial Pancreas
Model tuning
iCGM
AID

Figure 2. Collaborative Artificial Pancreas Intelligence

Processing: This cycle must be executed indefinitely, always looking for an optimal relationship for each patient.

Model evaluation techniques must be used and compared with the measured results to adjust the parameters in each of the phases.

Obviously, we will need special attention to avoid overfitting or under fitting. In other words, addicted models are unproductive and will not lead to adequate therapy.

Expected outcome: At the end of each step, the results must be collected and sent to the data collection step or for debugging the models, according to each case. Maintaining historical data over a period of time of at least 5 years is highly recommended as that way we can continually review the analyzed data, its benefits to patients and assess the overall performance of the system over time.

4. RESULTS

We conducted research to create a dataset, with more than 48 thousand blood glucose readings, as well as their statistical data were computed, via AvantAPI, for the first phase of the methodology, as represented in the table below:

Due to the high cost of the (AID) we are still unable to proceed with the other phases of the project for this paper. As the focus of the study is the methodology for feeding the model in closed-loop, it is not possible to continue detailing the results without the data from the (AID).

Table 1. Blood glucose reading of a patient

Data	Max	Min	Mean	Standard Deviation
Historic Glucose mg/dL	379.00	40.00	132.29	47.84
Scan Glucose mg/dL	382.00	40.00	127.13	50.03
Rapid-Acting Insulin (units)	14.00	1.00	5.72	2.34
Long-Acting Insulin (units)	50.00	3.00	43.72	4.49
Total				48,642

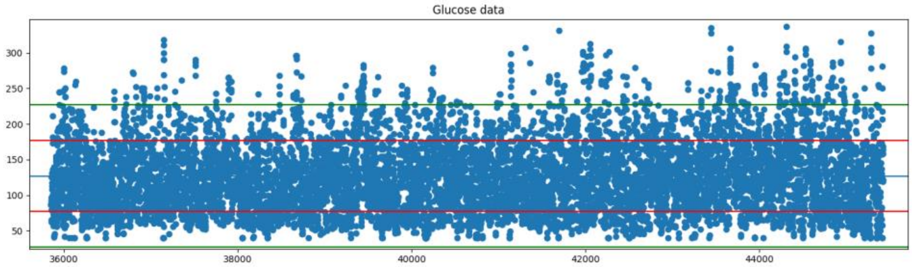


Figure 3. Glucose Sample Data

5. Discussion

The proposed MAIP methodology seeks to support the work to create a cybernetic pancreas. In the same way that we have pacemakers, in the near future, we may have digital pancreas to support T1D diabetes therapy.

The scope of this work is limited to supporting cyber intelligence for the best performance of insulin delivery to the individual. This intelligence would be a common asset, in an open source, interoperable platform, so that any company or individual can use the benefits of a worldwide collaborative system to simulate the processing of carbohydrates in the human body.

For this infrastructure to be available on a global scale, 5G technology brings tools that were previously impractical. However, in the area of technology, there are several cyber threats, for example: ransomware, denial of service attacks and malware in general.

For the success of this project, it is necessary that users participate as collaborators, and for this they must be aware that the data collected will be used exclusively for research and their names and personal information will not be stored. It is important to emphasize the aspects related to the protection of privacy and possible cyberattacks. Therefore, it is essential to mention that the General Data Protection Regulation of each country must be strictly respected for the use of their information.

Regarding information security, the complete anonymization [49] of the data is adequate, however, for the clustering algorithm to be useful in the future, it is necessary to collect additional data and habits to analyze the correlation with the levels of each person's glucose.

During the process of determining the control groups, attention must be paid to the categorization of variables, as it can generate a problem of mathematical preponderance between a class in relation to others, due to the reason that each class becomes a number, and numbers represent quantities that are mathematically comparable. But this is not the reality in the case of categories. For example: One race is not bigger or smaller than another, they are simply different. So, numbering the races would bring up the problem that one number is larger than the other, which is obviously not the case.

Thus, the ideal way would be to first transform each of the independent variables into a number and then each of the categories become other independent variables, represented in a boolean data. However, after this process, the dimensionality of the model will be increased.

Furthermore, it is necessary to monitor the dimensions of the model and adjust it, so that it is not too computationally burdensome, as the computational cost increases exponentially for each new dimension [15]. In theory, data sanitization should be restricted to disposing corrupted data. The data from the ADC are just about blood glucose and time, and this information will always be available. The other data related to associated factors comes from the user's registration, and the filling in of all information must be controlled in the data collection system.

Regarding the ”Search for patterns” phase, it is noted that the vector y should initially be related to the expected response time after the administration of rapid insulin, however, as we have a continuous monitoring system (iCGM), the possibility of creating several y vectors with a regression model for each reading after the application of rapid insulin, where each one is related to the probabilistic value of the expected blood glucose in relation to the time of medicine administration. This process occurs due to the large mass of data measuring the individual's metabolic response every 5 minutes. Even with the ability to measure every minute [45] the manufacturer suggests a reading at a minimum interval of 5 minutes. Thus, this was the time interval chosen in this work.

The advantage of this process is that insulin can be administered continuously in micro-doses [13] and with monitoring in a short period of time. In this way, insulin administration becomes safer, as the system is continuously fed back and the decision-making mechanism has the necessary elements for safe administration, as the closed-loop delivers the necessary feedback to increase the level of consciousness situational assessment of the individual's metabolic profile.

This would be an ideal mechanism to simulate the functioning of the pancreas, due to the continuous measurements and responses with the release of insulin into the bloodstream. Obviously, with the caveat that insulin administration in this case is not made directly into the bloodstream [31].

The use of multiple dimensions in time series is widely used in other forecasting domains, such as weather forecasting. The advantage of the proposed methodology is that the probability of not having difficulties related to missing values is very high. This stems from a refined data acquisition process in the “Data Collection” phase of the methodology.

Another fundamentally important factor in predicting glucose after insulin release is that we will have feedback material to support decision-making and to improve the regression model. We will be able to study the effect of each administration, comparing the expected glucose with the value reached in the therapy of everyone.

6. LIMITATIONS AND THREATS TO VALIDITY

Like any empirical study, this paper also has limitations and threats to its validity. As the study is a methodological proposal, a practical validation has not yet been carried out, collecting data from diabetic patients.

Due to the need to obtain numerous people during the process of implementing the proposed methodology, and to collect data from these patients to feed back to the system, it is necessary that patients are willing to provide the data, as the success of this project depends on primarily from the vast mass of data and assessments. In addition to glucose and insulin data, it is necessary to collect additional data such as: age, sex, volume of physical activity, if you are a user of other continuous-use medications, among other relevant data.

In order for this platform to be used on a global scale, an in-depth study of the best cyber defense practices is necessary in order to guarantee the availability, integrity and confidentiality of the collaborative effort. It would be useless for this intelligence to be available and the (AIDs) not to have adequate information on the measurement of insulin application.

In the same way, cyberterrorism attacks are another global threat and even attacks on patients' lives can occur, if security criteria are not very well worked out. This fact, in itself, obviously cannot be a demotivator for the research to continue. A joint effort with information security and cloud infrastructure companies is fully capable of enabling this technology to help improve the quality of life of T1D patients.

There are several possibilities to circumvent threats and make the system resilient and highly reliable. Such as blockchain to guarantee each of the transactions performed by the engine, as well as redundancies in (AID) and (iCGM). If we look more closely at Figure 3, we are talking about an artificial pancreas location, that is, even with the unavailability or integrity compromised, we would still have backups available with each of the users.

In summary, this material is not intended to exhaust the cybersecurity mechanisms to guarantee the system's functioning, but in-depth studies on the subject are of vital importance for the viability of the intelligence engine.

7. FUTURE WORKS

For the correct definition of the variables to be used in the dataset, a complementary study is necessary, where the associated factors in the control of diabetes [17] must be analyzed.
During data processing in clusters, the K-Mean and SVM algorithms can be used, however, the evaluation of the model requires further studies to ensure that the accuracy rate will be satisfactory.

The use of this methodology provides the structuring of a rich knowledge base, as future work can be carried out taking into account other associated factors, such as other syndromes or comorbidities. With a rich and cross-sectional basis for multivector analysis, we can think of asking new questions such as: What is the change in insulin therapy for patients affected with COVID-19? It is expected that with these data, this and several other responses will be answered more quickly, as we will have a historical memory of the individuals' metabolic response, as well as the effect caused by small nuances in the treatment.

The result of using the proposed methodology must be deeply validated with several endocrinologists to prevent the model from being wrongly trained and causing harm to the patient.

8. CONCLUSIONS

Studies indicate a significant increase in the quality of life of patients monitored via (iCGM), reaching a decrease in HbA1c of 0.3%. In addition to the 4-fold reduction in visits coded for diabetic ketoacidosis (DKA) in Emergency rooms. However, the control of insulin infusion via (AID) in relation to (MDI) does not represent a significant difference in hospitalization costs, as there is no gain in the use of (AID).

Thus, it is expected that with the model proposed in this work, assisted by artificial intelligence, for insulin administration, it will provide a gain in glycemic control and, consequently, an improvement in the patient's quality of life, as well as in the costs related to the hospitalization of patients from T1D [46].

REFERENCES

[1] Abbott: Abbott (November, 2021), https://www.abbott.com/, accessed in 11/26/2021

[2] Alexandre, K., Vallet, F., Peytremann-Bridevaux, I., Desrichard, O.: Identification of diabetes self-management profiles in adults: A cluster analysis using selected self-reported outcomes. Plos one 16(1), e0245721 (2021)

[3] Avantdata: Plataforma de análise, correlacionamento e gestão de dados em redes corporativas (December, 2018), https://www.avantdata.com.br, accessed in 11/25/2021

[4] Avantdata: Documentação das chamadas ao novo avantapi (December 2020), https://avantapi.avantsec.com.br/, accessed in 11/25/2021

[5] Avantsec: Avantsec - inovação em segurança da informação (December 2017), https://www.avantsec.com.br/, accessed in 11/25/2021

[6] Babic, B., Gerke, S., Evgeniou, T., Cohen, I.G.: Direct-to-consumer medical machine learning and artificial intelligence applications. Nature Machine Intelligence 3(4), 283–287 (2021)

[7] Bhargav, S., Kaushik, S., Dutt, V., et al.: A combination of decision trees with machine learning ensembles for blood glucose level predictions. In: Proceedings of International Conference on Data Science and Applications. pp. 533–548. Springer (2022)

[8] Blum, A.: Freestyle libre glucose monitoring system. Clinical Diabetes 36(2), 203–204 (2018)

[9] Borle, N.C., Ryan, E.A., Greiner, R.: The challenge of predicting blood glucose concentration changes in patients with type i diabetes. Health Informatics Journal 27(1), 1460458220977584 (2021)

[10] Charney, C.: Time to market: reducing product lead time. Society of Manufacturing Engineers (1991)

[11] Chauhan, T., Rawat, S., Malik, S., Singh, P.: Supervised and unsupervised machine learning based review on diabetes care. In: 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS). vol. 1, pp. 581–585. IEEE (2021)

[12] Clarke, W.L.: The artificial pancreas: how close are we to closing the loop. Pediatr Endocrinol Rev 4, 314–316 (2007)

[13] De Bock, M., Dart, J., Roy, A., Davey, R., Soon, W., Berthold, C., Retterath, A., Grosman, B., Kurtz, N., Davis, E., et al.: Exploration of the performance of a hybrid closed loop insulin delivery algorithm that includes insulin delivery limits designed to protect against hypoglycemia. Journal of diabetes science and technology 11(1), 68–73 (2017)

[14] Deiting, V., Mischke, R.: Use of the "freestyle libre" glucose monitoring system in diabetic cats. Research in veterinary science 135, 253–259 (2021)

[15] Donoho, D.L., et al.: High-dimensional data analysis: The curses and blessings of dimensionality. AMS math challenges lecture 1(2000), 32 (2000)

[16] Elgohary, A., Boehm, M., Haas, P.J., Reiss, F.R., Reinwald, B.: Compressed linear algebra for large-scale machine learning. Proceedings of the VLDB Endowment 9(12), 960–971 (2016)

[17] Faria, H.T.G., Rodrigues, F.F.L., Zanetti, M.L., Araújo, M.F.M.d., Damasceno,
M.M.C.: Fatores associados à adesão ao tratamento de pacientes com diabetes mellitus. Acta Paulista de Enfermagem 26, 231–237 (2013)

[18] Food, (FDA), D.A.: Summary of safety and effectiveness data (ssed) (September 2016), https://www.accessdata.fda.gov/cdrh docs/pdf15/p150021b.pdf, accessed in 11/26/2021

[19] Food, (FDA), D.A.: Fda approves first continuous glucose monitoring system for adults not requiring blood sample calibration (September 2017), https://www.fda.gov/news-events/press-announcements/fda-approves-first-continuous-glucose-monitoring-system-adults-not-requiring-blood-sample, accessed in 11/26/2021

[20] Galvão, M.C.B.: Classificações, terminologias e ontologias no campo da saúde. Asklepion: Informação em Saúde 1(2), 41–54 (2021)

[21] Hart, D.: Artificial intelligence & machine learning for effective management of blood glucose levels in patients with type 1 diabetes. Tech. rep., EasyChair (2021)

[22] Hawryliszyn, L.O., Coelho, N.G.S.C., Barja, P.R.: Lei geral de proteção de dados (lgpd): O desafio de sua implantação para a saúde. Revista Univap 27(54) (2021)

[23] Heinemann, L., Fleming, G.A., Petrie, J.R., Holl, R.W., Bergenstal, R.M., Peters,
A.L.: Insulin pump risks and benefits: a clinical appraisal of pump safety standards, adverse event reporting and research needs. a joint statement of the european association for the study of diabetes and the american diabetes association diabetes technology working group. Diabetologia 58(5), 862–870 (2015)

[24] Huang, E.S., Basu, A., O'grady, M., Capretta, J.C.: Projecting the future diabetes population size and related costs for the us. Diabetes care 32(12), 2225–2229 (2009)

[25] Huang, J., Li, Y.F., Xie, M.: An empirical analysis of data preprocessing for machine learning-based software cost estimation. Information and software Technology 67, 108–127 (2015)

[26] (IDF), I.D.F.: International diabetes federation (idf) (November 1950), https://idf.org/, accessed in 27/11/2021

[27] Jarrahi, M.H., Sutherland, W.: Algorithmic management and algorithmic compe-tencies: Understanding and appropriating algorithms in gig work. In: Taylor, N.G., Christian-Lamb, C., Martin, M.H., Nardi, B.A. (eds.) Information in Contempo-rary Society - 14th International Conference, iConference 2019, Washington, DC, USA, March 31 - April 3, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11420, pp. 578–589. Springer (2019). https://doi.org/10.1007/978-3-030-15742-5 55, https://doi.org/10.1007/978-3-030-15742-5 55

[28] Kølendorf, K., Bojsen, J., Deckert, T.: Clinical factors influencing the absorption of 125i-nph insulin in diabetic patients. Hormone and Metabolic Research 15(06), 274–278 (1983)

[29] Kuyoro, S., Ibikunle, F., Awodele, O.: Cloud computing security issues and chal-lenges. International Journal of Computer Networks (IJCN) 3(5), 247–255 (2011)

[30] Lau, A., Passerat-Palmbach, J.: Statistical privacy guarantees of machine learning preprocessing techniques. arXiv preprint arXiv:2109.02496 (2021)

[31] Lewis, D.M.: Do-it-yourself artificial pancreas system and the openaps movement. Endocrinology and Metabolism Clinics 49(1), 203–213 (2020)

[32] Libre, F.: Freestyle libre (December 2017), https://www.freestylelibre.de/, accessed in 11/26/2021

[33] Libre, F.: Freestyle libre (November 2021), https://www.libreview.com/, accessed in 11/26/2021

[34] Lopez, D.M., Blobel, B.G.: A development framework for semantically interoperable health information systems. International journal of medical informatics 78(2), 83–103 (2009)

[35] Marabelli, M., Newell, S., Handunge, V.: The lifecycle of algorithmic decision-making systems: Organizational choices and ethical challenges. The Journal of Strategic Information Systems 30(3), 101683 (2021)

[36] Mattos, F.A.M.d., Chagas, G.J.d.N.: Desafios para a inclusão digital no brasil. Perspectivas em Ciência da Informação 13, 67–94 (2008)

[37] Meyer, G., Adomavicius, G., Johnson, P.E., Elidrisi, M., Rush, W.A., Sperl-Hillen,
J.M., O'Connor, P.J.: A machine learning approach to improving dynamic decision making. Information Systems Research 25(2), 239–263 (2014)

[38] Milan, S., Hintz, A.: Networked collective action and the institutionalized policy debate: bringing cyberactivism to the policy arena? Policy & Internet 5(1), 7–26 (2013)

[39] Monitoring, D.C.G.: Dexcom continuous glucose monitoring (November 2021), https://www.dexcom.com/, accessed in 27/11/2021

[40] Moore, G.E., et al.: Progress in digital integrated electronics. In: Electron devices meeting. vol. 21, pp. 11–13. Washington, DC (1975)

[41] Narayanan, S., Ganesan, A., Joshi, K., Oates, T., Joshi, A., Finin, T.: Cognitive techniques for early detection of cybersecurity events. arXiv preprint arXiv:1808.00116 (2018)

[42] Nguyen, M., Jankovic, I., Kalesinskas, L., Baiocchi, M., Chen, J.H.: Machine learning for initial insulin estimation in hospitalized patients. Journal of the American Medical Informatics Association 28(10), 2212–2219 (2021)

[43] Noaro, G., Cappon, G., Sparacino, G., Del Favero, S., Facchinetti, A.: Nonlinear machine learning models for insulin bolus estimation in type 1 diabetes therapy. In: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). pp. 5502–5505. IEEE (2020)

[44] Obeidat, Y., Ammar, A.: A system for blood glucose monitoring and smart insulin prediction. IEEE Sensors Journal 21(12), 13895–13909 (2021)

[45] Olafsdottir, A.F., Attvall, S., Sandgren, U., Dahlqvist, S., Pivodic, A., Skrtic, S., Theodorsson, E., Lind, M.: A clinical trial of the accuracy and treatment experience of the flash glucose monitor freestyle libre in adults with type 1 diabetes. Diabetes technology & therapeutics 19(3), 164–172 (2017)

[46] Parkin, C.G., Graham, C., Smolskis, J.: Continuous glucose monitoring use in type 1 diabetes: longitudinal analysis demonstrates meaningful improvements in hba1c and reductions in health care utilization. Journal of diabetes science and technology 11(3), 522–528 (2017)

[47] De la Paz, E., Barfidokht, A., Rios, S., Brown, C., Chao, E., Wang, J.: Extended noninvasive glucose monitoring in the interstitial fluid using an epidermal biosensing patch. Analytical Chemistry 93(37), 12767–12775 (2021)

[48] Priami, C: Algorithmic systems biology. Commun. ACM 52(5) 80-88 (2009) https://doi.org/10.1145/1506409.1506427, https://doi.org/10.1145/1506409.1506427

[49] Ribeiro, S.L., Nakamura, E.T.: Privacy protection with pseudonymization and anonymization in a health iot system: results from ocariot. In: 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE). pp. 904–

908. IEEE (2019)

[50] Rodríguez-Rodríguez, I., Chatzigiannakis, I., Rodríguez, J.V., Maranghi, M., Gentili, M., Zamora-Izquierdo, M. Á.: Utility of big data in predicting short-term blood glucose levels in type 1 diabetes mellitus through machine learning techniques. Sensors 19(20), 4482 (2019)

[51] Rohini, M., Surendran, D.: Toward alzheimer's disease classification through machine learning. Soft Computing 25(4), 2589–2597 (2021)

[52] de Senne, F.J.N.: Desigualdades digitais e inclusão social: uma análise da trajetória do acesso e uso da internet em regiões metropolitanas brasileiras

[53] Shapiro, J.S., Mostashari, F., Hripcsak, G., Soulakis, N., Kuperman, G.: Using health information exchange to improve public health. American journal of public health 101(4), 616–623 (2011)

[54] Shin, D., Y.J.: Role of fairness, accountability and transparency algorithmic affordance. Computut. Hum, Behav. 98, (2019). https://doi.org/10.1016/j.chb.2019.04.019, https://doi.org/10.1016/j.chb.2019.04.019

[55] Silk, A.D.: Diabetes device interoperability for improved diabetes management. Journal of diabetes science and technology 10(1), 175–177 (2016)

[56] Song, S., Song, K., Xu, T., Zhou, W., Li, H., Liu, W.: The electronics design of real-time feedback control system in ktx. IEEE Transactions on Nuclear Science (2021)

[57] Sora, N.D., Shashpal, F., Bond, E.A., Jenkins, A.J.: Insulin pumps: review of technological advancement in diabetes management. The American journal of the medical sciences 358(5), 326–331 (2019)

[58] Tang, X., Yao, H., Sun, Y., Aggarwal, C., Mitra, P., Wang, S.: Joint modeling of local and global temporal dynamics for multivariate time series forecasting with missing values. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 5956–5963 (2020)

[59] Tsoukas, M., Rutkowski, J., El-Fathi, A., Yale, J.F., Bernier-Twardy, S., Bossy, A., Pytka, E., Legault, L., Haidar, A.: Accuracy of freestyle libre in adults with type 1 diabetes: the effect of sensor age. Diabetes technology & therapeutics 22(3), 203–207 (2020)

[60] Tucker, A.P., Erdman, A.G., Schreiner, P.J., Ma, S., Chow, L.S.: Examining sensor agreement in neural network blood glucose prediction. Journal of Diabetes Science and Technology p. 19322968211018246 (2021)

[61] Veale, M., Kleek, M.V., Binns, R.: Fairness and accountability design needs for al-gorithmic support in high-stakes public sector decision-making. In: Mandryk, R.L., Hancock, M., Perry, M., Cox, A.L. (eds.) Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018,

Montreal, QC, Canada, April 21-26, 2018. p. 440. ACM (2018). https://doi.org/10.1145/3173574.3174014, https://doi.org/10.1145/3173574.3174014

[62] Vencio, S., Caiado, A., Morgental, D., Bufaiçal, N., Carneiro, R., Vencio, R.C.: 21st brazilian diabetes society congress (2018)

[63] Vimal, V.: Prediction of diabetes mellitus using machine learning algorithms. In-ternational Journal of Advanced Engineering Science and Information Technology 4(4) (2021)

[64] Weykamp, C.: Hba1c: a review of analytical and clinical aspects. Annals of laboratory medicine 33(6), 393–400 (2013)

[65] Wu, X., Guo, X., Zhang, Z.: The efficacy of mobile phone apps for lifestyle modifica-tion in diabetes: systematic review and meta-analysis. JMIR mHealth and uHealth 7(1), e12297 (2019)

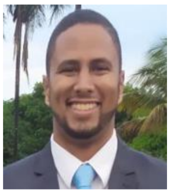
[66] Zelaya, C.V.G.: Towards explaining the effects of data preprocessing on machine learning. In: 2019 IEEE 35th international conference on data engineering (ICDE). pp. 2086–2090. IEEE (2019)

AUTHORS

Rodrigo Vilanova

Higher Education Graduation in Information Security Technology at ICESP on March 1, 2007.
Fluency in English. Native Portuguese. Knowledge of Spanish and Mandarin.
Creator of the AvantData product suite (https://www.avantdata.com.br).

Anderson Jefferson Cerqueira



I am a PhD student in Computer, I hold a Master's Degree in Application Computer from University of Brasília (UnB), Brazil in 2019. MBA in Software Engineering (2009) and graduated in Systems Information (2007). I'm Professor of Computer Science Course since 2009. My research interests include Software Engineering, Government Technology, Cybernetics and Health Informatics.

ANIME4YOU: AN INTELLIGENT ANALYTICAL FRAMEWORK FOR ANIME RECOMMENDATION AND PERSONALIZATION USING AI AND BIG DATA ANALYSIS

Kaiho Cheung[1], Ishmael Rico[2], Tao Li[3] and Yu Sun[4]

[1]Sentinel Secondary, 1250 Chartwell Dr, West Vancouver, BC V7S 2R2
[2]University of California, Berkeley, CA, 94709
[3]Purdue University, West Lafayette, IN 47907
[4]California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

In recent years the popularity of anime has steadily grown. Similar to other forms of media consumers often face a pressing issue: "What do I watch next?". In this study, we thoroughly examined the current method of solving this issue and determined that the learning curve to effectively utilize the current solution is too high. We developed a program to ensure easier answers to the issue. The program uses a Python-based machine learning algorithm from Scikit-Learn and data from My Animelist to create an accurate model that delivers what consumers want, good recommendations [9]. We also carried out different experiments with several iterations to study the difference in accuracy when applying different factors. Through these tests, we have successfully created a reliable Support vector machine model with 57% accuracy in recommending users what to watch.

KEYWORDS

Machine learning, anime, recommendations, Python.

1. INTRODUCTION

Anime commonly refers to Japanese animated cartoons and despite common belief it is aimed at both adults and children [7]. Over the last decade the popularity of anime has significantly increased especially in North America. We are starting to see subscription streaming services such as Crunchyroll pop up. For reference the anime industry as a whole is worth around 20 billion USD and still growing. However anime watchers suffer the same problem as other forms of entertainment watchers: the dilemma of what to watch next [12]. The simple solution is to go on the internet and ask; however like any other medium, taste is subjective. We developed a solution that recommends anime to you. Using AI we can provide quality and objective recommendations [10]. This solves the issues of scrolling past pages of forums of people arguing about who has the better taste. This solution is geared towards you and can be corrected to fit your taste perfectly. We are using data from MyAnimeList (MAL) as this is by far the largest wiki/rating site currently online. What you rate in the past is automatically in the programming, significantly reducing training time. By extension this program can also be attuned to manga and light novels, to give users the full experience of this culture [8]. With the growth of these systems more people are going to be watching anime and thus driving the industry forward and creating a positive feedback loop.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 217-225, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112317

For most people, the main solution to the dilemma of what to watch next is to go on the internet and search up said problem. Upon hitting enter the user is flooded with a wall of links to various internet forums and posts. Most noticeably Reddit, Reddit is one of the largest online communities and as such contains a lot of varying opinions. Take the subreddit r/AnimeSuggest, a forum dedicated to help others find anime. Most of the posts have replies with very different recommendations, actually, you even see people arguing over what they suggested. To newcomers of the anime community, this is incredibly overwhelming as they don't know who to listen to and have to base their decision on popularity or cross checking other posts. This process can take hours and can keep people away from watching anime. A similar recommendation program that implements AI is RikoNet, RikoNet uses an auto-encoder to cluster and filter data to make predictions [13]. This requires training time and user feedback and is not suitable for everyday users.

The method that we are using is building a machine learning model in Python that classifies anime similar to the user input. To do this we are using a non linear support vector machine (SVM) which classifies data points (vectors) using linear regression. This method allows us to have multiple variables of consideration which makes the predictions more accurate. SVMs are also very easy to build, come with built-in optimization, designed to work with unstructured data and scale well to high dimensional data. This makes SVMs perfect for recommending anime. We are using data from myanimelist.net which is the most well known and largest anime wiki/forum to ensure accuracy.

Experiments are performed to calculate and compare accuracy for determining the most appropriate machine learning model through the implementation of support vector learning (SVM) and regression models. In the experiment, we tested different models through adjusting the regression model, polynomial parameters, and inputted data sets.

We experimented with both the training data set size and the genre used for the SVM model to look at their effects on accuracy. In the first experiment we assessed the effects of changing the rows used for the train data [14]. For example instead of using row 0-100 as train data we would change it to row 200-300. This is to determine the optimal row for training data. The second experiment we modified the genre and examined what genres had more impact on the SVM. From this, we can determine what genre was more "important" and what can be removed. This experiment was mainly to increase performance as the more X variables we have, the more complicated the model is leading to longer compute time.

The way the paper will be structured after this introduction will be as follows: Section two is about challenges that the user face using other solutions currently available; Section three will discuss our proprietary solution and what methodology is associated with the solution; Section four details the experiments we've conducted and analyses on the data we gathered; Section five is dedicated to exploring related work in this area, as well their methodology when approaching the same problem. At the end section six is a conclusion and future development of this project.

2. CHALLENGES

Challenges for current systems have been identified as follows.

2.1. Finding out useful information

When answering the question of what to watch next most people turn towards the internet for help [15]. There are a lot of good informative recommendations out there on forums and

YouTube [11]. The problem is that there is too much information and too many factors at play for people to sort out what recommendation is good for them. The main factor that makes it impossible to decide who to listen to online is the arguments. Different users post different recommendations and often argue that their tastes are the best. A Lot of these posts are very long and consist of many back and forth requiring a lot of time to sort through. Sometimes people speek highly of an anime because the anime is infamously bad which can lead to even more confusion. Searching online can give you a general sense of what is preferable to watch but that takes time and previous experience.

2.2. Saving training time

Currently most of the anime recommended on anime sites are manually coded meaning that it is based on the tastes of the person who coded it or just based on popularity. Although there are ones that do incorporate machine learning, some examples we found include RikoNet(RKNE) which utilize "deep auto-encoders for the tasks of predicting ratings". However thesis types are limited by their models. The model RKNE has is very similar to what Netflix has, where it takes user data, makes a prediction and then uses user feedback to improve itself. In the long run this system is incredibly effective. The downside is that it takes time to train and you need constant user feedback. So it is hard to apply to small sites and gives sub-optimal recommendations in the early stages. Our program just gives a few anime based on statistics and has zero training time making it more viable to consumers.

2.3. Limited choices

When it comes to what to watch, premium streaming services like Crunchyroll and Funimation have very well developed "manual" recommendation systems. They are big enough to hire professional people to recommend what to watch next and for the most part, they are on point. However, anime licensing is incredibly expensive and can cost millions of dollars. For example Neon Genesis Evangelion, a very famous anime, will not be recommended on any of the aforementioned sites because Evangelion is licensed by Netflix. This makes Crunchyroll and Funimation recommendations undesirable as they can only recommend what they have. The alternative is to go to MAL -which has all the anime- but that leads to the first challenge: Over complicated reviews and user arguments. Our system has the best of both worlds where it is simple to navigate and has all the possible anime.

3. SOLUTION

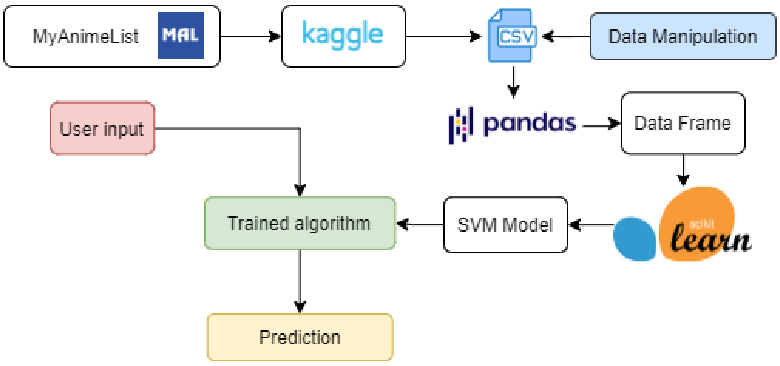
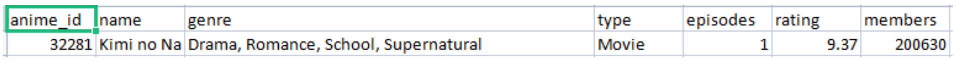


Figure 1. Overview of the project

To create this program we first had to collect data to train the machine learning algorithm. We decided to collect data from a site called My Anime List (MAL) because it is the largest databases and communities site for anime, manga and light novels. From this data, the machine learning algorithm will fit the data and can create a regression cure. The user will be asked to input whether they like a genre or not. From the user's answer, a point will be created on the regression curve and that will be used to predict what anime to recommend.

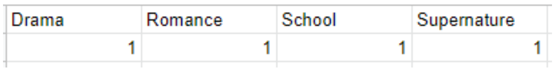
Originally we wanted to get the information directly off of MAL using their built-in API calls however that overreach the scale of this project. Instead, we got MAL data from Kaggle, a public database. We downloaded a CSV file that included anime_id, name, genre type, episodes, rating and number of members. However, this data was not usable as the genre types were not isolated in individual columns but instead a cluster of strings. Pandas the data analysis and sorter does not take strings so a work around was needed.



anime_id	name	genre	type	episodes	rating	members
32281	Kimi no Na	Drama, Romance, School, Supernatural	Movie	1	9.37	200630

Figure 2. CSV file

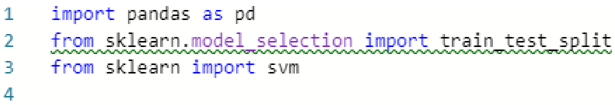
We had to reformat the data so that genre was isolated and had an integer value attached such as the picture below [6]. When we reformatting the data we had to create multiple columns so excel could check each time if it had the string it was looking for. For example it would check if the anime had a romance tag it would put a 1 and everything else would be 0. The data would look like the example below.



Drama	Romance	School	Supernature
1	1	1	1

Figure 3. Data example

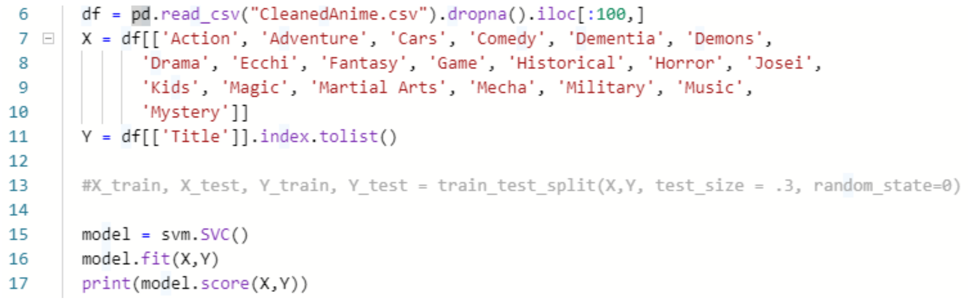
This way we can actually feed these numeric values into Pandas to create data frames. The data frames are then used to train our model. For this project we decided to use Scikit-learn which is a python based machine learning algorithm and more specifically using Support Vector Machines to do classification. Our X axis is the genre and Y is the titles. From there we get a set of user inputs which will define a specific parameter and using linear regression predict or in this context suggest an anime to the user.



```
1  import pandas as pd
2  from sklearn.model_selection import train_test_split
3  from sklearn import svm
4
```

Figure 4. Code line 1-3

Lines 1-3 Imports the necessary code libraries to function. pandas is the data sorter and sklearn is the SVM machine learning code.

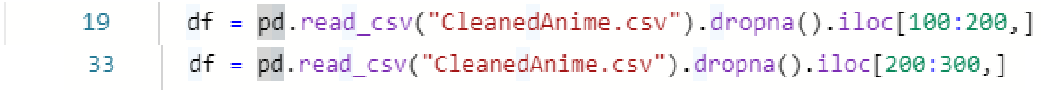


```
6   df = pd.read_csv("CleanedAnime.csv").dropna().iloc[:100,]
7   X = df[['Action', 'Adventure', 'Cars', 'Comedy', 'Dementia', 'Demons',
8           'Drama', 'Ecchi', 'Fantasy', 'Game', 'Historical', 'Horror', 'Josei',
9           'Kids', 'Magic', 'Martial Arts', 'Mecha', 'Military', 'Music',
10          'Mystery']]
11  Y = df[['Title']].index.tolist()
12
13  #X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = .3, random_state=0)
14
15  model = svm.SVC()
16  model.fit(X,Y)
17  print(model.score(X,Y))
```

Figure 5. Code line 6-17

Lines 6-11 is Setting up the training data location from pandas (pd) (CleanedAnime.csv) and the training set size. Secondly it defines what the X and Y variables are for the SVM. As seen above the X includes all the genres an anime can have and the Y is the title.

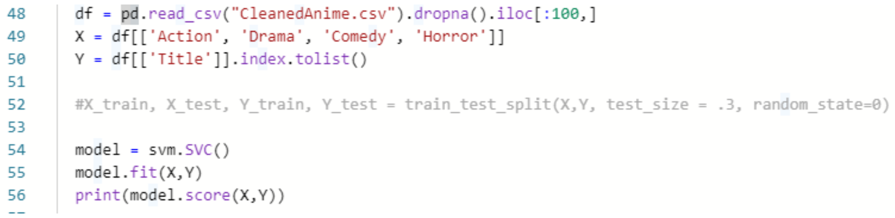
Lines 15-17 is creating a SVM, testing it on our data and then printing out the score.



```
19  df = pd.read_csv("CleanedAnime.csv").dropna().iloc[100:200,]
33  df = pd.read_csv("CleanedAnime.csv").dropna().iloc[200:300,]
```

Figure 6. Code line 19 and 33

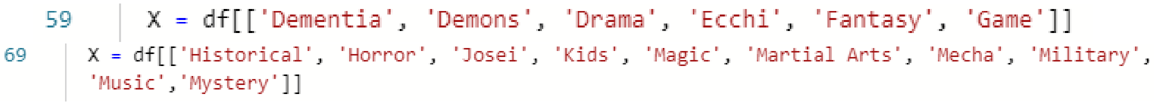
Lines 19-45 follow the same code showing lines 6-11 except for the training data set is changed to row 100 to 200 and row 200 to 300 in the CVS. show in Lines 19 and 33.



```
48  df = pd.read_csv("CleanedAnime.csv").dropna().iloc[:100,]
49  X = df[['Action', 'Drama', 'Comedy', 'Horror']]
50  Y = df[['Title']].index.tolist()
51
52  #X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = .3, random_state=0)
53
54  model = svm.SVC()
55  model.fit(X,Y)
56  print(model.score(X,Y))
```

Figure 7. Code line 48-56

Line 48-56 set the training set to the first 100 rows. However we changed the genres usd in the X axis. Then in line 54-56 we test the accuracy of the data.



```
59  X = df[['Dementia', 'Demons', 'Drama', 'Ecchi', 'Fantasy', 'Game']]
69  X = df[['Historical', 'Horror', 'Josei', 'Kids', 'Magic', 'Martial Arts', 'Mecha', 'Military',
    'Music','Mystery']]
```

Figure 8. Code line 59 and 69

Line 56-76 follow the same code showing lines 48-56 except for the different genres shown in Lines 59 and 69 aside from that the rest performs the same tasks.

4. EXPERIMENT

4.1. Experiment one

In experiment one as previously mentioned we wanted to determine the effect of specific rows used as training data and the accuracy of the SVM model associated with those rows. We performed a series of tests with and limited the test data to 100. The reason why we choose 100 rows as the size is because we are testing accuracy by checking how seminar our model's prediction is compared to the validation set(a set of data with known characteristics) however we do not have a validation set. We simply check the similarity between the predicted and the whole data set. This means that the larger the training set the more accurate it is going to be because the SVM would be based exactly on the test set [5]. This is undesirable because like a student studying with the actual test question the algorithm isn't developing anything, instead it would already know the answer when challenged with the real data. We found that 100 rows was not enough to trigger the aforementioned problem as our data set is about 11300 rows long, and that it yielded realistic results. Our data are sorted by popularity and score meaning that "better"/ more desirable anime was at the top and less known/ lower score ones were on the bottom. On to the test itself we wanted to see if using high ranking shows would affect accuracy so we set the training data rows to 0 to 100, then 100-200 and finally 200-300 and here are the results. Surprisingly when using the top 100 anime in training the accuracy is less compared to the training with less popular anime. Rows 100-200 and Rows 200-300 yielded the same results. The reason why this might happen is because popular anime often come with sequels effectively doubling the effect on the model thus making it worse facing the full data set. Further research is needed.

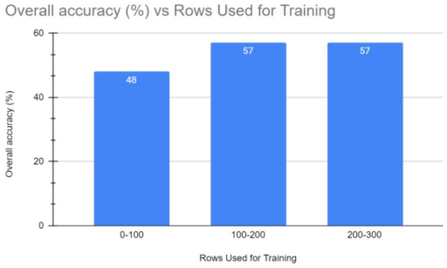


Figure 9. Overall accuracy VS rows used for training

4.2. Experiment 2

In experiment 2 as previously mentioned we wanted to determine the effect of specific genre on the accuracy. Initially we designed this experiment because we suspect that some genres are more of a key indicator than other genres. By knowing what is key we can reduce the number of x variables thus increasing performance subsequently seaming resources. We believe that common genres are better because they cover more of the data set while distinct genres will yield poorer results due not enough data. Our theory is based on the way SVM operates. More common genres would create support vectors that nudge the regression line in a more general direction thus covering more and leading to high accuracy. There is the chance that because of the commonality the hypnosis line covers what they should. On the other hand more niche genres would perform worse because of their lack of data points [4]. Also more niche anime would have more niche genres and as a support vector is based on niche anime it would not accurately predict the rest of the population. For the purpose of testing the genres we set the training data to the first 100 rows. The first iteration is what we thought is the most common genre which includes: Action, Drama, Comedy, Horror. The second iteration is a mix of both common : Drama, Ecchi, Fantasy and uncommon: Dementia, Demons, Game. The third iteration has all the uncommon tags: Historical, Horror, Josei, Kids, Magic, Martial Arts, Mecha, Military, Music, Mystery. The result was unexpected. If we look at the experiment 's first iteration which scored 48% in accuracy of courses by remaining genres we expected the accuracy to drop but not by this much. The accuracy of all three iterations did not reach half the accuracy of the one with all the genres. Another surprising result was that the one with all the uncommon tags scored the highest with the mix scoring the lowest. But based on the difference the one with all common tags performed decently. After this experiment we think the perforce increase in optimizing the genre is simply not due to the heavy accuracy penalties and our hypothesis is partly true.

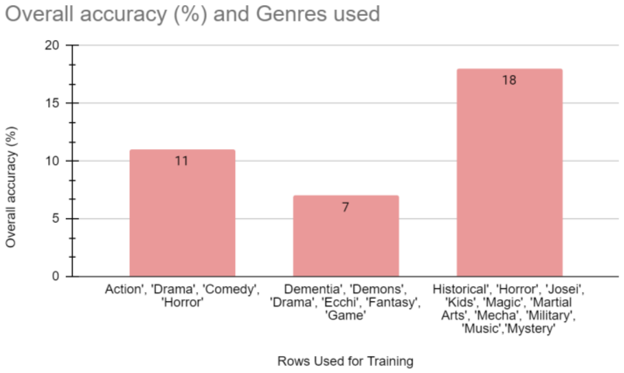


Figure 10. Overall accuracy and genres used

5. RELATED WORK

Mangaki is a person who recommends that quickly profiles users of what precedence they have. It uses a series of anime to identify what the user likes with buttons Like, Dislike, Neutral, Will see, Wont see. Then after the initial set of anime they would start recommending anime based on similarity. This is similar to our system to recommend based on similarity but it uses a filter with a formula instead of algorithm. This site is easy to use but it requires the user to have watched a few anime before and especially the initial set “classics” to actually work.

AniReco is a visual based anime person who recommends that finds other anime with similar tags. These tags include genre, voice actuator, director etc. they then connect anime with lines and the closer an anime is the more similarities it has. AniReco has a similar approach of finding similarity where we differ is the algorithm used. They use a vector space model while we used a support vector machine. Another difference is that AniReco requires the user to “train” the filter first by giving ratings of shows they watched. while ours can just be used base on preferred genres.

RikoNet is a person who recommends both a novel and Nike but for the purpose of this paper we will only look at the anime recommendation. The person who recommends let the user put in five anime they liked from there it creates a profile. This profile is then put through a hybrid filter and from the database it gives the user five recommended anime. Like Mangaki this requires the user to have watched previous shows or else it will just give out five most popular anime. One downside is that RikoNet is an engine so individual consumers have a hard time accessing it as it was designed for websites.

6. CONCLUSIONS

We created a machine learning algorithm to classify anime to ensure that algorithm is optimized and identify possible factors that can affect it [1]. We carried out two experiments. In the first experiment we decided to see if popularity within the training set affected overall accuracy. We found that using the top 100 anime acquired a lower accuracy and we suspect that is due to popular anime squalls and the nature of SVM [2]. In our second experiment we carried trials to determine whether common and uncommon genres had an effect on accuracy and the results showed that having uncommon genres will yield better accuracy. However it is best to use all the genres when modeling as the perforce gamin is not worth the 62.5% drop in accuracy. In conclusion, our SVM model with training data set rows from 100-200 and all the genres delivers excellent performance (52% accuracy) when classifying an anime. As such we believe this is a reliable anime recommended for consumers.

The current limitation of the program comes from the data set we are currently using. Due to the not using API calls, the current data we are using is about one years old which can cause issues regarding accuracy. In terms of accuracy the program scored very high however it was tested on the data it trained on. The validation data set was also derived from the main data, this can lead to questionable accuracy results if new data is introduced. The biggest limitation is taste, whether an anime is good or not is highly subjective, using data to predict something subjective is sub-optimal.

A stretch goal of this project would be MAL profile integration [3]. This means that we would collect data from the user's MAL profile and analyze what anime they have watched. Then train with that data to better suit them. Another thing to implement is a score and member count factors into the program where it would favour high ranking and more popular shows to improve end user experiences.

REFERENCES

[1] Anime and Manga Database and Community. MyAnimeList.net. (n.d.). Retrieved October 10, 2021, from https://myanimelist.net/.

[2] R/animesuggest. reddit. (n.d.). Retrieved October 10, 2021, from https://www.reddit.com/r/Animesuggest/.

[3] Hernández, Á. D. H. (2018, September). The anime industry, networks of participation, and environments for the management of content in Japan. In Arts (Vol. 7, No. 3, p. 42). Multidisciplinary Digital Publishing Institute.

[4] Ota, S., Kawata, H., Muta, M., Masuko, S., & Hoshino, J. I. (2017, September). AniReco: Japanese Anime Recommendation System. In International Conference on Entertainment Computing (pp. 400-403). Springer, Cham.

[5] Soni, B., Thakuria, D., Nath, N., Das, N., & Boro, B. (2021). RikoNet: A Novel Anime Recommendation Engine. arXiv preprint arXiv:2106.12970.

[6] Vie, J. J., Laıly, C., & Pichereau, S. (2015). Mangaki: an anime/manga recommender system with fast preference elicitation. Tech. Rep.

[7] Clements, Jonathan. Anime: A history. Bloomsbury Publishing, 2017.

[8] Brenner, Robin E. Understanding manga and anime. Greenwood Publishing Group, 2007. [9] Epstein, Joshua M. "Why model?." Journal of artificial societies and social simulation 11.4 (2008): 12. [10] Cames, Kathleen, and Administrative Analyst II-Grants. "Recommendation." City (2006).

[11] Shani, Guy, and Asela Gunawardana. "Evaluating recommendation systems." Recommender systems handbook. Springer, Boston, MA, 2011. 257-297.

[12] Dyer, Richard. Only entertainment. Routledge, 2005.

[13] Soni, Badal, et al. "RikoNet: A Novel Anime Recommendation Engine." arXiv preprint arXiv:2106.12970 (2021).

[14] Vanschoren, Joaquin, et al. "Experiment databases." Machine Learning 87.2 (2012): 127-158. [15] Gralla, Preston. How the Internet works. Que Publishing, 1998.

AUTHORS

Kaiho is a student currently attending Sentinel Secondary in West Vancouver. He is interested in pursuing computer science and engineering in the future.

SMART TAB PREDICTOR: A CHROME EXTENSION TO ASSIST BROWSER TASK MANAGEMENT USING MACHINE LEARNING AND DATA ANALYSIS

Brian Hu[1], Evan Gunnell[2], and Yu Sun[2]

[1]Arnold O. Beckman High School, Irvine, CA 92602
[2]California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

The outbreak of the Covid 19 pandemic has forced most schools and businesses to use digital learning and working. Many people have repetitive web browsing activities or encounter too many open tabs causing slowness in surfing the websites. This paper presents a tab predictor application, a Chrome browser extension that uses Machine Learning (ML) to predict the next URL to open based on the time and frequency of current and previous tabs. Nowadays, AI technology has expanded in people's daily lives like self-driving cars and assistive-type robots. The AI ML module in our application is more basic and is built using Python and Scikit-Learn (Sklearn) machine learning libraries. We use JavaScript and Chrome API to collect the browser tab data and store it in a Firebase Cloud Firestore. The ML module then loads data from the Firebase, trains datasets to adapt to a user's patterns, and predicts URLs to recommend opening new URLs. For Machine Learning, we compare three ML models and select the Random Forest Classifier. We also apply SMOTE (Synthetic Minority Oversampling Technique) to make the data-set more balanced, thus improving the prediction accuracy. Both manual tests and Cross Validation are performed to verify the predicted URLs. As a result, using the Smart Tab Predictor application will help students and business workers manage the web browser tabs more efficiently in their daily routine for online classes, online meetings, and other websites.

KEYWORDS

Machine Learning, Chrome extension, Task Management.

1. INTRODUCTION

The lockdown after the Covid-19 pandemic has caused almost all schools to switch to online classes such as Google classroom or Zoom, and businesses to work from home and use online meetings such as WebEx. Using Internet browsers constantly becomes a daily route for many people like high school or college students, individual workers, and company employees. Quite frequently, web browsing activities are repetitive. For example, a student needs to go to a classroom link for each period and click through multiple links to get assignments and class material. In this case, a student's class schedule is somewhat fixed during school hours, meaning that the same URL may be open at the corresponding bell time. Similarly, a businessman could have the same URLs for a daily status report or access repeated websites for work communication. Having a utility that can generate those repetitive URLs automatically will make it more convenient for people who use Internet browsers a lot daily.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 227-240, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112318

Moreover, a common issue when using a web browser is to have many tabs opened simultaneously, making it difficult to see the title of each website and quickly lose track of which tab is for what purpose. Also, many open tabs will consume more computer resources such as memory, leading to slower page rendering time and sometimes the crash of the browser. When this happens, a person usually feels frustrated and has to close the browser and start over. A survey with our friends and teachers indicates that they have similar issues. It is desirable to reduce the number of open tabs to reduce memory usage and not lose track of different tabs.

The above situations raise a question: is it possible to create an application that can automatically predict URLs and open a tab based on the history of web browsing activities? Browsers like Google Chrome have an Auto-fill feature, which automatically fills the address, user login, or other info on forms and websites when turned on. However, there seems to be no such predictive software akin to Auto-fill for web links. This has motivated us to develop the project presented in this paper, intending to help improve user productivity with fewer clicks and reduce stress. There are some key requirements to such an application. For instance, the prediction accuracy should be high enough for a user willing to use it; The application must be tailored for each user and segregate each user's data from each other; The algorithms should be able to handle randomized and possibly large data-set, etc.

Artificial Intelligence (AI) has gained more usage in real-world applications in recent years, such as robots that can help elders in their daily life, including opening doors, moving boxes and furniture, and performing health checks, etc. Using AI Machine Learning (ML) is a natural way to connect between a user's web browser usage pattern and a heuristic prediction of the next URL to open. Many studies used Machine Learning algorithms in recommendation systems [1]. Some researchers used content-based filtering [2], collaborative filtering [3], and hybrid or extended model [4][5][6] to provide personalized web recommendations. Most of these studies are suitable to be used in a back-end search engine rather than on the client for individual users. Several papers researched URL-based classification and prediction for categorizing web pages to predefined types of websites [7][8]. Some other papers built models based on browser history to either visualize the browser history [9][10], provide interactive prediction [11], or assist revisiting web pages [12][13]. Our goal is to have an auto suggestion of tab URL for front-end users.

In terms of ML algorithms, regression, classification, and clustering are three major types [14]. Regression is predicted based on a math function (Linear or Polynomial) and gives a result which is a scale of a specific type of data. Classification uses input data examples as a training dataset, finds matches, and outputs a prediction based on the existing labeled data. Clustering is an unsupervised classification to separate unlabeled data into several subcategories based on similarity. For the tab prediction in our study, the regression would not work well as the URLs are random, and there is no pattern to build an equation to predict a good result on the function graph. Clustering may not be the ideal model either because some URLs the user accesses could have the same root URL, and these URLs are likely grouped into the same cluster. Classification fits the problem best as the data-set is a collection of URLs, and each one has its class label. By inputting all website URLs a user visits into a database, the classification model can find the best map of the input data and predict where he will go next as every URL has its prediction. In this paper, we explore three ML classification techniques for tab prediction: SVM Support Vector Classifier (SVC), Passive Aggressive Classifier (PAC),and Random Forest Classifier (RFC) [15]. Through executing manual tests and Cross Validation for each Classifier, we demonstrate that the Random Forest Classifier is the winner based on both prediction accuracy and the ability of handling an unbalanced data-set.

In addition, we remove over-fitting by applying the SMOTE [16] oversampling method to maintain a better accuracy and customize each prediction to an individual user's web browsing history. Our approach is implemented using open source Scikit-Learn (Sklearn) [17] machine learning libraries. For user interaction we develop the application as a Google Chrome extension [18] that a user can download and install. The Chrome browser extension is a plug-in running on the Google Chrome browser for customized features. We leverage JavaScript, Chrome API, Python, Flask and Flask API [19] to build the connection and communication between various components like the chrome extension, back-end database, web server, and the ML module. We utilize Firebase Cloud Firestore [20] to store encrypted tab history data for each user to protect users' privacy. The ML module loads data from the Firebase, trains datasets, and predicts a URL, which will be returned to the user to open. Using the Tab Predictor Chrome extension application will help users save time, need less clicks thus more efficient, making it easier for people to manage their browser tabs. Further work includes the research and application of unsupervised machine learning algorithms, as well as modifying the application for scalability so that it may handle a larger number of users and higher volume of web history data.

The rest of the paper is organized as follows: Section 2 lists the challenges that we met during the design and development of the Tab Predictor Chrome extension. Section 3 describes the methodology, solutions, and details of software modules. Section 4 presents the test results and data analysis on the results. Section 5 summarizes related work. Section 6 provides the conclusion and future work of the project.

2. CHALLENGES

There are various challenges we encounter during the development of the application. This section describes three challenges, related to the machine learning algorithms, user data customization, and dealing with unbalanced data-set.

2.1. Selecting proper Machine Learning models to get a high accuracy

Machine Learning models only learn from what they know and are used purely to predict a result based on their testing data. Even with a lot of data, machine learning models may not reach a high level of accuracy. Depending on how the dataset is organized, certain ML algorithms are better suited than others. Because our application is directly interfacing with the user consistently, trying to ease their web lookup, we need a higher level of accuracy to maintain utility. Otherwise, the program itself seizes to be helpful and will ultimately not be used. Our application involves multiple elements to maintain a high accuracy: removing overfitting by using the SMOTE oversampling method, customizing each prediction to individual user's web history, and selecting a specific machine learning model. Due to the high amount of data gathered for each specific user, the best machine learning model was the Random Forest Classifier. It is easily able to handle all the different large data it is given and can make accurate predictions based on the time tabs were opened.

2.2. Tailoring the Machine Learning prediction for individual users

The Tab Predictor is a user interaction application where it requires specific data for each user. The machine learning algorithm needs to be able to pull data from a specific user's web history. Because each user is unique, prediction data must be tailored to them, or else the predictions would not be useful. In addition, since we are trying to predict every webpage they go to frequently, we are not only going to have unique datasets for individual users but also the relatively large data for each user. This requires us to set up a non-relational database where each user has their own subcategory of web history, allowing us to easily categorize them. In this paper we utilized Firebase to organize all users into a specific area, with each user having separate data collections. When a user runs a prediction, it pulls from their data collection specifically without interfering with other user's datasets.

2.3. Dealing with unbalanced datasets to improve prediction accuracy

The problem domain lends itself to an imbalanced data-set which is naturally weighted towards one specific result. Due to how frequently users browse the web, most of our data from users do not have an immediate follow up tab to be opened. This leads to most of the predicted results being wrong and having a false accuracy where many of the predicted results are blank. The data-set needs to be modified so that most of the predicted results are not biased towards a singular prediction. We used the Synthetic Minority Oversampling Technique, or "SMOTE", which increases the number of minority classes. By increasing the quantity of less common data elements, the accuracy of our model increased as the data-set became more balanced.

3. SOLUTION

Figure 1 shows the overview of the Tab Predictor application. It mainly contains four components:

- Chrome extension: the front-end interface with users, a plug-in running on the Chrome browser
- Firebase Cloud Firestore: the backend cloud database used for storing the Chrome tab events and web browsing history for individual users.
- Predictor engine: the main Machine Learning modules use the open source Scikit-Learn (Sklearn) Machine Learning libraries, running in a Python Repl.it.
- Flask web server: the bridge between the Predictor engine and the Chrome extension, which will fetch the predicted results from the Machine Learning modules and send back the prediction to the Chrome extension.

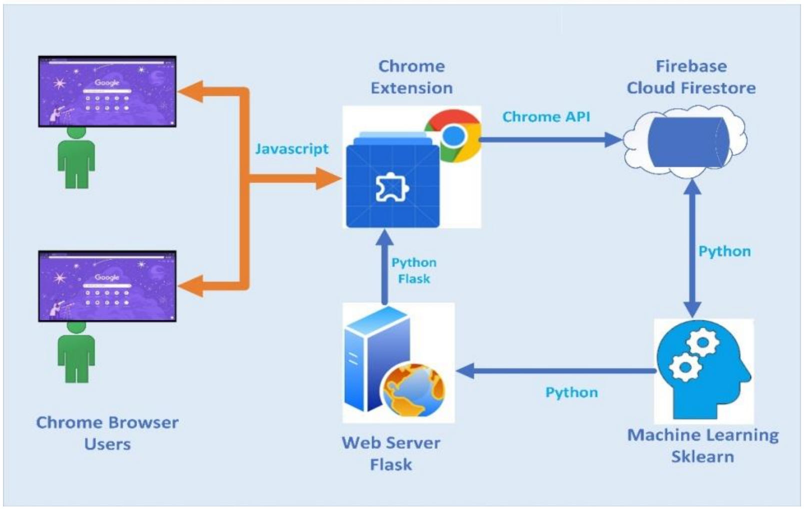
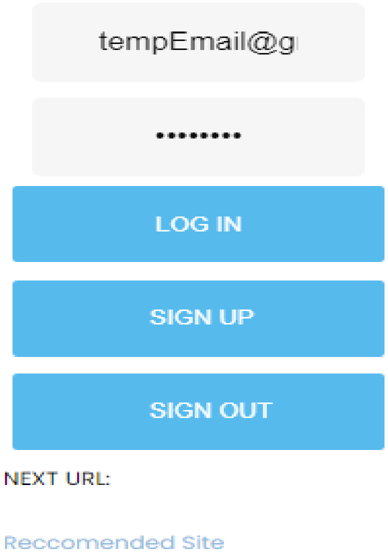


Figure 1. System Overview of the Tab Predictor

We use a code collection of JavaScript, Chrome API, Python, and Python Flask API to develop the Chrome extension and integrate various components. The entire program starts at the Chrome browser tab with a user opening a tab and typing in the search box or address bar. If the user signs in and runs the Chrome extension, the Chrome extension will add the websites or tab history the user accesses into the Firebase, then the Machine Learning module pulls the web history from the Firebase, iterates through all the websites, and runs its ML algorithms to make the prediction. The predicted URL will be returned to the extension through the Flask web server and prompt for the user to either open or ignore the recommendation. If the ML engine does not find a valid prediction for the extension, then it would return -1, telling the web server not to return any URL. The more accurate prediction should relieve users' searching and typing actions in the browser but not overwhelming them with URLs they don't want to open.

Extensions are software programs that enables users to customize the browsing experience. We use JavaScript and Chrome API to create the Chrome extension as the user interface of the Tab Predictor application. Figure 2 shows the Tab Predictor Chrome extension on the browser:



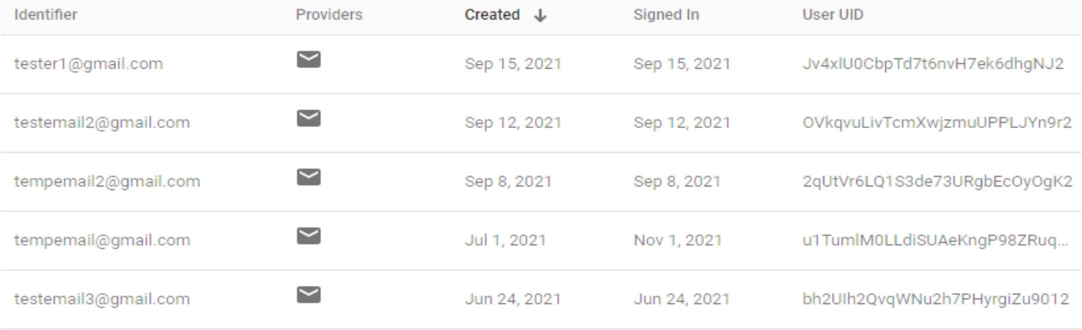
tempEmail@g

••••••••

Figure 2. Tab Predictor Chrome Extension User Interface

There are mainly three different areas in the Chrome extension: Back-end scripts, popup script, and content scripts.

The popup scripts handle the user login with a popup window asking for a user's email and password, see the screenshot shown in Figure 3 for stored user login info. The popup scripts are triggered by browser button click action. When the popup is created, it will send a handshake request to tell the background scripts that the popup window is live.



Identifier	Providers	Created ↓	Signed In	User UID
tester1@gmail.com	✉	Sep 15, 2021	Sep 15, 2021	Jv4xlU0CbpTd7t6nvH7ek6dhgNJ2
testemail2@gmail.com	✉	Sep 12, 2021	Sep 12, 2021	OVkqvuLivTcmXwjzmuUPPLJYn9r2
tempemail2@gmail.com	✉	Sep 8, 2021	Sep 8, 2021	2qUtVr6LQ1S3de73URgbEcOyOgK2
tempemail@gmail.com	✉	Jul 1, 2021	Nov 1, 2021	u1TumlM0LLdiSUAeKngP98ZRuq...
testemail3@gmail.com	✉	Jun 24, 2021	Jun 24, 2021	bh2Ulh2QvqWNu2h7PHyrgiZu9012

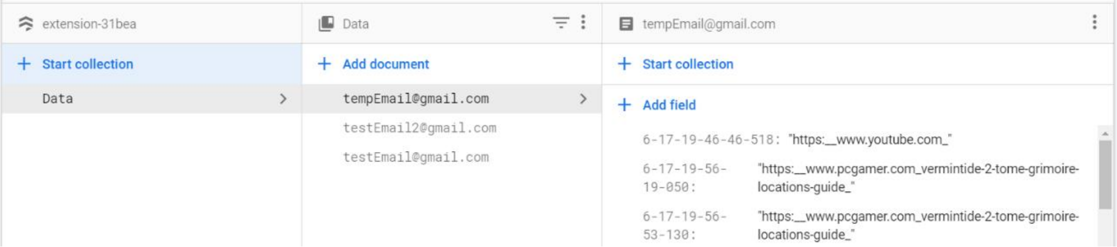
Figure 3. Tab Predictor Chrome Extension User Login Sample

The back-end scripts initialize the connection to the back-end Firebase Cloud Firestore, collect the tab information such as tab Id, tab URL, date, and time, combine the tab info and user login data, then store them into the Firebase. The history is organized in a way that the most recent tabs are always at the bottom, and we know the exact time it was opened. In addition, the back-end scripts listen for events and will send information back to the popup when getting the handshake from the popup window.

The content scripts run whenever a web page is loaded and are between the web page and back-end scripts. They communicate with the back-end scripts through messaging.

One of the paper's essential parts is to define where to save the browser events and history. There are many cloud databases available, such as Firebase Cloud Firestore, Parse Open Source back-end Platform, Back4app, Heroku, etc. [21]. Firebase is selected as it can hold a lot of data while being simple to implement into the project. It provides certain free features and has large data libraries and commands to pull data from or push data into it. Also, it was capable of handling user accounts, allowing for each user to have their own history saved separately.

We organize all users into a specific user data collection where each user has their own subcategory of data. Each user is identified and segregated by a login token, typically an email address and password, and the web history for a user is stored as a child collection of the root. This way individual users have their own section of history URLs. Figure 4 is a sample of the Firebase structure:



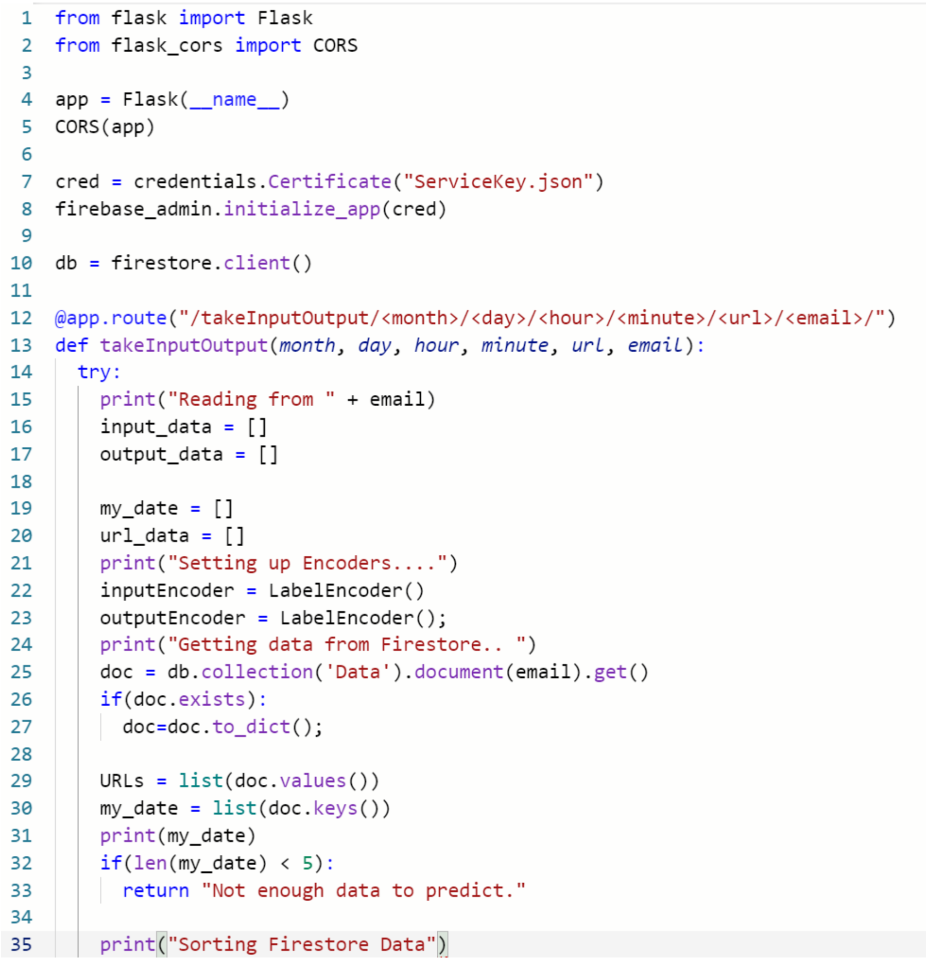
Data

tempEmail@gmail.com
testEmail2@gmail.com
testEmail@gmail.com

```
6-17-19-46-46-518: "https:__www.youtube.com_"
6-17-19-56-19-050: "https:__www.pcgamer.com_vermintide-2-tome-grimoire-locations-guide_"
6-17-19-56-53-130: "https:__www.pcgamer.com_vermintide-2-tome-grimoire-locations-guide_"
```

Figure 4. Firebase Cloud Firestore for Saving Tab History Info for Individual Users

Python code is used to access the Firebase and pull data from it. Figure 5 demonstrates the code snippet of connecting and loading data from Firebase: First the user logs in to the Firebase via stored credentials, next the machine learning module reads the data collection from the Firebase for a particular user identified by the user's email, then sorting the data, and at last encoding the dataset for the Machine Learning Classifier.



```
from flask import Flask
from flask_cors import CORS

app = Flask(__name__)
CORS(app)

cred = credentials.Certificate("ServiceKey.json")
firebase_admin.initialize_app(cred)

db = firestore.client()

@app.route("/takeInputOutput/<month>/<day>/<hour>/<minute>/<url>/<email>/")
def takeInputOutput(month, day, hour, minute, url, email):
  try:
    print("Reading from " + email)
    input_data = []
    output_data = []

    my_date = []
    url_data = []
    print("Setting up Encoders....")
    inputEncoder = LabelEncoder()
    outputEncoder = LabelEncoder();
    print("Getting data from Firestore.. ")
    doc = db.collection('Data').document(email).get()
    if(doc.exists):
      doc=doc.to_dict();

    URLs = list(doc.values())
    my_date = list(doc.keys())
    print(my_date)
    if(len(my_date) < 5):
      return "Not enough data to predict."

    print("Sorting Firestore Data")
```

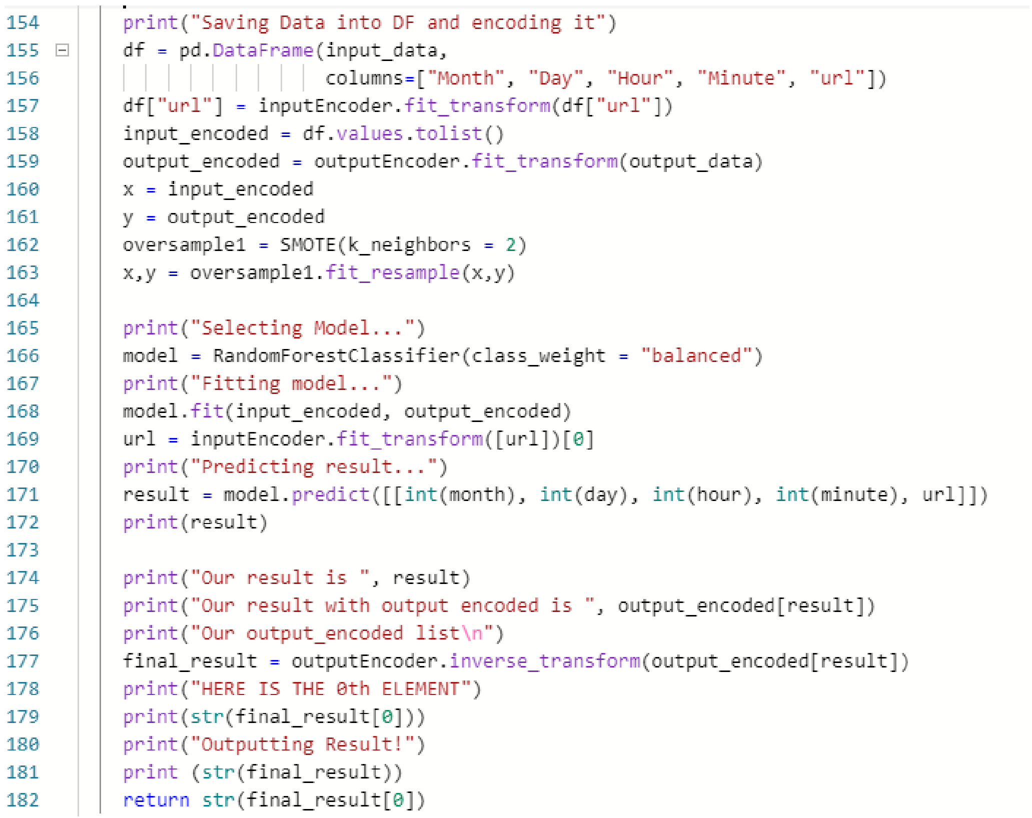
Figure 5. Code Sample for Accessing and Pulling Data from Firebase

The Machine Learning module is built using Python running on a boosted Repl.it platform. Repl.it is a browser-based online development environment for interactive programming.

There are many open-source Machine Learning (ML) libraries. Some popular ones are TensorFlow, Scikit-Learn, Theano, Caffe, Keras, and PyTorch [22]. We select Scikit-Learn(Sklearn) as it contains a robust and easy-to-use library capable of handling most predictions. Importing Sklearn in Python is simple, and there are no external programs that need to be downloaded. Sklearn also has a built-in cross-validation library, which we can use to test the accuracy of the Machine Learning algorithms.

Figure 6 shows the code section for the Machine Learning module. We encode the dataset from the Firebase to the format that the ML classifiers can read, call one of the ML classifiers from the Sklearn libraries to perform the prediction, then decode the predicted result to a human readable format.

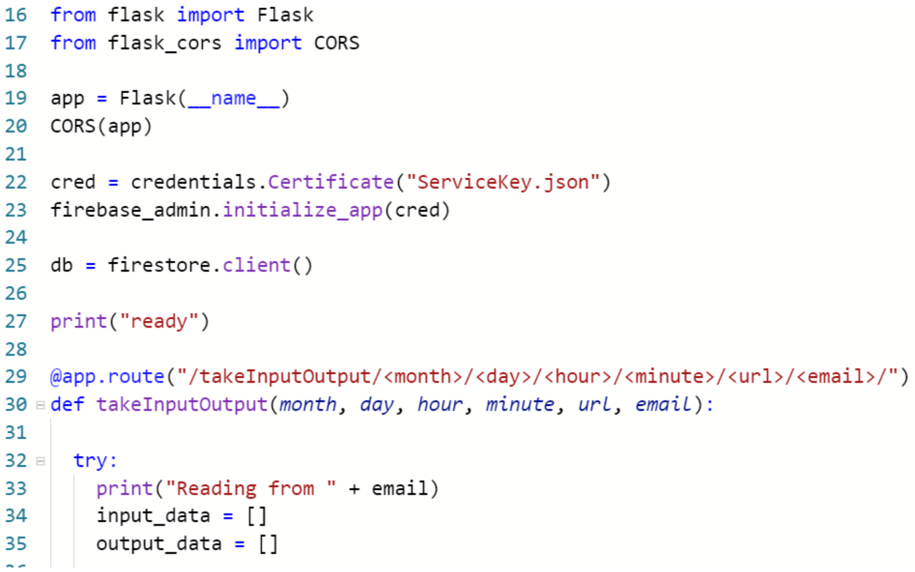
The python machine learning algorithm is refitted each time it is used. Since there are multiple users, each with their own unique search history, the machine learning algorithm must learn the patterns of each database every time a new user logs on and types in a new URL.



```
    print("Saving Data into DF and encoding it")
    df = pd.DataFrame(input_data,
                      columns=["Month", "Day", "Hour", "Minute", "url"])
    df["url"] = inputEncoder.fit_transform(df["url"])
    input_encoded = df.values.tolist()
    output_encoded = outputEncoder.fit_transform(output_data)
    x = input_encoded
    y = output_encoded
    oversample1 = SMOTE(k_neighbors = 2)
    x,y = oversample1.fit_resample(x,y)

    print("Selecting Model...")
    model = RandomForestClassifier(class_weight = "balanced")
    print("Fitting model...")
    model.fit(input_encoded, output_encoded)
    url = inputEncoder.fit_transform([url])[0]
    print("Predicting result...")
    result = model.predict([[int(month), int(day), int(hour), int(minute), url]])
    print(result)

    print("Our result is ", result)
    print("Our result with output encoded is ", output_encoded[result])
    print("Our output_encoded list\n")
    final_result = outputEncoder.inverse_transform(output_encoded[result])
    print("HERE IS THE 0th ELEMENT")
    print(str(final_result[0]))
    print("Outputting Result!")
    print (str(final_result))
    return str(final_result[0])
```

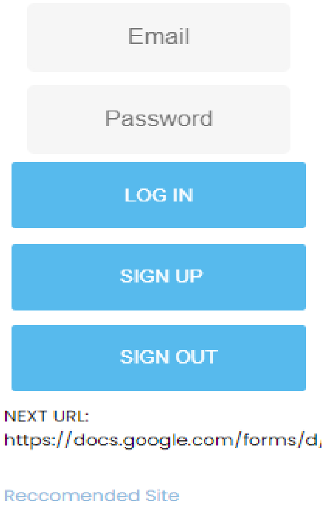
Figure 6. Code Sample for Prediction Using Machine Learning Algorithm

The Flask server is one of the most popular Python web application frameworks, easy to get started with, and has good readability [19]. We leveraged the Python Flask to fetch the prediction outputs from the Machine Learning Module, send results back to the Chrome extension, and present the recommended URL to the user. Figure 7 shows the code section for the Python Flask server. Figure 8 is the screenshot of presenting the recommended URL.



```
from flask import Flask
from flask_cors import CORS

app = Flask(__name__)
CORS(app)

cred = credentials.Certificate("ServiceKey.json")
firebase_admin.initialize_app(cred)

db = firestore.client()

print("ready")

@app.route("/takeInputOutput/<month>/<day>/<hour>/<minute>/<url>/<email>/")
def takeInputOutput(month, day, hour, minute, url, email):

  try:
    print("Reading from " + email)
    input_data = []
    output_data = []
```

Figure 7. Code Sample for Python Flask Server



NEXT URL:
https://docs.google.com/forms/d/

Reccomended Site

Figure 8. Screenshot of Presenting the Recommended URL

4. Experiment

Three Classification ML algorithms have been explored: SVM Support Vector Classifier (SVC), Passive Aggressive Classifier (PAC), and Random Forest Classifier (RFC). The purpose is to identify a classification model that supports better prediction. Sklearn K-fold cross-validation is used to calculate the accuracy of the different learning algorithms. Figure 9 and Figure 10 show the Cross-validation (CV) score for each test and the final average. The SVM machine learning model creates vectors with large margins in between the data. It essentially creates sections of fit for all the data points graphed. A Passive Aggressive model is similar to SVM in that it is also a vector classifier; however, it will strongly adjust itself whenever it is incorrect. It will keep doing this until it is correct. Random Forest is a decision tree machine learning model that uses slight random variance when it splits a decision. Random Forest in particular uses this to its advantage and creates multiple slightly different decision trees and averages the result.

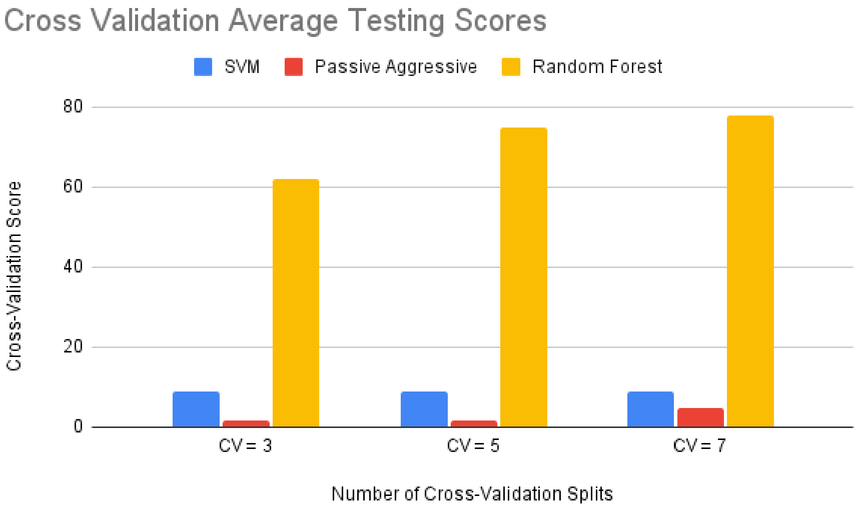


Figure 9. Average Cross-Validation Score for Three Machine Learning Algorithms

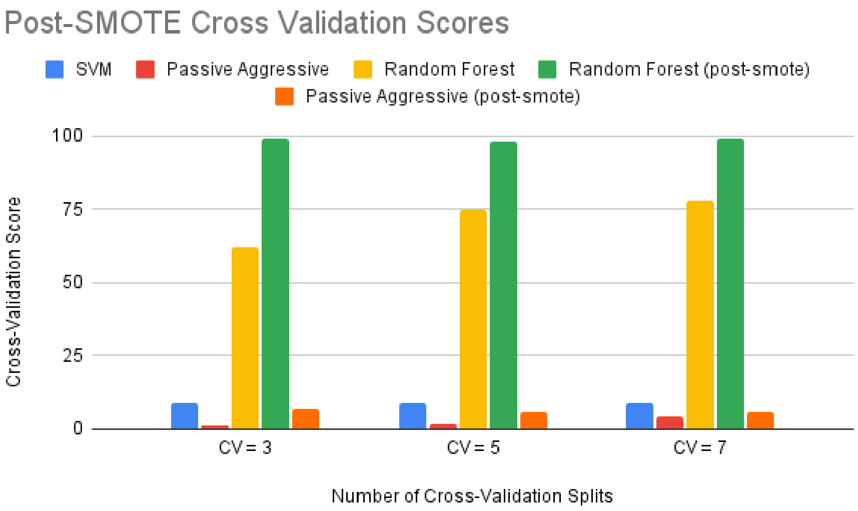


Figure 10. Average Cross-Validation Score for Three Machine Learning Algorithms post SMOTE included

As can be seen from the charts, the CV scores indicate that the Random Forest has the highest accuracy among the three ML algorithms. In some cases, the Passive Aggressive has a similar accuracy score as Random Forest, but it is not as consistent. Random Forest is the best decision for the current datasets not only because it has the highest accuracy but also because it is designed to handle imbalanced datasets very well. Even then post-SMOTE it did a good job. Order of events: 1. Our data was 99% accurate because it was so imbalanced. 2. We used random forest to help with the balanced data-set, but it still resulted in a “no tab” prediction 99% of the time. 3. We then decided to use SMOTE to balance our data-set. This got us a more realistic accuracy of around 65%. As we ran the extension over time we gathered a much larger data-set that helped with the initial imbalances we had. Because of this, with SMOTE, our accuracy shot up to 98% accuracy. With random forest, SMOTE, and now a much broader data-set, it is likely that the model is returning such a high result because it is over-fitted. This over-fitting, while normally a problem, is not a negative outcome because the project is associated with your personal browsing habits and should not base its prediction on anything else. As we now have a much larger data-set, SMOTE is not as necessary. And if we remove it, we get less over-fitting with an accuracy of about 77% accuracy.

Cross Validation results were the determining factor (as well as our own testing) with ~60% accuracy using the Random Forest Classifier. The prediction accuracy without SMOTE is around 99% accurate, as the data was unbalanced, with most results being the same link. It would return “No Tab” as a prediction and get it correct most of the time as most predictions had “No Tab” after it as nothing was opened after a 2-minute period. The SMOTE filled the imbalanced data set, helping the cross-validation produce a more accurate result of 60%.

SVM Feature Importance is utilized to determine how relevant the data is when it comes to predicting the next URL. It checks how many features or properties are in the data, calculates the importance rank of a feature by changing the value of that feature to see the impact on the end results, and returns a percentage indicating the weight of a feature for the prediction. Figure 11 shows the importance of Month, Day, Hour, Minute, and Second collected for Chrome Tab info, followed by the detailed table.

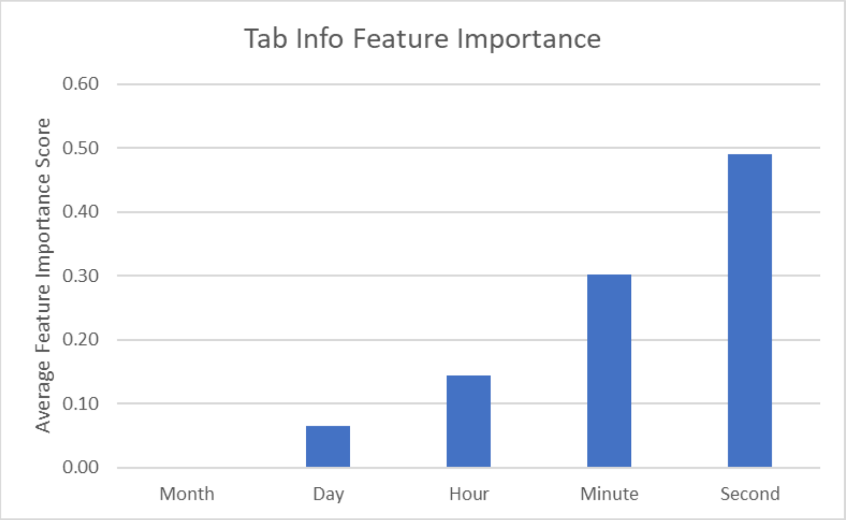


Figure 11. Feature Importance for Chrome Tab URL Time Info

Table 1. Feature Importance for Chrome Tab URL Time Info

Feature Importance	Month	Day	Hour	Minute	Second
#1	0.00000000	0.06086918	0.14408149	0.30362221	0.49142712
#2	0.00000000	0.06527741	0.14247498	0.30155574	0.49069188
#3	0.00000000	0.06473688	0.14882411	0.30013972	0.48629929
#4	0.00000000	0.06289494	0.14125969	0.30368546	0.49215991
#5	0.00000000	0.06487327	0.14333041	0.29936943	0.49242689
#6	0.00000000	0.06380527	0.14345727	0.30237818	0.49035928
#7	0.00000000	0.06240662	0.14423776	0.30371136	0.48964426
#8	0.00000000	0.06759765	0.14065858	0.30333982	0.48840396
#9	0.00000000	0.06714231	0.14755350	0.30054375	0.48476044
#10	0.00000000	0.06529274	0.14443936	0.30105877	0.48920913
Average	0.00000000	0.06448963	0.14403172	0.30194044	0.48953822

As can be seen from the chart and table, the Second and Minute have the highest feature importance, indicating they are weighted more when finding the best URL for the next prediction. The reason why the Feature Importance of the month is 0, is because the data used to calculate the feature importance was all in the same month. The reason it is not removed is that the algorithm will eventually incorporate months into its predictions.

Before feeding into the Machine Learning Classifiers, we organize the browser tab history data from the Firebase into sorted URLs based on months, days, hours, minutes, seconds, milliseconds, and user email. We go through each instance of the sorted URLs and create input datasets for the ML module, as well as generate the output data-set based on the input data-set.

In order to find the best time interval for the prediction, we executed tests with various levels of intervals such as 30 seconds, 1 minute, 2 minutes, 5 minutes, and 10 minutes. 2 minutes was picked because it is a relatively realistic time for users to reopen something related to their own work. The ML model checks if two URLs were opened within 2 minutes of one another, and if they were, the prediction for that site would be the one opened second. The months and days are used to check for consistency between the URLs that are being predicted. Milliseconds are used to make sure that two URLs are not opened within the same second.

Since our application is targeted for students and people who have repetitive tab opening activities for their daily online classes or work assignments, a lot of times these users do not have an immediate follow up tab to be opened. Also, the tab activities vary between users. These behaviors lead to large, broad, and overly thin datasets in the Firebase. Due to these imbalanced datasets, the ML model tends to predict a blank page or singular prediction. To relieve this type of biased prediction, we utilize the SMOTE technique to balance the datasets. SMOTE is a type of data augmentation by randomly increasing minority class examples by replicating them. We set the k nearest neighbor value to 2, meaning each minority instance will find its 2 nearest neighbors using Euclidean distance [17]. By adding multiple of the minority classes, the data-set becomes more balanced. Results reveal higher accuracy rates after applying SMOTE.

5. Related Work

Shawon and Zuhori propose two methods to make a web browser more intelligent [9]. They used linear regression to calculate website access frequency based on first visit, last visit, and URL counts, sorted the frequency, and then predicted the web link with highest frequency when a user typed in the address bar. They also provide content recommendation by classifying the URLs into several categories (Computers, Arts, Business, Games, etc.) using optimized Naïve Bayes Classifier and then suggesting a list of websites for a particular category. Instead of frequency, our study uses the time frame of the web history URLs as the input data for the Machine Learning module.

Kotapalle and Kandala built a cross-browser plugin, a client-side browser history management extension, that saved the user browsing history and the relationship of tabs in a tree mode using IndexedDB and local storage on the client [10]. They provided a front-end visualization of users' browsing behavior in a tree view or linear structure. Their extension was not yet connected to any back-end Machine Learning system. Our application focuses on Google Chrome extension, and it is integrated with a Cloud data-store and Machine Learning engine.

Woo and Lee proposed a web interaction profiling framework for real time interaction prediction [11]. They developed an event tracing tool to collect both users' navigation and click events using JavaScript event handlers. They adopted Gated Recurrent Unit (GRU) deep learning with URL grouping and Web embedding techniques for the interaction prediction. Their approach collected users' browsing activities at a very detailed level. It seems their framework fits better for commercial web applications to provide recommendations during user interaction with the websites. Our application uses a JavaScript event listener to collect the browser tab URL information only and leverage Random Forest classification.

6. Conclusions

In this paper we developed a Google Chrome extension tool that uses Machine Learning to predict future URLs to open and prompt the recommendation to the front-end users. We use JavaScript and Chrome API to create the Chrome extension, store tab events and history into the Firebase Cloud Firestore, and integrate the Chrome extension, Firebase, Machine Learning module through the Python Flask web server. The Machine Learning module is built using Sklearn libraries and SVM in Python. Various machine learning models such as Support Vector, Passive Aggressive, and Random Forest were executed and cross-validated to identify the method with a more accurate prediction. Feature Importance is calculated for the time granularity of Month, Day, till Second. In addition, we stacked SMOTE technique and Random Forest classification to make the thin and imbalanced data-set more balanced and achieve better accuracy. With the current solution model, the system is able to output the predicted URL at a decent prediction accuracy.

The limitation of our study is currently we experiment with a small number of users. Future work involves increasing the Machine Learning prediction performance as accuracy is always the king, testing, and enhancing the solution to be salable to a larger number of users. This could be done via modifying our back-end to incorporate parallel processing and adding more powerful servers. Moreover, we also want to add some adaptive modules using unsupervised Machine Learning techniques so that whenever the user opens or ignores the recommended new tab, we will update the machine learning model with that result. This could be added into our extension's background scripts, with the model being updated once or twice a week.

REFERENCES

[1] Portugal, Ivens, Alencar, Paulo, and Cowan, Donald. "The use of machine learning algorithms in recommender systems: A systematic review." Expert Systems with Applications, Volume 97, 2018, Pages 205-227, ISSN 0957-4174. https://doi.org/10.1016/j.eswa.2017.12.020. (https://www.sciencedirect.com/science/article/pii/S0957417417308333)

[2] Gangurde, R. and Kumar, B. "Web Page Prediction Using Genetic Algorithm and Logistic Regression based on Weblog and Web Content Features," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 68-74, doi: 10.1109/ICESC48915.2020.9155634.

[3] Yu, X., Chu, Y., Jiang, F., Guo, Y., and Gong, D. "SVMs Classification Based Two-side Cross Domain Collaborative Filtering by inferring intrinsic user and item features. "Knowl. Based Syst. 141 (2018): 80-91.

[4] Wanaskar, U., Vij, S., and Mukhopadhyay, D. "A hybrid web recommendation system based on the improved association rule mining algorithm." arXiv preprint arXiv:1311.7204 (2013).

[5] Geetha, G., Safa, M., Fancy, C., and Saranya, D. "A hybrid approach using collaborative filtering and content based filtering for recommender system." Journal of Physics: Conference Series. Vol. 1000. No. 1. IOP Publishing, 2018.

[6] Bhavithra, J. and Saradha, A. "Personalized web page recommendation using case-based clustering and weighted association rule mining." Cluster Comput 22, 6991 –7002 (2019). https://doi.org/10.1007/s10586-018-2053-y

[7] Hernández, I., Rivero, C. R., Ruiz, D., and Corchuelo, R. "CALA: ClAssifying Links Automatically based on their URL." Journal of Systems and Software, 115(2016), 130-143

[8] Rajalakshmi, R.and Xaviar, S. "Experimental Study of Feature Weighting Techniques for URL Based Webpage Classification." Procedia Computer Science, Volume 115, 2017, Pages 218-225, ISSN 1877-0509. https://doi.org/10.1016/j.procs.2017.09.128.

[9] Shawon, A., Zuhori, S. T., Mahmud, F., and Rahman, M. "Web Links Prediction and Category-Wise Recommendation Based on Browser History." arXiv preprint arXiv:1902.08496 (2019).

[10] Kotapalle, G., Kandala, H., and Gade, K. "Extracting relationship between browser history items for improved client-side analytics and recommendations." 2018 3rd International Conference on Contemporary Computing and Informatics (IC3I), 2018, pp. 141-146, doi: 10.1109/IC3I44769.2018.9007258.

[11] Joo, Minwoo and Lee, Wonjun. "WebProfiler: User interaction prediction framework for Web applications." IEEE Access 7 (2019): 154946-154958.

[12] Papadakis, G., Kawase, R., Herder, E. et al. "Methods for web revisitation prediction: survey and experimentation." User Model User-Adap Inter 25, 331 –369 (2015). https://doi.org/10.1007/s11257-015-9161-7

[13] Uddin, I., Khusro, S., Ullah, I., and Rauf, A. "Semantic History: Ontology-Based Modeling of Users' Web Browsing Behaviors for Improved Web Page Revisitation." In Proceedings of the Computational Methods in Systems and Software (pp. 204-215). Springer, Cham, 2018. https://doi.org/10.1007/978-3-030-00184-1_19

[14] Müller, Andreas C., and Guido, Sarah. "Introduction to machine learning with Python: a guide for data scientists." O'Reilly Media, Inc.", 2016.

[15] "Learning model building in Scikit-learn: A Python machine learning library." (2019, August 06). Retrieved February 28, 2021, from https://www.geeksforgeeks.org/learning-model-building-scikit-learn-python-machine-learning-library/

[16] "ML | Handling Imbalanced Data with SMOTE and Near Miss Algorithm in Python." Retrieved June 28, 2021, from https://www.geeksforgeeks.org/ml-handling-imbalanced-data-with-smote-and-near-miss-algorithm-in-python/

[17] Hackeling, Gavin. "Mastering Machine Learning with scikit-learn." Packt Publishing Ltd, 2017.

[18] Berke-Williams, G. "How to make a chrome extension." (2019, March 23). Retrieved February 28, 2021, from https://thoughtbot.com/blog/how-to-make-a-chrome-extension

[19] Grinberg, Miguel. "Flask web development: developing web applications with python." O'Reilly Media, Inc., 2018.

[20] Hajian, Majid. "Deploying to Firebase as the Back End." 10.1007/978-1-4842-4448-7_2, 2019.

[21] "What are some alternatives to Firebase? (n.d.)." Retrieved February 28, 2021, from https://stackshare.io/firebase/alternatives

[22] Dr. Garbade, Michael J. "Top 8 open source AI technologies in machine learning." (May 15, 2018). Retrieved February 28, 2021, from https://opensource.com/article/18/5/top-8-open-source-ai-technologies-machine-learning

HANDLING TRUST IN A CLOUD BASED MULTI AGENT SYSTEM

Imen Bouabdallah[1] and Hakima Mellah[2]

[1]Department of Computer Science, USTHB, Bab ezzouar, Algeria
[2]Information and multimedia system Department,
CERIST, Ben Aknoun, Algeria

ABSTRACT

Cloud computing is an opened and distributed network that guarantees access to a large amount of data and IT infrastructure at several levels (software, hardware...). With the increase demand, handling clients' needs is getting increasingly challenging. Responding to all requesting clients could lead to security breaches, and since it is the provider's responsibility to secure not only the offered cloud services but also the data, it is important to ensure clients reliability. Although filtering clients in the cloud is not so common, it is required to assure cloud safety.

In this paper, by implementing multi agent systems in the cloud to handle interactions for the providers, trust is introduced at agent level to filtrate the clients asking for services by using Particle Swarm Optimization and acquaintance knowledge to determine malicious and untrustworthy clients. The selection depends on previous knowledge and overall rating of trusted peers. The conducted experiments show that the model outputs relevant results, and even with a small number of peers, the framework is able to converge to the best solution. The model presented in this paper is a part of ongoing work to adapt interactions in the cloud.

KEYWORDS

Multi agent system, cloud, trust, interaction, PSO.

1. INTRODUCTION

Due to the pandemic of Covid-19 and the increase of remote work, the use of cloud computing has been highly increased [1] [2], leading to a higher number of both cloud clients and cloud providers.

The high number of users can cause the emergence of malicious entities Identifying malicious agents between the large numbers all at once would be difficult due to the insufficient amount of information regarding the users. The non-detection of such entities can produce security problems and restrictions in cloud use [3].

To overcome the extant security flaws in the cloud many paradigms have been introduced to improve the protection levels or reinforcing security.

Multi agent system (MAS) is a distributed structure [4] that has been employed to solve complex problems such us detecting security flaws in multiple fields.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 241-255, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112319

MAS include multiple interacting autonomous agents existing in a shared environment. The agents are distributed all over the system and capable of remaining connected through their interactions mechanism. Their intelligence allows them to autonomously take actions in order to solve their assigned tasks and evolve in the system.

Since distribution is common in Multi Agent System (MAS) and the cloud, and both paradigms consist of interconnected agents (in MAS) or users (in the cloud), combining them would draw the strong aspects of both.

Yet, due to the openness of the cloud [5] (multiple entities exist and can leave or enter the network randomly) and diversity of users, the combination still present some flaws that was the reason for the lack of early research in this area.

Nowadays, MAS is explored through its strong features to respond and overcome the cloud's flaws. Multi Agent System features include autonomy, perception and awareness, reactivity, interaction, cooperation [5], and many more depending on the implementation. This work is a prelude for an adaptive interaction framework, the propose model uses interaction along with cooperation to share knowledge among agents to determine trustworthiness of cloud consumers.

In this paper, Multi Agent System interaction mechanisms are used along with Particle Swarm Optimization (PSO) in the cloud to determine the trustworthiness of the clients and eliminate the malicious users. The following section discusses some of the recent related works, section 3 presents a background to: define MAS and the cloud, introduce trust in the cloud and present the main actors of the system; section 4 discusses the proposed PSO-Trust model and section 6 presents the evaluation and experimentation results.

2. RELATED WORKS

2.1. Multi agent system across the cloud

MAS have successfully solved many problems in the cloud, and it is even more promising when combined with other promising techniques. E.g. enhancing security by detecting intrusions [6], decision-making, service discovery [7], service reservation [8] …

Implementing MAS in the cloud is using the agents of the system to represent the actors of the cloud system, execute their commands and negotiate instead of them.

Service discovery was studied in [9]; authors used a hybridization of MAS and web services. Where they presented sets of agents that act on behalf of cloud consumers and cloud providers to generate and agree to the contract, the service discovery agent acts on behalf of the client using semantic ontology to select the best-fitted service provider.

Despite the diverse works joining MAS with the cloud, the majority focus on the concept of agents' autonomy, but yet neglect a critical aspect: cooperation.

In this work, cooperation between agents is considered through knowledge sharing among acquaintance agents to provide more accurate data to the system. We make use of the interaction mechanisms of MAS to exchange relevant and up to date data.

2.2. Implementing trust in the cloud

Trust management in the cloud have been increasingly studied to improve both security and intrusion detection, and is employed for service discovery and recommendation.

In the cloud, privacy is a major concern for all users and providers. Most of the works considering trust in the cloud, employ it on the client side ([10] [11] [12]) to help choose a service provider.

Using users' feedback and behaviour in [11], they proposed an Evidence Trust model to select the trusted cloud service provider (CSP). The final assigned trust value is computed through cumulating feedback and behavioural trust and would determine the trustworthiness of a CSP. To detect false feedback they compared the previously submitted feedback from user with the current one. In this case, if the client is always giving a false feedback it will pass undetected.

Li, et al in [13] used trust along reputation in an IOT cloud-based to determine the trustworthiness of a cloud service, by using feedback rating of costumers and evaluation of security metrics.

Yet, the works on guaranteeing only trusted clients in the network are very insignificant; but since it's the provider's responsibility to secure the cloud, it is highly important to check clients' background before granting the service and experiencing the security problems (identity theft, data breach…), which this paper discuss.

For the reason that if a provider in the cloud is under attack or experiencing security flaws that would certainly influence clients data, but also provider's reputation which can hardly be changed, and may result in losing all users.

Mehraj and Banday discus in [14] an application of trust in cloud deployment using Zero-Trust engine where, instead of trust being handled by a role agent, they use the engine to determine trustworthy entities. As for end users, they relay on authentication to validate users' identity and therefore trustworthiness, which may not always be the source of user's credibility.

A dynamic multi-dimensional trust evaluation method was proposed in [15] to determine trustworthiness of cloud providers and cloud consumers. In this work, the trustworthiness is considered to define the honest behaviour of the consumer, and since it does not interact only with providers but also with: brokers, auditors and SLA agents, they collect compliance information from all entities to calculate trust.

2.3. Implementing trust using PSO

Bio-inspired algorithms follow real life principals and are evolutionary aiming to improve the quality of problem solving. They derive from the behaviour of large groups but still can be applied to small data and provide precise and accurate data [16].

Particle swarm optimization is based on social behaviour [17] and thus can make use of shared knowledge between entities of the system. It products precise results and does not require large data (compared to other algorithms).

For group decision making, [18] discuss how, by introducing trust, individuals in a group can be influenced by others to change their opinions and decisions. Individuals' opinions (or expert evaluations as given in the illustrating example) represent the particles of PSO that is employed to determine their optimum.

Authors argue, by comparing the proposed model to another without trust, that the drawbacks can be avoided using trust and making use of group intelligence.

In [19] PSO was introduced to optimize neural network (NN) parameters and use NN to determine costumers trust rate in the cloud, Although the experiments showed that the results were promising, the NN require large data and need training over multiple times to really outputs precise data. Unlike that, PSO-Trust model presented in this paper requires only small swarm population to converge to the best solution.

3. BACKGROUND

Multi agent systems and the cloud are the main technologies involved in this paper, the following section present a brief review of these paradigms.

3.1. Multi agent system

-

Multi agent system (MAS) [20] consists of multiple interacting intelligent agents that are capable of handling tasks and cooperating to solve complex problems. Each agent receives a different task and outputs a different action, that would eventually be collected to solve a complex problem representing the global goal of the system.

Intelligent Agents are autonomous and situated in an environment [21], they are capable of perceiving their surroundings, making and choosing their own decisions, and taking actions without assistance.

Agents are endowed of multiple features: autonomy, intelligence, interactivity, interaction… [21][5].

The latter is a mean of communication and cooperation; by interacting, agents would share knowledge or resources and would cooperate to accomplish complex tasks. Interaction's type changes with the goals and tasks. Negotiation, a type of interaction, is usually set up to settle over an agreement or to reach a compromise to outcome conflicts.

Agents' autonomy drew much attention to them in research area, and are therefore applied in various domain.

3.2. The cloud

Cloud computing [22] is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources.

The users in the cloud are distributed over the network yet interconnected and able to access the needed resources at any time remotely.

These users can be classified, NIST [22] presents an architecture of five main roles in the cloud. In our opinion, and while implementing MAS along the cloud, the roles could be brought to two main classes: users and helper agents. In this work, we only consider users; however, the presence or absence of helper agents would not effect the model.

As for users, two actors are identified: cloud clients (consumers): the users asking for services, and cloud service providers: the actors offering the services at a given price with some defined criteria.

Many security flaws are present at this level effecting clients' privacy and data, and providers database and reputation [3]. To encounter some of the difficulties, multi agent systems (MAS) are introduced to make use of agents' intelligence for handling complex tasks. Agents would negotiate for the provider and study the opponent before and during the iteration.

Agent's interaction strategy would have a direct impact on data security and user satisfaction that can be gained by ensuring the protection of client's privacy. Moreover, for that we introduced trust.

3.2.1. Trust in the cloud

In literature, trust is defined as the firm belief in the reliability of the received information or also as "the vesting of confidence in persons or abstract systems, made on the basis of a leap of faith which brackets ignorance or lack of information" [23]. Between agent, trust could be defined based on multiple criteria that differentiate based upon the context which might lead to having different definition; in some works, it is based on reputation from other agents [24], reliability of the agent, respect of the given information, insurance of privacy…

Trust and security are constantly associated, mostly because trust engendera feeling of safety that is earned, acquired or build on strong knowledge [25]. Trusting an entity implies asserting the credibility of information it shares and the identity it presents and thus guarantying its access to data or platforms, which is major security concern.

3.2.2. Actors in the cloud based MAS

In order to clarify the subsequent sections, we start here by defining the main actors of the cloud-based system.

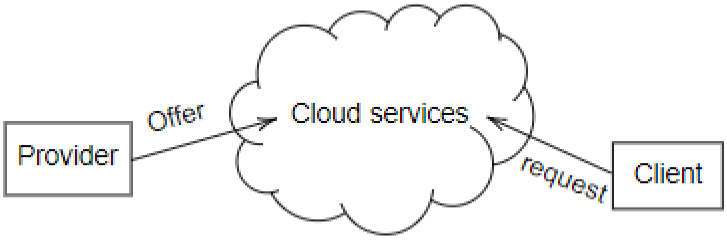


Figure 1. Main actors in the cloud

The cloud is a meeting environment for providers and clients (Figure 1), where the providers try to satisfy the clients need and expectations for services' quality.

Whereas clients (consumers) ask for services (software, database, collaborative environment…), providers work to deliver the best services at best cost and quality:

1. A cloud provider: delivers cloud computing services and solutions, responsible for making a service available for consumers [22],

2. A cloud consumer: could be individual or organization, using cloud services and resources, and maintain a business relationship with the provider. The consumer itself could represent a cloud provider.

In a cloud-based MAS, agents take actions for the cloud actors; client are represented by a client agent and providers by service agents (to handle the large amount of requests, multiple service agents are assigned to a single cloud provider (Figure 2)).

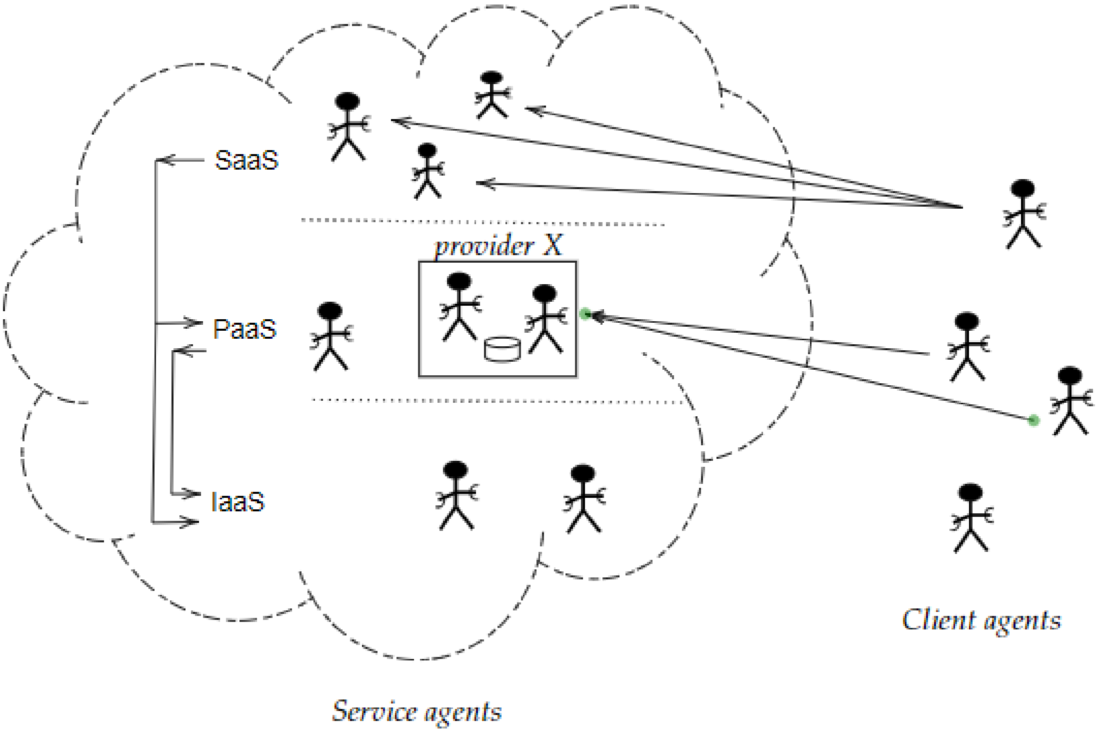


Figure 2. MAS cloud agents

- Client Agent (CA): carries cloud consumer request and handle the search and negotiation until service delivery,
- Service Agent (SA): represents the cloud service provider, it handles the consumers requests, it can also act as a cloud consumer to ask for complimentary services from other providers.

Agents conduct interactions instead of cloud actors to save time and effort because of agent's autonomy they are able to complete their assigned tasks without assistance.

For a service agent (SA: agent that guarantees services access) a client is trusted if it: respects the agreements, does not violate the cloud resources term of use and privacy, does not barging for so long that it cause famine (if the system does not have a timeout limit). On the other hand, a client could be less trusted if it has a bad reputation regarding the previous discussed points.

4. PSO-Trust Model

In the following section, we will present the proposed model starting by defining PSO.

4.1. Particle Swarm Optimization

Particle Swarm Optimization (PSO) [16] is a search algorithm [26], inspired by the social behaviour of bird flocks. Each individual in PSO represents a potential solution and is called a particle. The whole population is referred to as a swarm.

The algorithm aims to find the best solution (global best) from the swarm by running multiple iterations until the whole population converges or the maximum number of iterations is exceeded. Each particle possesses a position, and a velocity that represents the speed and direction by which the particle moves at each iteration.

At each iteration, velocity and position are updated through the following equations:

$$v(t+1) = v(t) + c_1 r_1(t)\left(p^{best}(t) - x(t)\right) + c_2 r_2(t)\left(g^{best}(t) - x(t)\right) \tag{1}$$

$$x(t+1) = x(t) + v(t+1) \tag{2}$$

Where:

- t : time/iteration
- v(t) : velocity at time stamp t
- x(t) : position
- c_1,c_2 : constants for nostalgia and envy (resp.), also called trust parameters
- r_1,r_2 : random vectors $\in[0,1]$
- p^{best}, g^{best}: personnel and global best (resp.)

When a particle moves to a new position, it is effected by three component: previous position, social component and cognitive component.

The whole population's (swarm) size effects the convergence of the algorithm for: the smaller the size the slower to converge, and the bigger the size the more space to explore.

4.2. Model flowchart

The following model (Figure 3) casts a presentation of the proposed PSO-Trust model.

Upon receiving a service request, and instead of start negotiation on the spot, the SA would first start by checking the client's trustworthiness by either retrieving its trust weight from the database (if the agent have had previous interactions with the client), or acquire it from its acquaintances (Figure 3).

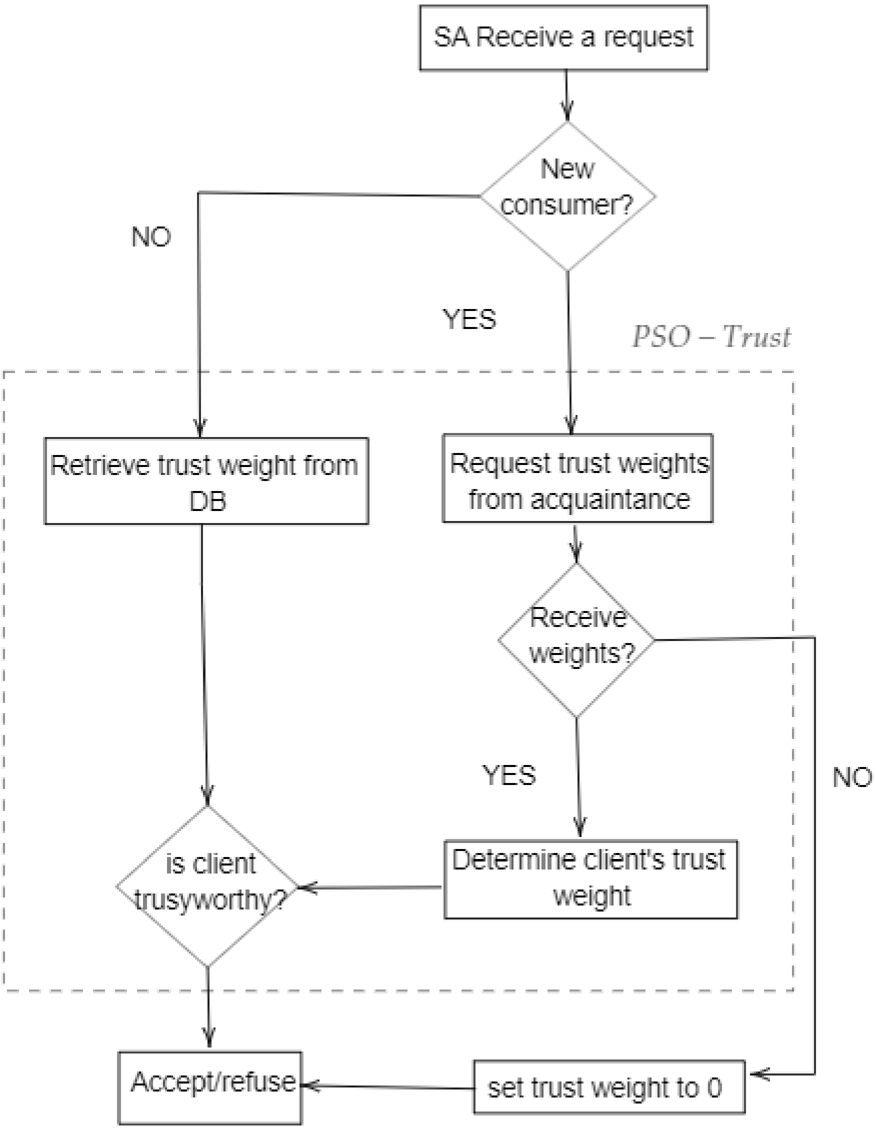


Figure 3. Steps to evaluating client's trustworthiness

If none of the acquaintances possesses information about the client, its trust weight is set to zero (as a neutral value). Otherwise, the PSO model is used to determine clients' trust weight, and is judged upon the resulting value whether to be trusted or not.

Trust values fall between [-1,1] instead of [0,1] since clients could be untrustworthy. (-1) denotes an untrustworthy agent and (1) denotes a trusty one. The median value (zero) is mostly used for agents with no history of interaction whatsoever.

4.3. Model presentation

The service agents (SA) are responsible for negotiating with the client and offering the needed services. When a SA is contacted by a new client C, and to ensure security, and to assign a trust weight for the new costumer, the SA would inquire its acquaintance send their trust weights of the client C. If present, the agent would receive several different weights and it would be hard to pick one from the large choice.

To overcome that, Particle Swarm Optimization is used to select the most appropriate weight in a 2D plan considering each particle as a tuple ($T_{a(x)}$, $R_{Ta(x)}$) where: $T_{A(x)}$: The trust weight of the agent A(x), and $R_{TA(x)}$: The trust value received from agent A(x).

The error of each particle is determined using Euclidian distance as a fitness function where the distance between each particle and the global best is calculated.

$$Fintess\ function = e = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (3)$$

Where: (x_1, y_1) are the particle's position and (x_2, y_2) the global best postion. The personnel best of the particles is updated if the error is minimal.

The algorithm outputs the global best of the population by generating the mean of all particles.

$$Global\ Best = \sum_{1}^{size} \frac{ParcticlePosition(i)}{size} \qquad (4)$$

Equation (4) assigns the same importance to all trust values nevertheless the positive ones should have more impact in the equation. In order to respond to that the equation is changed so that the particle position is multiplied by a variant as an impact factor (Θ) depending on the trust values:

$$Global\ Best = \frac{1}{size} \sum_{1}^{size} \Theta * ParcticlePosition(i) \qquad (5)$$

Since the population positions are between [-1,1] (to correspond to the trust weights), and so that the generated solution can be used immediately, the results of equation (5) are normalized in$[-1, 1]$.

$$Global\ Best' = 2 * \frac{GlobalBest - \min x}{\max x - \min x} - 1 \qquad (6)$$

With: x = Particle Position.

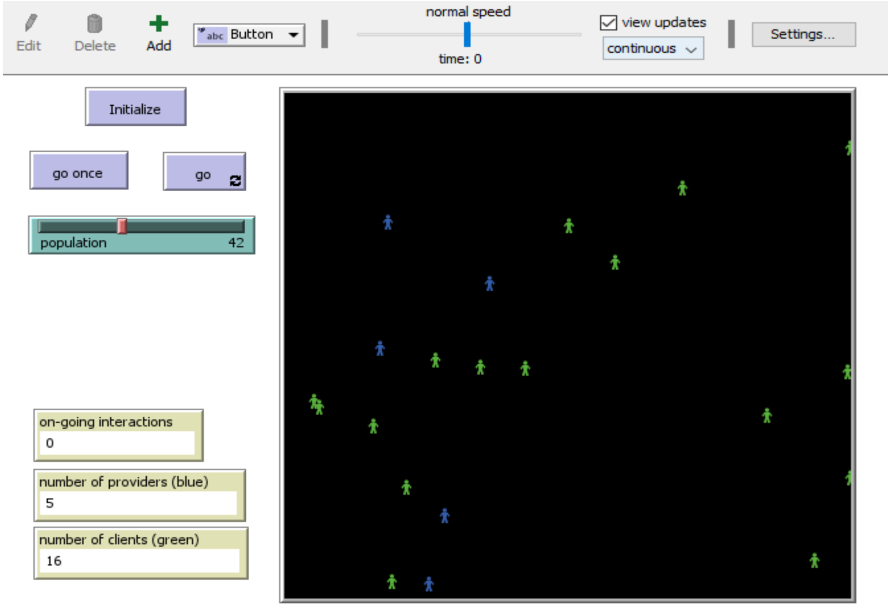
5. EVALUATION

First, to have a realistic view of the proposed model we started by implementing the framework in Netlogo [27].

Netlogo is a MAS modelling environment for simulating social and complex phenomenal in a world allowing the user to observe its state through time. It is mostly used by research to project the ideas into an interactive environment.

Because the cloud is open, we configure the number of agents in the population as variable, and could be increased or decreased by the user (increases even up to 150).

To have a better view of the world we chose a small population number for the illustrated experiment. At initialization (Figure 4), a random number of clients (in green) and providers (in blue) are present in the network. The "go" function allow the system to simulate the cloud environment interactions.



normal speed

time: 0

☑ view updates

continuous

population 42

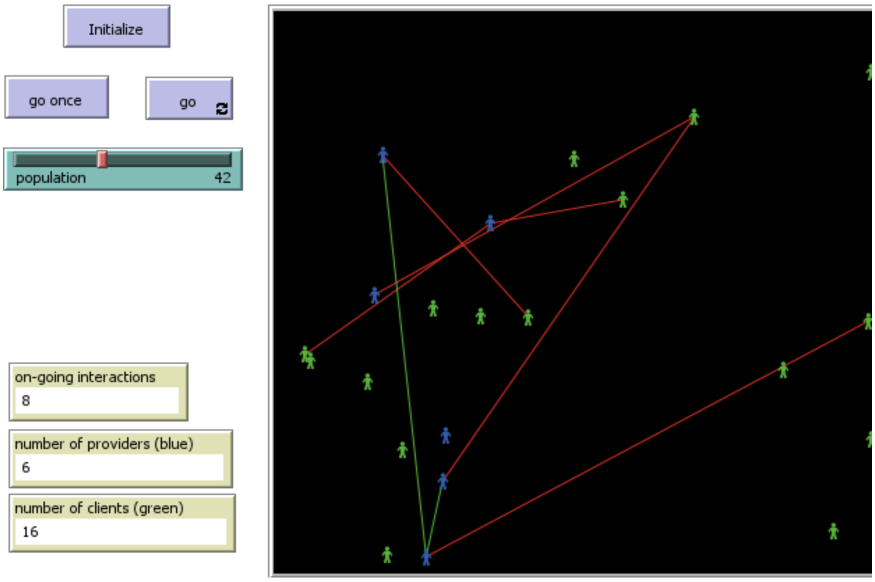
on-going interactions 0

number of providers (blue) 5

number of clients (green) 16

Figure 4. The Netlogo world after initialization

After running the program for some time (Figure 5), it can be notice how some providers need to contact their acquaintance after receiving requests from consumers, while others do not express this need. This is due to the presence or absence of data of the contacting clients in the providers' knowledge database.



population 42

on-going interactions 8

number of providers (blue) 6

number of clients (green) 16

Figure 5. Ongoing interactions in the cloud

The number of agents in the cloud can change during runtime due to the openness of the cloud (agents can enter or leave the environment randomly), but when the maximum number of agents in the population is reached no more agent can enter until others leave (since the population number is previously set by the user).

To experiment our work and evaluate the performance of the PSO-Trust model, we developed it under Python.

Python offers a huge number of libraries, frameworks, and even plotting library (matplolip) that ease creating animated and interactive visualisations. Due to the lack of clients' data (for the purpose of confidentiality), we used a random dataset to evaluate the model.

The population size may differentiate depending of the received trust weights and the number of agent's acquaintance. To test the algorithm efficiency, we used different population sizes in the experiments (between 5 and 100).

To illustrate experiment's results, we start by exploiting large population; the initial high sized population present in Figure 6 (where each particle is represented by a different colour each to ease observation) is able converges after a minimal number of iterations

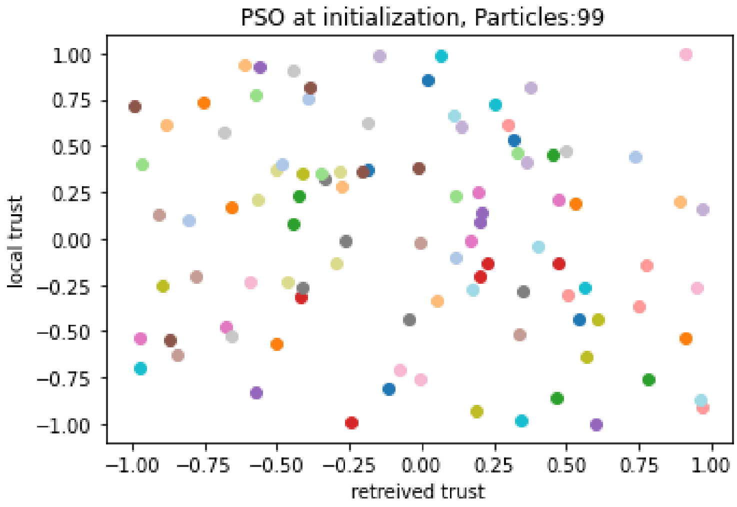


Figure 6. Particles' distribution with large population

Before conducting the experiments we stated by configuring the parameters from equation 1 and 2. While c1 represent how much confidence the particle have in itself (the local position), c2 represents how much confidence it have in its neighbourhood. Hence, since we need the particles to be more attracted to the global than personal best and converge fast, envy value is set slightly bigger than nostalgia (c2 >> c1).

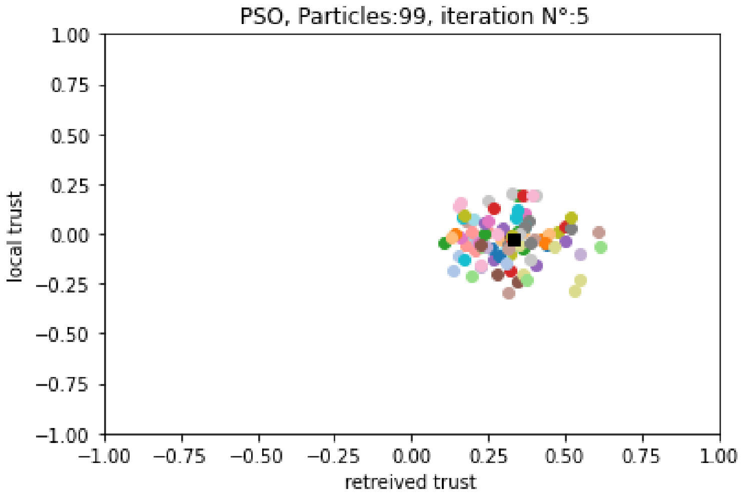


Figure 7. Convergence after five iterations.

The algorithm runs for a several number of iterations until the global best is reached or the error rate is minimum.

In Figure 7, after several iteration (five to be precise) particles have almost fully converge to the global best (represented by the black square).

Experiments also showed that even with a small swarm size (Figure 8).

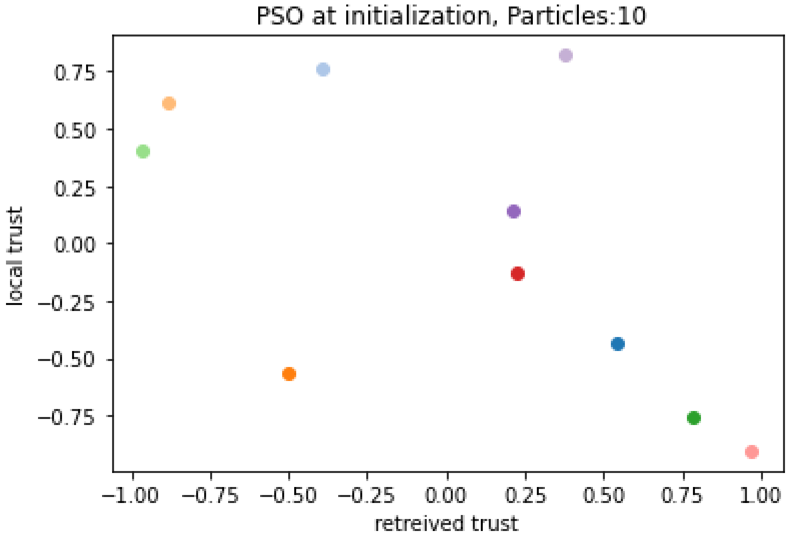


Figure 8. Small swarm size.

The algorithm succeeds this time too in finding the best trust weight value from the received values in Figure 9 (the black triangle in figure 9 represents the global best position).

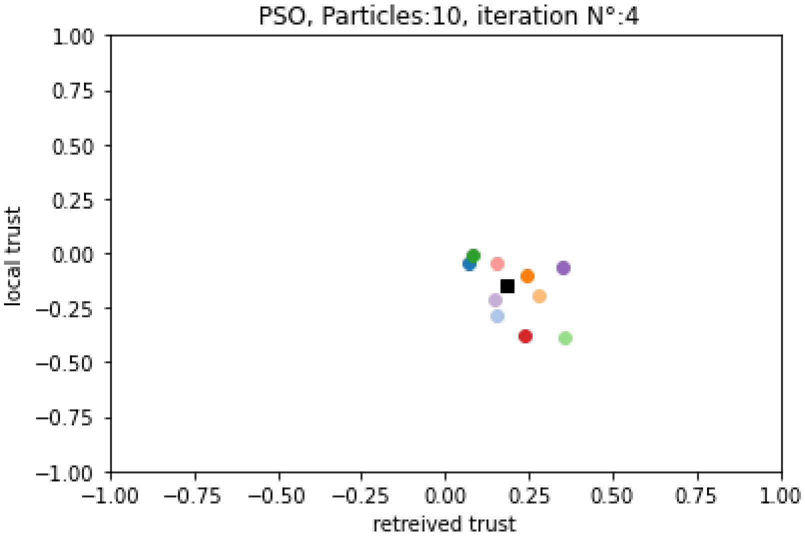


Figure 9. Swarm population near convergence

Therefore, even if the acquaintance offering information are numbered, the algorithm can still provide an accurate solution.

Because providers refuse to share their data (due to security concerns), we could not find data as to compare our proposed solution with other models and the generated data would not fit with the models description.

6. CONCLUSIONS AND FUTURE WORKS

Since it is the providers' responsibility to keep the data and clients secure, trust in the cloud should not be only considered by clients. Malicious users can attack the cloud and effect the data it stores, therefore, effect many clients and risk their privacy.

To secure the network and identify the clients that are worthy of trust, we proposed a PSO-Trust model where we make use of the interaction mechanisms of MAS to share reliable data.

One of the difficulties we faced is the lack of data, for this we aim to generate a database that represents clients' history and feedback from providers. For the next step, we would extend our work to construct an opponent interaction model. The current work would represent a pre-interaction phase to the interaction model and adapt the ongoing interactions based on the previous shared knowledge.

REFERENCES

[1] "En pleine pandémie, Google enregistre des profits records grâce à la publicité," [Online]. Available: https://www.france24.com/fr/amériques/20210428-en-pleine-pandémie-google-enregistre-des-profits-records-grâce-à-la-publicité. [Accessed 02 06 2021].

[2] "Wall Street : l'heure de la pause malgré Amazon," 2021. [Online]. Available: https://www.boursedirect.fr/fr/actualites/categorie/marche-us/wall-street-l-heure-de-la-pause-malgre-amazon-boursier. [Accessed 2 June 2021].

[3] Subramanian, Nalini & Jeyaraj, Andrews. (2018). Recent security challenges in cloud computing. Computers & Electrical Engineering. 71. 28-42. 10.1016/j.compeleceng.2018.06.006.

[4] C. Psaltis, "The True Meaning of 'Open Cloud' ", 2021 [Online]. Available: https://thenewstack.io/the-true-meaning-of-open-cloud/

[5] Ferber, Jacques. (2001). Multi-Agent System: An Introduction to Distributed Artificial Intelligence. J. Artificial Societies and Social Simulation. 4.

[6] O. Achbarou, M. Kiram, S. Elbouanani and O. Bourkoukou, "A New Distributed Intrusion Detection System Based on Multi-Agent System for Cloud Environment," International Journal of Communication Networks and Information Security, vol. 10, pp. 526 - 533, 2018.

[7] K. M. Sim, "Agent-Based Cloud Computing," in IEEE Transactions on Services Computing, vol. 5, no. 4, pp. 564-577, Fourth Quarter 2012, doi: 10.1109/TSC.2011.52.

[8] S. Son and K. Sim, "A Price- and-Time-Slot-Negotiation Mechanism for Cloud Service Reservations," IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics: a publication of the IEEE Systems, Man, and Cybernetics Society, vol. 42, pp. 13-28, 2011.

[9] Parhi, Manoranjan & Pattanayak, Binod & Patra, Manas. (2015). A Multi-agent-Based Framework for Cloud Service Description and Discovery Using Ontology. 10.1007/978-81-322-2012-1_35..

[10] A. M. Mohammed, E. I. Morsy and F. A. Omara, "Trust model for cloud service consumers," 2018 International Conference on Innovative Trends in Computer Engineering (ITCE), 2018, pp. 122-129, doi: 10.1109/ITCE.2018.8316610.

[11] Mujawar, Tabassum & Bhajantri, Lokesh. (2020). Behavior and Feedback based Trust Computation in Cloud Environment. Journal of King Saud University - Computer and Information Sciences. 10.1016/j.jksuci.2020.12.003.

[12] C. Mao, R. Lin, C. Xu and Q. He, "Towards a Trust Prediction Framework for Cloud Services Based on PSO-Driven Neural Network," IEEE Access, pp. 2187-2199, 2017.

[13] X. Li, Q. Wang, X. Lan, X. Chen, N. Zhang and D. Chen, "Enhancing Cloud-Based IoT Security Through Trustworthy Cloud Service: An Integration of Security and Reputation Approach," IEEE Access, vol. 7, pp. 9368-9383, 2019.

[14] S. Mehraj and M. T. Banday, "Establishing a Zero Trust Strategy in Cloud Computing Environment," 2020 International Conference on Computer Communication and Informatics (ICCCI), 2020, pp. 1-6, doi: 10.1109/ICCCI48352.2020.9104214.

[15] Challagidad, Praveen & Birje, Mahantesh. (2020). Multi-dimensional Dynamic Trust Evaluation Scheme for Cloud Environment. Computers & Security. 91. 101722. 10.1016/j.cose.2020.101722.

[16] Mukhopadhyay, Mayukh. (2014). A brief survey on bio inspired optimization algorithms for molecular docking. International Journal of Advances in Engineering & Technology. 7. 868-878. 10.7323/ijaet/v7_iss3.

[17] A. Engelbrecht, “Particle Swarm Optimization,” in Computational Intelligence: An Introduction, John Wiley & Sons, Ltd, 2007.

[18] Zhou, Xiaoyang & Ji, Feipeng & Wang, Liqin & Ma, Yanfang & Fujita, Hamido. (2020). Particle swarm optimization for trust relationship based social network group decision making under a probabilistic linguistic environment. Knowledge-Based Systems. 200. 105999. 10.1016/j.knosys.2020.105999.

[19] C. Mao, R. Lin, C. Xu and Q. He, “Towards a Trust Prediction Framework for Cloud Services Based on PSO-Driven Neural Network,” IEEE Access, vol. 5, 2017.

[20] G. Weiss, Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, Cambridge, MA, USA, 1999.

[21] M. Wooldridge and N. R. Jennings, “Intelligent agents: theory and practice,” The Knowledge Engineering Review, vol. 10, no. 2, p. 115–152, 1995.

[22] P. M. Mell and T. Grance, “SP 800-145. The NIST Definition of Cloud Computing,” National Institute of Standards & Technology, Gaithersburg, MD, USA, 2011.

[23] M. Guido, “TRUST: REASON, ROUTINE, REFLEXIVITY,” in SCARR Conference on Risk & Rationalities, Queens’ College Cambridge, 2007.

[24] J. Granatyr, V. Botelho, O. R. Lessing, E. E. Scalabrin, J.-P. Barthès and F. Enembreck, “Trust and Reputation Models for Multiagent Systems,” ACM Comput. Surv., vol. 48, no. 2, 2015.

[25] Lamsal, Pradip. (2002). Understanding Trust and Security.

[26] K. . J. and . E. R., “Particle swarm optimization,” in Proceedings of ICNN'95 - International Conference on Neural Networks, 1995.

[27] U. Wilensky, “NetLogo,” Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 1999.

A DAILY COVID-19 CASES PREDICTION SYSTEM USING DATA MINING AND MACHINE LEARNING ALGORITHM

Yiqi Jack Gao[1] and Yu Sun[2]

[1]Sage High School, Newport Coast, CA 92657
[2]California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

The start of 2020 marked the beginning of the deadly COVID-19 pandemic caused by the novel SARS-COV-2 from Wuhan, China. As of the time of writing, the virus had infected over 150 million people worldwide and resulted in more than 3.5 million global deaths. Accurate future predictions made through machine learning algorithms can be very useful as a guide for hospitals and policy makers to make adequate preparations and enact effective policies to combat the pandemic. This paper carries out a two pronged approach to analyzing COVID-19. First, the model utilizes the feature significance of random forest regressor to select eight of the most significant predictors (date, new tests, weekly hospital admissions, population density, total tests, total deaths, location, and total cases) for predicting daily increases of Covid-19 cases, highlighting potential target areas in order to achieve efficient pandemic responses. Then it utilizes machine learning algorithms such as linear regression, polynomial regression, and random forest regression to make accurate predictions of daily COVID-19 cases using a combination of this diverse range of predictors and proved to be competent at generating predictions with reasonable accuracy.

KEYWORDS

Covid-19 Case Prediction, Data Mining, Machine Learning Algorithm.

1. INTRODUCTION

On January 21st, 2020, Wuhan entered into lockdown after outbreaks of the novel coronavirus, SAR-COV-2 [1, 2] appeared, and not long after, on March 11st, 2020, the novel coronavirus pandemic was declared a global pandemic by the WHO [3]. Known for its severe pneumonia-like symptoms, the coronavirus rapidly spread throughout the world despite various preventative measures such as travel restrictions, social distancing mandates, and lockdowns [4, 5]. At the time of writing, the virus has infected over 150 million people worldwide and led to more than 3.5 million coronavirus related deaths [6]. The pandemic has placed a heavy burden on the world's medical systems, especially those already battling regional instabilities and lacking in adequate sanitation and medical supplies. However, many regions with advanced medical systems, such as Europe and the United States, were also hard hit because policy makers were not presented with enough information on the pandemic to adjust to the rapidly evolving situation [7, 8, 9]. Forecasts of the pandemic's development, though potentially inaccurate, help hospitals and policy makers make preparations to combat the spread of SAR-COV-2. In addition, while the lethality of the novel coronavirus cannot be understated, the virus also devastated the global economy due to repeated nationwide shutdowns and travel restrictions. The issue of shutdowns has since emerged as a heated point of political discourse regarding the efficacies of differing

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 257-268, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112320

legislative approaches towards the pandemic [10, 11]. Therefore, it is necessary to weigh the importance of different factors, natural and legislative, so policy makers can make informed decisions to efficiently combat the current pandemic and future pandemics to come.

Many studies have utilized machine learning algorithms such as regularized ridge regression and deep learning models such as ARIMA and SARIMAX to accurately forecast the spread of COVID-19 [12, 13]. Others have utilized a combination of random forest and Bayesian models to forecast daily COVID deaths and cases using both recorded and forecasted data [14]. While such predictions can be accurate even over longer periods, most only utilize total cases, total days, and previously recorded daily new cases as predictors in their forecasting models. Including more diverse predictors such as population density and daily available COVID tests would theoretically increase the accuracy of such predictions. Using Random Forest Regression to analyze the importance of individual predictors may yield additional insight into how legislative policies and demographic information influences the spread of COVID-19.

In this paper, we follow the same line of research and utilize three regressive models from the sklearn machine learning library—linear regression, polynomial regression, and random forest regression—to predict daily changes in COVID-19 cases based on previously recorded data from the COVID-19 data set maintained by Our World in Data. The prediction models are coded in Python and the data file is read using pandas and modified using a label encoder, along with other manually written codes, to remove nan values and select specific parts of the data file based on its date. In comparison to other works, we fitted our model with a wider range of potential predictors such as the daily amount of COVID tests, ICU capacity, and population density. These additions should increase the accuracy of the predictions if they also have a correlation to the overall trend of recorded COVID-19 cases. The use of random forest regression to analyze the importance of various predictors may yield insight into the effects of aregion's demographic and societal factors on the recorded trends of COVID-19.

The results of the experiment were proven primarily using cross validation where a cross validation score was assigned based on the model, input data, and output data. Additionally, we qualitatively examined the accuracy of the model by making a prediction using the input data from a select day and later proved that the predictions made by the model were reasonably close to the actual recorded values for that day.

The rest of the paper is organized as follows: Section 2 focuses on the difficulties in organizing the code and building the model; Section 3 gives more details on how some of the difficulties were addressed and how the model functions; Section 4 further elaborates on our findings regarding the predictions we made with our model; Section 5 discusses related works.Finally, Section 6 provides concluding remarks and suggested future work for the project.

2. CHALLENGES

In order to use the feature significance of random forest regression to compare the influence of the individual predictors on the general trend of COVID-19, a few challenges have been identified as follows.

2.1. Challenge 1: Missing Data Points and "nan" Values

The data file we selected for the analysis was ambitious in that it included a plethora of relevant data points such as the density of handwashing facilities, the proportion of the population in certain age groups, the proportion of smokers within the population, and other factors. Though a variety of data points is certainly beneficial for the analysis process, it also presented the greatest challenges in the form of “nan” values, which are added whenever a datapoint is not present for a certain country. Before continuing, it is worth mentioning that the data file referred to is a two-dimensional array with rows representing a day in a certain country and columns representing the data points on that day from each category listed. Since the data file is ambitious in its categories of data, every row of data contains numerous “nan” values where the data points cannot be found. First, we contemplated replacing the “nan” values with the mean value of the values above and below it. However, we discarded the idea since in most situations, the data points did not exist across all the rows representing days in the country. Replacing the datapoints with the mean value of the data points on the left and right would also cause inaccuracies since they would be from unrelated categories. Secondly, we considered replacing all the “nan” values with zeros, but this would also lead to inaccuracies since most data points are “nan” values not because they are zero, but because there are no data points from that category in that region to be found. An example of this is the number of handwashing facilities which exists as a “nan” value in roughly half of the rows. Handwashing facilities are clearly present in virtually every country, and thus replacing “nan” values with zeros in this case clearly does not make sense because it will negatively influence the correlation between the number of handwashing facilities and the number of daily COVID cases. In the end, we decided on a model where all rows containing “nan” values for its columns or categories of interest are removed. Although this is a crude way of organizing data, it is sufficient to generate accurate statistical correlations between categories of interest to which we can apply various models to generate predictions regarding daily COVID cases.

2.2. Challenge 2: The Introduction of Viral Mutations

The second challenge was concerning the approach to data analysis and allowing for the emergence of more infectious and potentially more deadly COVID-19 variants. The core purpose of this research was to compare the influences of various factors on daily COVID-19 cases and generate predictions based on the existing data points. However, the emergence and spread of new variants in late 2020 introduced an additional layer of unpredictability and possible error into the analysis. For example, the spike in COVID-19 cases between September of 2020 to December of 2020 may be attributed to a change in the virulence and contagiousness of the virus, though the spread of mutations operates on similar principles. An analysis of the data points without considering changes in the virus itself may yield a trend that is nonexistent or exaggerated. To account for this, the data points analyzed were divided into different time frames, and specifically focused on earlier time frames between the start of the pandemic until August, when relevant data points such as total cases and daily testing numbers had the clearest correlation with daily increases in COVID-19 cases.

2.3. Challenge 3: Selecting the Preferred Predictive Model

Finally, another challenge was selecting the preferred predictive model to use for analysis. Linear Regression, Polynomial Regression, and Random Forest Regression are easy to work with and potentially effective methods of data analysis. In the end, though it takes more computational power and some data structures are less compatible with certain models of analysis, we decided to use all three models and compare the accuracy of the various predictions generated. We believe having data from three predictive models is valuable to allow for comparison among their analytical methods and accuracy.

3. SOLUTION

In this paper, we focus on predicting the daily increases in COVID-19 cases through various predictors we provide to the model (see Figure 1). The overall experimental process can be separated into four distinct steps: data preparation, the creation of input and output data, prediction and validation, and the recording of results. Pandas is first used to convert the CSV file into a data frame where it is later turned into two dimensional lists through the values.tolist() command. The output depended on the predictors selected, which ranged from date and country name to new hospital admissions and total cases.

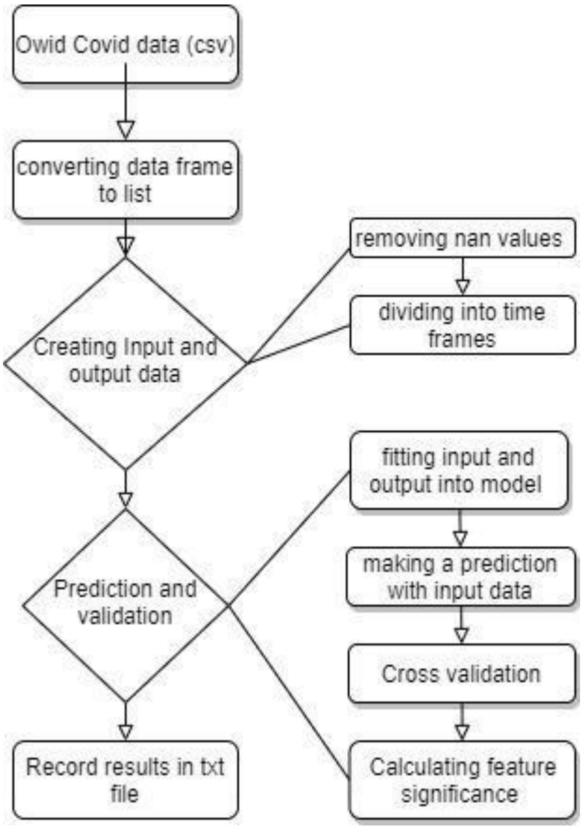
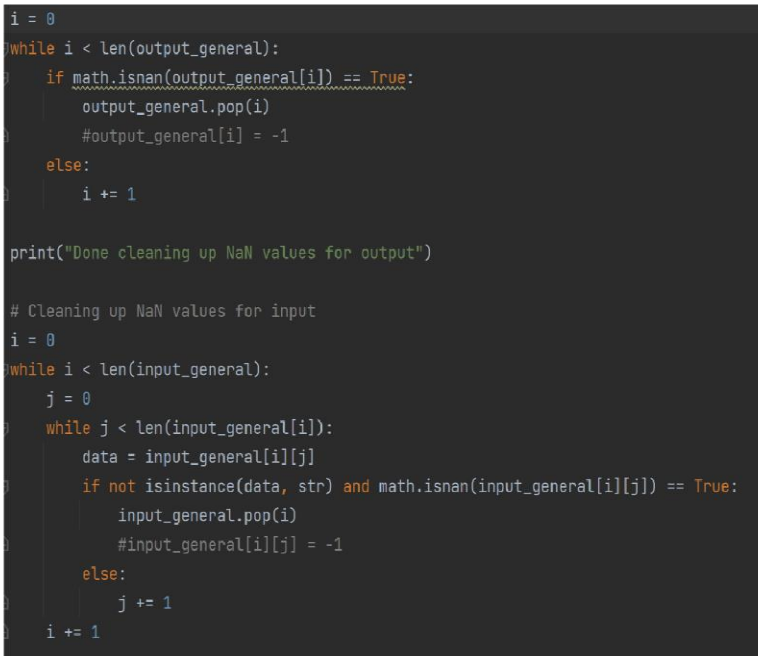

Owid Covid data (csv)
converting data frame to list
Creating Input and output data
removing nan values
dividing into time frames
Prediction and validation
fitting input and output into model
making a prediction with input data
Cross validation
Calculating feature significance
Record results in txt file

Figure 1. Schematic of the predictive model

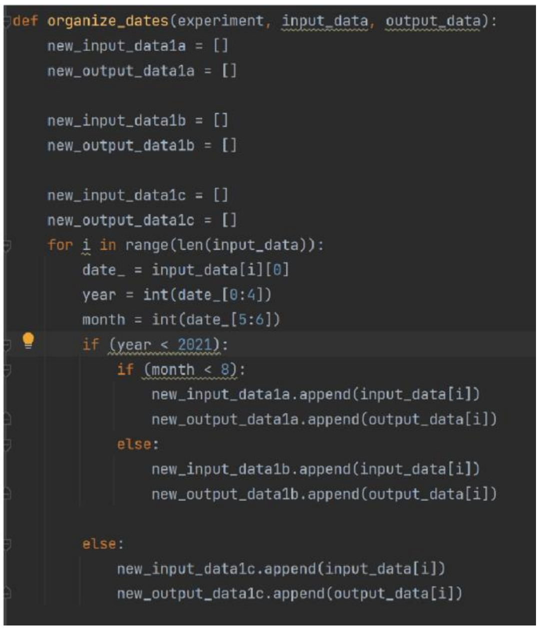
Demographic data such as median age, density of handwashing facilities, and population density, along with other factors were also included. The date must be the first variable inputted since this is essential to assigning time frames to other data points. Any non-integer values such as date and country name are converted using the fit_transform function of the label encoder.



```python
i = 0
while i < len(output_general):
    if math.isnan(output_general[i]) == True:
        output_general.pop(i)
        #output_general[i] = -1
    else:
        i += 1

print("Done cleaning up NaN values for output")

# Cleaning up NaN values for input
i = 0
while i < len(input_general):
    j = 0
    while j < len(input_general[i]):
        data = input_general[i][j]
        if not isinstance(data, str) and math.isnan(input_general[i][j]) == True:
            input_general.pop(i)
            #input_general[i][j] = -1
        else:
            j += 1
    i += 1
```

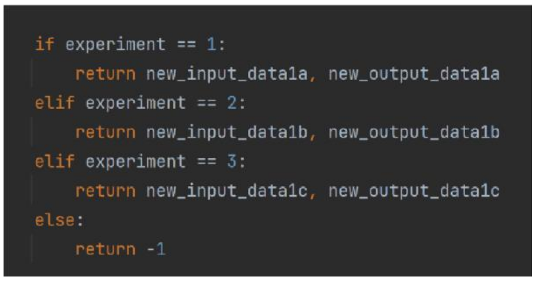
Figure 2. Nan values adjusted for output

Cleaning up nan values in each input and output data is done with while loops (see Figure 2). Nan values are found using the isnan() function of the math class, and if the row contains any nan values, the entire row will be removed as seen above in Figure 2. An alternative way of removing nan values is shown in Figure 3, where all nan values are replaced with -1. This is not preferable because it may potentially establish erroneous trends since the machine learning algorithm will use -1 as an input value when the actual value is not -1 and is simply unrecorded. The alternative method of removing nan values is only used when the primary way results in all rows being removed. This occurs with less documented predictors like the number of hand washing facilities or smokers in a population. Dividing the data into time frames requires another algorithm as shown in Figure 4. The input and output data are divided into three separate lists depending on experiment number: the function will output data from the start of pandemic to August of 2020 with an experiment number of 1; August of 2020 to December of 2020 with an experiment number of 2; and December of 2020 to latest time available, which by the time of testing was early March of 2021, when the experiment number is 3. The function is called as a part of the function of each individual model. This is done to account for the unquantifiable effects of the differing virulence of the different strains of COVID-19 such as the UK and South African variants, which were more deadly than the original COVID-19 variant. However, this difference is not seen within the separate tests possibly because the virulence of the different COVID-19 strains has little impact on the overall correlation between some predictors and daily COVID cases.



```python
def organize_dates(experiment, input_data, output_data):
    new_input_data1a = []
    new_output_data1a = []

    new_input_data1b = []
    new_output_data1b = []

    new_input_data1c = []
    new_output_data1c = []
    for i in range(len(input_data)):
        date_ = input_data[i][0]
        year = int(date_[0:4])
        month = int(date_[5:6])
        if (year < 2021):
            if (month < 8):
                new_input_data1a.append(input_data[i])
                new_output_data1a.append(output_data[i])
            else:
                new_input_data1b.append(input_data[i])
                new_output_data1b.append(output_data[i])

        else:
            new_input_data1c.append(input_data[i])
            new_output_data1c.append(output_data[i])
```

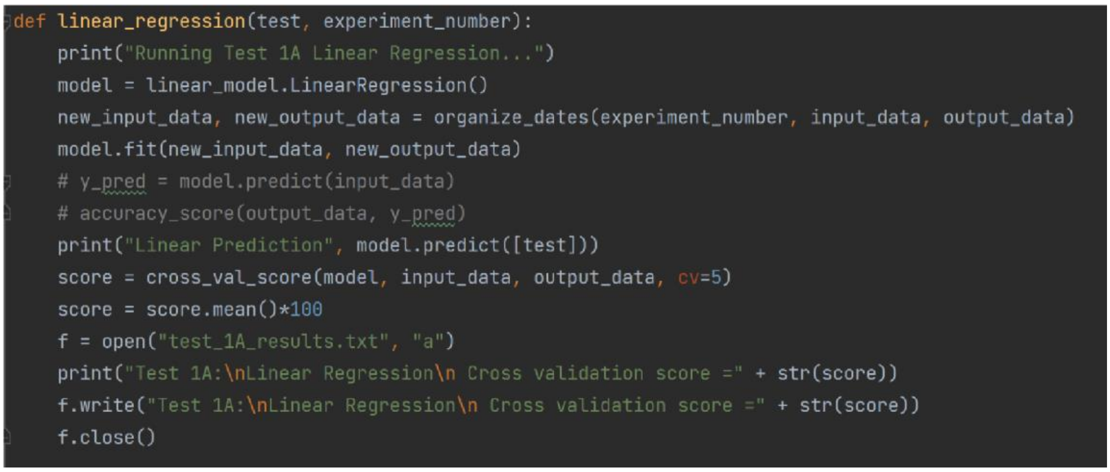
Figure 3. Nan values replaced with -1



```python
    if experiment == 1:
        return new_input_data1a, new_output_data1a
    elif experiment == 2:
        return new_input_data1b, new_output_data1b
    elif experiment == 3:
        return new_input_data1c, new_output_data1c
    else:
        return -1
```

Figure 4. Dividing data into time frames

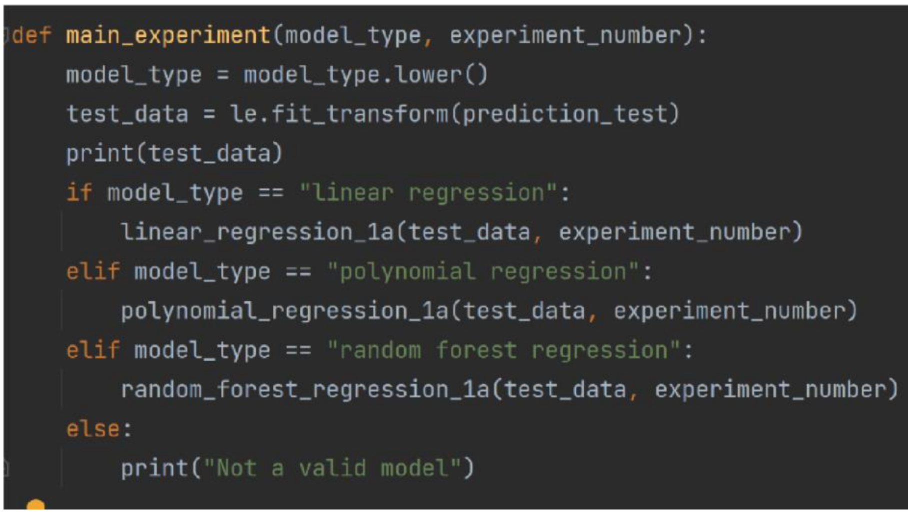
The functions for each of the three machine learning algorithms are almost identical to the one shown in Figure 5, which shows the function for linear regression. First the model is defined to be linear regression and the organize_dates function is called, outputting two lists for input and output data depending on the time frame selected as indicated by the experiment number. Then the new_input_data and new_output_datais used to train the model. The different models are set up appropriately based on their specific design. The polynomial regression is implemented initially with a polynomial feature of 2, although that is changed if the trend can be more accurately predicted with a different value for the polynomial feature. The random forest regressor is implemented with max depth of 5 and random state of 0 (mostly due to limited computational power of the laptop used). The model is then used to make a prediction through the predict() function with a list of predictors. As this article is focused mainly as a proof of concept, the data given is a mix of real-world data selected from a date outside of the timeframe and realistic generated data meant to test the model's accuracy. To test the accuracy of the test, we decided to use K-fold cross validation at 5 folds instead of the default accuracy_score since the acuracy_score is more beneficial for analyzing classifiers rather than the regressive models used in the experiment. All predictions and cross validation scores are stored in a txt file to be accessed later for data analysis.



```
def linear_regression(test, experiment_number):
    print("Running Test 1A Linear Regression...")
    model = linear_model.LinearRegression()
    new_input_data, new_output_data = organize_dates(experiment_number, input_data, output_data)
    model.fit(new_input_data, new_output_data)
    # y_pred = model.predict(input_data)
    # accuracy_score(output_data, y_pred)
    print("Linear Prediction", model.predict([test]))
    score = cross_val_score(model, input_data, output_data, cv=5)
    score = score.mean()*100
    f = open("test_1A_results.txt", "a")
    print("Test 1A:\nLinear Regression\n Cross validation score =" + str(score))
    f.write("Test 1A:\nLinear Regression\n Cross validation score =" + str(score))
    f.close()
```

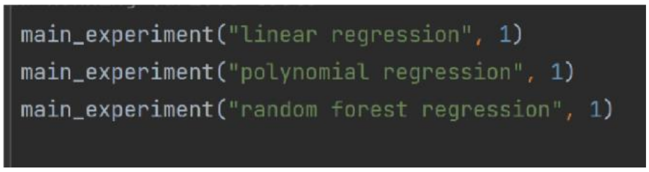
Figure 5. Function for linear regression

Finally, Figure 6 shows the main function that ties all the functions together and makes the experiment easier to debug and implement. The experimental process involves calling the main_experiment function and inputting the model type (model selection) and the experiment number (time frame selection) and results will be printed to the screen and recorded in a txt file. An example of this is shown in Figure 7.



```
def main_experiment(model_type, experiment_number):
    model_type = model_type.lower()
    test_data = le.fit_transform(prediction_test)
    print(test_data)
    if model_type == "linear regression":
        linear_regression_1a(test_data, experiment_number)
    elif model_type == "polynomial regression":
        polynomial_regression_1a(test_data, experiment_number)
    elif model_type == "random forest regression":
        random_forest_regression_1a(test_data, experiment_number)
    else:
        print("Not a valid model")
```

Figure 6. Main function that ties all other functions together



```
main_experiment("linear regression", 1)
main_experiment("polynomial regression", 1)
main_experiment("random forest regression", 1)
```

Figure 7. Results (txt file)

4. Experiment

The first part of the experiment aimed to test the feature significance of various predictors using feature significance. The model is used to perform predictions of daily COVID-19 cases based on real and artificially made data. Then, the feature significance of each element within the list of values used to make the prediction is calculated by a feature within random forest regression and given a value of 0 through 1. A value of 0 indicates that the predictor's correlation with the trend is statistically insignificant compared to the other predictors and a value of 1 indicates that the predictor can be used as the sole input value for the predictions. Out of all of the 66 potential predictors, eight were chosen as final contenders due to their feature significance, availability in the datafile, and similarity to output. Predictors such as number of handwashing facilities or vaccinations were not chosen despite moderate correlation due to their limited number of data points. Predictors such as new cases per million were also not used to predict new cases because they were too similar and thus would not reflect the model's ultimate goal. The eight predictors for predicting daily new cases were: date, new tests, weekly hospital admissions, population density, total tests, total deaths, location, and total cases. All these predictors, with few exceptions, are dynamic predictors, which is reasonable considering the value of the output is constantly changing.

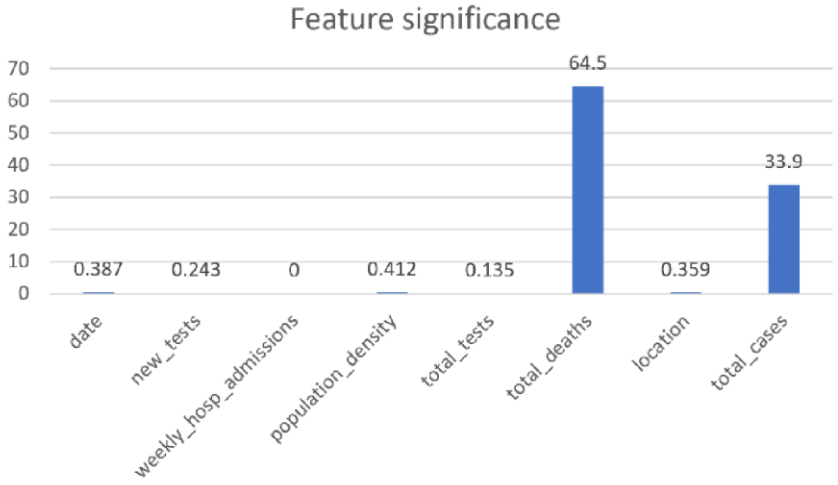


Figure 8. Mean value of feature significance after 10 tests with various realistic artificial data

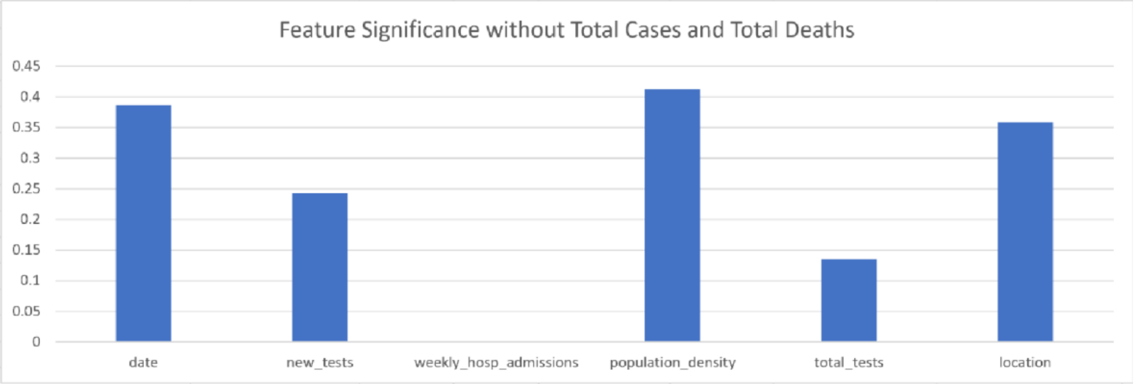
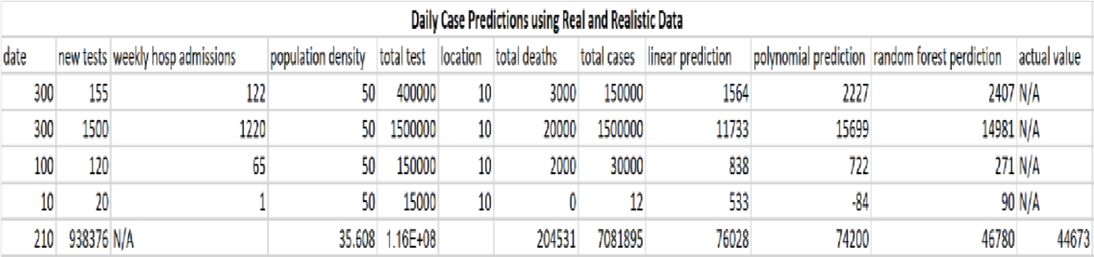


Figure 9. Mean value of feature significance without total deaths and total cases

Figure 8 shows the mean value of feature significance after 10 tests with various realistic artificial data. Figure 9 shows the mean value of feature significance without total deaths and total cases.

As seen Figures 8 and 9, the total cases and deaths combined is overwhelmingly more significant to the prediction than other factors. This is likely because predictors such as location and population density are stagnant. These predictors likely have a strong influence on the rate of increase of new cases rather than how many new cases there are. Predictors such as new tests, weekly hospitalizations, and total tests suffered due to the lack of data. Despite selecting the predictors with availability in mind, the preferred way of removing nan values removed every row within the list. Replacing every nan value with -1 dramatically harmed the accuracy. The date had a low feature importance since it increases linearly while daily new cases increases nonlinearly. The feature significance of total deaths was almost that of total cases. This was a surprise since there is often a considerable lag between cases and deaths. Our initial hypothesis was that total deaths would be a better predictor since deaths are better documented than cases and thus total deaths might be considered a better predictor than total cases for predicting adiseases' spread in a region. Further analysis is needed to provide a conclusive answer.



Daily Case Predictions using Real and Realistic Data

date	new tests	weekly hosp admissions	population density	total test	location	total deaths	total cases	linear prediction	polynomial prediction	random forest perdiction	actual value
300	155	122	50	400000	10	3000	150000	1564	2227	2407	N/A
300	1500	1220	50	1500000	10	20000	1500000	11733	15699	14981	N/A
100	120	65	50	150000	10	2000	30000	838	722	271	N/A
10	20	1	50	15000	10	0	12	533	-84	90	N/A
210	938376	N/A	35.608	1.16E+08		204531	7081895	76028	74200	46780	44673

Figure 10. Predictions

The second part of the experiment focuses on predicting daily Covid increases with the predictors shown in Figure 10. Displayed are five predictions made by each of the three models based on five sets of input, four artificially created and one taken from the USA on September 26th. It can be seen that the predictions made are mostly reasonable with random forest regression being the most accurate both logically and in terms of direct comparisons as seen with the predictions made with the fourth and fifth set of values.

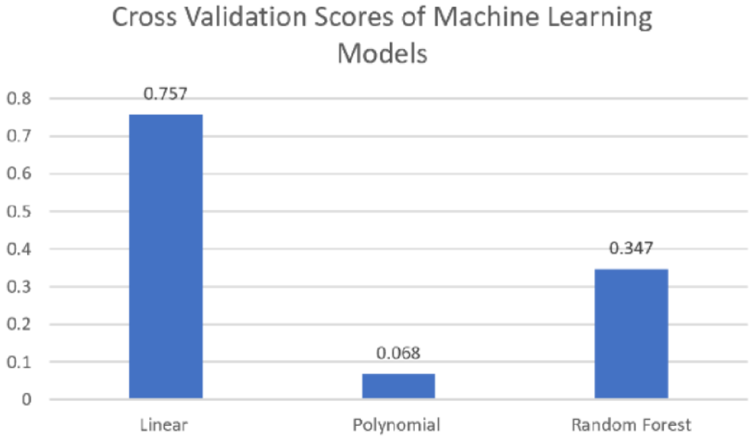


Figure 11. Results

Interestingly, K-fold cross validation values varied dramatically within the three regressive models despite little noticeable difference, especially between linear and polynomial regression, despite the two models having little appreciable difference in the predictions made. However, polynomial regression struggles to make predictions with numerically very small or very large inputs so that likely accounted for this significant difference. Additionally, the random forest regressor, despite being consistently more accurate when assessed qualitatively, received a much lower cross validation score than linear regression. This is likely due to overfitting where linear regression excels at making predictions based strictly on the training data, but is not as effective as random forest at analyzing new data outside the range of training data given to the model (See Figure 11).

The results of experiment one identified the eight most important predictors of daily increases in COVID-19 cases and the hierarchy of influences for each predictor within the selected group. Despite potential sources of error, this analysis contributes insight into the heated debate regarding the public's perception of the pandemic since it reveals that new tests and total tests do not have as significant an impact on new cases as some have suggested. This makes sense logically since the positivity rate remained relatively low on a national level for all intermediate periods of the pandemic and thus additional tests should not account for a significant influx of COVID cases. The results additionally highlight areas for future research in predicting the trends of total deaths and total cases. As they are the major predictors of daily increases in COVID-19 cases, factors that can significantly influence total deaths and total cases will likely have a significant impact on the spread of the pandemic as well. The second experiment shows the accuracy of the regressive models to make predictions on new cases of COVID when provided with both real and artificial data. Ultimately, despite certain inaccuracies and shortcomings, these experiments achieved their purpose of demonstrating the ability of machine learning models, especially the random forest regressor, to analyze and predict the trends of COVID-19, which affirms their potential in the field of epidemiology.

5. RELATED WORK

Solanki and Singh examined different machine learning models such as regularized ridge regression and deep learning (ARIMA and SARIMAX) to make predictions for COVID-19's spread in the form of cases and deaths over a set time period, which they did with commendable accuracy [15]. In particular, the study utilized a ridge regression model with given predictors such as "days since the first case," as well as "growth factors and growth ratios," which they estimated using a ridge polynomial regression to create a projection of active COVID cases in India over a seven-month period. In comparison, our experiment used regressive models to analyze a wider range of predictors such as daily available COVID-tests and hospitalizations to make next-day predictions rather than longer time periods.

Painuli, et al. developed a predictive model for individual COVID-19 cases in the major states of India using the ARIMA model and past COVID-19 infections with random forest and the extra tree classifier as well as a forecasting model of COVID-19 trends [16]. Those models were able to accomplish their respective tasks with accuracy. While our experiment is similar in our attempt to create a predictive model for COVID-19 cases, we utilized different machine learning models—polynomial regression, linear regression, and random forest regression—as opposed to univariate regression analysis models such as ARIMA. In addition, our experiment aimed to explore the differing significance of correlations between various predictors and the predicted values through random forest regression.

Watson, et al. utilized a robust combination of a Bayesian model for a location specific trajectory of COVID-19 and the random forest model for death predictions all within a compartmental model to accurately forecastCovid cases, deaths, and recoveries based on observed and projected data. This parallels our attempt to predict daily Covid cases, though our methodology differs in that we focused on purely regressive models. In addition, we utilized a greater diversity of predictors in an attempt to not only produce accurate predictions, but also to compare the feature significance of the predictors.

6. CONCLUSION AND FUTURE WORK

In this paper, we explored a method of predicting daily COVID-19 cases with three regressive machine learning models—linear regression, polynomial regression, and random forest regression. These models were trained using a diverse variety of predictors ranging from total cases of COVID-19 and date to demographic data like median age and population density. The feature significance was calculated through random forest regression to compare the influence of the various predictors on the prediction being made. In the experiments, random forest regression and polynomial regression performed much better than linear regression, as confirmed by both comparisons with real world data and cross validations scores. This is reasonable because the trends are not linear in nature and an overall decreasing trend or increasing trend would lead the linear regression model to predict a number that is either far too small or far too big depending on the selected time frame. Ultimately, the purpose of the models is not to pinpoint accurate forecasts of a specific region, but rather to identify overarching, pandemic-related trends on a national level, estimate the daily increases in COVID-19 cases to a reasonable degree, and recognize the significant influencers of these trends. Though lacking specificity and precision, holistic views of the pandemic are arguably as important as precise community-specific forecasts, since they provide general insight into the national-level factors influencing the disease's spread through a diverse area. The handling of a pandemic requires effective local- and national-level responses, but knowledge of holistic trends may benefit decision making to help manage future, national-level pandemics.

REFERENCES

[1] Bernard Stoecklin, Sibylle, et al. "First cases of coronavirus disease 2019 (COVID-19) in France: surveillance, investigations and control measures, January 2020." Euro surveillance : bulletin Europeen sur les maladies transmissibles = European communicable disease bulletin vol. 25,6 (2020): 2000094.

[2] "China Coronavirus: Lockdown Measures Rise across Hubei Province." *BBC News*, BBC, 23 Jan. 2020, www.bbc.com/news/world-asia-china-51217455.

[3] Mahase, Elisabeth. "Covid-19: WHO Declares Pandemic Because of 'Alarming Levels' of Spread, Severity, and Inaction." *BMJ*, 2020, p. m1036.

[4] Jia, L., Li, K., Jiang, Y., Guo, X.: Prediction and analysis of coronavirus disease 2019. arXiv preprint arXiv:2003.05447 (2020)

[5] "Transmission of SARS-CoV-2: Implications for Infection Prevention Precautions." *World Health Organization,* World Health Organization, 9 July 2020, www.who.int/news-room/commentaries/detail/transmission-of-sars-cov-2-implications-for-infect ion-prevention-precautions.

[6] "Coronavirus Cases:" *Worldometer*, www.worldometers.info/coronavirus/.

[7] Kulick, Debbie. "COVID Has Had an Impact on Emergency Medical Service Providers: Something to Think About." *Pocono Record,* Pocono Record, 28 Apr. 2021, www.poconorecord.com/story/lifestyle/columns/2021/04/28/debbie-kulick-covid-has-had-impact -emergency-medical-services/4860410001/.

[8] Rubin, Rita. "COVID-19's Crushing Effects on Medical Practices, Some of Which Might Not Survive." *JAMA*, vol. 324, no. 4, 2020, p. 321.

[9] "COVID-19 and the Least Developed Countries | Department of Economic and Social Affairs." *United Nations,* United Nations, 1 May 2020,

www.un.org/development/desa/dpad/publication/un-desa-policy-brief-66-covid-19-and-the-leastdeveloped-countries/.

[10] Munywoki, Gilbert. "Economic Effects of Novel Coronavirus (COVID – 19) on the Global Economy." *SSRN Electronic Journal*, 29 Oct. 2020.

[11] Boettke, Peter J., and Benjamin Powell. "The Political Economy of the COVID-19 Pandemic." *SSRN Electronic Journal*, 12 Feb. 2021

[12] Solanki, Arun, and Tarana Singh. "COVID-19 Epidemic Analysis and Prediction Using Machine Learning Algorithms." Emerging Technologies for Battling Covid-19: Applications and Innovations 16 Feb. 2021, vol. 324 57–78.

[13] Painuli, Deepak et al. "Forecast and prediction of COVID-19 using machine learning." Data Science for COVID-19 (2021): 381–397.

[14] Watson, Gregory L., et al. "Pandemic Velocity: Forecasting COVID-19 in the US with a Machine Learning & Bayesian Time Series Compartmental Model." PLOS Computational Biology, vol. 17, no. 3, 2021.

[15] Solanki, Arun, and Tarana Singh. "COVID-19 Epidemic Analysis and Prediction Using Machine Learning Algorithms." *Emerging Technologies for Battling Covid-19: Applications and Innovations* vol. 324 57–78. 16 Feb. 2021.

[16] Painuli, Deepak et al. "Forecast and prediction of COVID-19 using machine learning." *Data Science for COVID-19* (2021): 381–397.

[17] Watson, Gregory L., et al. "Pandemic Velocity: Forecasting COVID-19 in the US with a Machine Learning & Bayesian Time Series Compartmental Model." *PLOS Computational Biology*, vol. 17, no. 3, 2021

AIRBORNE SOFTWARE DEVELOPMENT PROCESSES CERTIFICATION REVIEW STRATEGY BASED ON RTCA/DO-178C

Jinghua Sun[1], Samuel Edwards[2], Nic Connelly[3],
Andrew Bridge[4] and Lei Zhang[1]

[1]COMAC Shanghai Aircraft Design and Research Institute, Shanghai, China
[2]Defence Aviation Safety Authority, 661 Bourke St, Melbourne, VIC, Australia
[3]School of Engineering, RMIT University, Melbourne, VIC, Australia
[4]European Union Aviation Safety Agency, Cologne, Germany

ABSTRACT

Airborne software is invisible and intangible, and is frequently used to provide safety critical functionality for aircraft. Highly complex software, however, cannot be exhaustively tested and only assured through a structured, process, activity, and objective-based approach. This paper studied the development processes and objectives applicable to different software levels based on RTCA/DO-178C, and identified 82 technical focus points based on each airborne software development sub-process, then created a Process Technology Coverage matrix to demonstrate the technical focuses of each process. This paper proposes an objective-oriented top-down and bottom-up sampling strategy for the four software Stage of Involvement reviews by considering the frequency and depth of involvement. Finally, this paper provides a Technology Objective Coverage matrix, which can support the reviewers to perform the efficient risk-based SOI reviews by considering the identified technical points, thus efficiently achieving confidence in the level of safety of the aircraft from the software assurance perspective.

KEYWORDS

Airborne Software, Stage of Involvement, DO-178C, Safety Critical Software Oversight.

1. INTRODUCTION

Modern transport aircraft are developed and certified with numerous, complex systems that rely on embedded software to control and optimise the flight of the aircraft. With the development of computer technology, more and more aircraft system functions are implemented by airborne software, however, software is an intangible asset, having no physical presence, which is stored on various media (CASA, 2014). The software will fail only when there is a latent defect, virus, design error, or single event exception. Software design errors may exist for many years without manifesting or causing malfunctions. Thus quality should be built into the software and be reviewed by assuring the development and verification processes (CASA, 2014) (Rierson, 2013). Airborne software is always one of the critical concerns in the aircraft certification process (EASA, 2012).

Software safety is an increasingly prominent issue in today's aviation industry (Mendis, 2008). The aircraft systems can directly affect the safety of aircraft, however, the software is fundamentally different from the physical components installed on the aircraft. The structural

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 269-282, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112321

components of the aircraft can be tested to ensure that there are no design and manufacturing defects, whereas the Mean Time between Failures (MTBF) and programmed replacements do not apply to software components (CASA, 2014). Continuous testing cannot demonstrate that software has a reliability level similar to that of physical components, as the software does not degrade with use, rather, defects are experienced in exact states of operation. The software embedded in physical systems directly impacts the safety of the aircraft and its occupants (Hilderman & Baghai, 2007). Employing software review technology can ensure that rigour has been applied during the applicant's design commensurate with the worst-case failure condition associated with airborne software (RTCA, 2011a).

A level of assurance is required to have confidence in software to ensure aircraft safety. In October 2018 and March 2019, two Boeing 737MAX planes belonging to Indonesian Lion Air and Ethiopian Airlines crashed, respectively, causing a total of 346 deaths, which was directly related to the design of the Manoeuvring Characteristics Augmentation System (MCAS) and its flight control law software (COMMITTEE, 2020). This tragedy is a stark reminder of the criticality of software, and has been a significant loss for Boeing, and the operators of aircraft that were grounded. At the same time, the FAA as the supervisor also triggered a crisis of public trust. Wayne Rash stated that "As is the case where software controls hardware, there are ways things can go wrong either because something happened that was not anticipated, or because the response was wrong" (Rash, 2019). So what can be done to ensure that the software to be maintained at an acceptable level of safety?

Due to the particularity of airborne software and the professionalism of software-related technologies, significant pressure is placed on airborne software reviewers. However, the complexity and scale of software keeps increasing as modern civil aircraft are getting more and more integrated and complex. Therefore, formulating a set of airborne software review strategies with related technical focuses is an important issue.

For more than three decades, airborne software has been developed and assured through a structured approach based on objectives and activities (Rierson, 2013). The most commonly used method to measure software goodness is DO-178[], which is recognised as Means of Compliance (MOC) by NAAs(National Airworthiness Authority) via their respective Advisory Circular (AC) (Hilderman & Baghai, 2007). This study was conducted based on DO-178C to establish the airborne software review strategy to support certification of safety critical software.

2. ANALYSIS OF SOFTWARE REVIEW TECHNICAL FOCUSES

2.1. Quantitative Analysis of DO-178C Software Life Cycle Process and Objectives

DO-178C is a process-based, activity-driven, objective-oriented standard. It is not a software development standard, but a method to measure the goodness of software, and provide a safety benchmark that is commensurate with the safety criticality. It contains six processes (represented in Figure 5), which are the planning process, development process, and four integral processes (verification process, configuration management process, quality assurance process, and certification liaison process). The integral processes are supported throughout the whole software lifecycle (RTCA, 2011a). Notably, not all the projects follow a Waterfall lifecycle model, but a variation in the representation of the waterfall model instead (Santos and Ferreira 2019). The DO-178C proposed process, output, and input are represented in Figure 1, and can be adapted to the project lifecycle model as required.

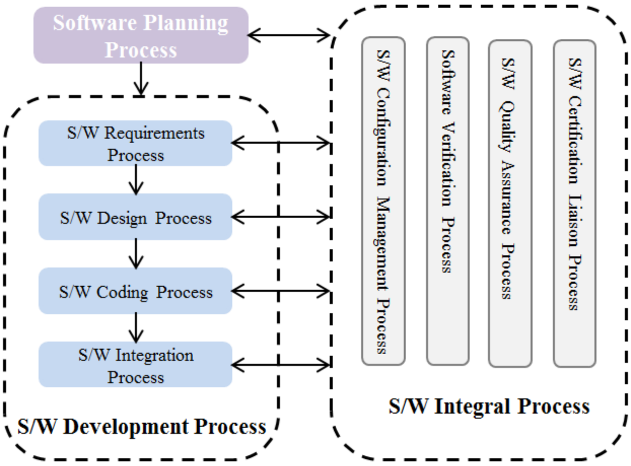

Software Planning Process
S/W Requirements Process
S/W Design Process
S/W Coding Process
S/W Integration Process
S/W Development Process
S/W Configuration Management Process
Software Verification Process
S/W Quality Assurance Process
S/W Certification Liaison Process
S/W Integral Process

Figure 1. DO-178C software life cycle processes

A latent software error in data or the final product can cause a fault of the software, then the abnormal behaviours of software can lead to a system failure condition, which can finally affect the aircraft operations. The rigour of software development is determined by the software level. DO-178C defined five software levels as listed in Table 1. DAL A is the severest, while DAL E has no safety impact. The software DAL is determined by the system safety assessment process. The different level has different objectives requirements. Table 2 *and Figure 2* are the comparison of DO-178C's Objectives in Annex A from Table A-1 to Table A-10 for different DALs of software.

Table 1. DO-178C Software DAL, related failure conditions and objectives.

Source: (Marques & Yelisetty , 2019) (Jimenez el. 2020)

System Failure Condition	Required Software Level	Number of Associated Objectives	Number of Associated Objectives with Independence
Catastrophic	A	71	31
Hazardous	B	69	19
Major	C	62	5
Minor	D	26	2
No Safety Effect	E	0	0

Table 2. Comparison of DO-178C objectives for different software levels.

Annex A	**A**	**B**	**C**	**D**
Table A-1 Software Planning Process	7	7	7	2
Table A-2 Software Development Process	7	7	7	4
Table A-3 Verification of Outputs of Software Requirements Process	7	7	6	3
Table A-4 Verification of Outputs of the Software Design Process	13	13	9	1
Table A-5 Verification of Outputs of Software Coding & Integration Processes	9	9	8	1
Table A-6 Testing of Outputs of Software Integration Process	5	5	5	3

Table A-7 Verification of Verification Process Results	9	7	6	1
Table A-8 Software Configuration Management Process	6	6	6	6
Table A-9 Software Quality Assurance Process	5	5	5	2
Table A-10 Certification Liaison Process	3	3	3	3

The experience accumulation of reviewers can start from Level D software review, and gradually master the review methods and techniques of higher-level software, to finally be competent for the review of Level A software:

a) Level D can be treated as a black box, focusing on high-level requirements development and verification. If updating a level D software to level C, there will be a leap of workload by 36 objectives.

b) The objectives differences between level C and level B include 1 Objective in Table A-3 “High-level requirements are compatible with target computer”, 4 objectives in Table A-4 about the compatibility and verifiability of low-level requirements and architecture, 1 objective in Table A-5 “Source code is verifiable”, and 1 objective about decision coverage in Table A-7.

c) The main differences between A and B are 2 objectives in Table A-7, which are requirements of MCDC Structural Coverage Analysis (SCA) and verification of additional code that cannot be traced to source code.

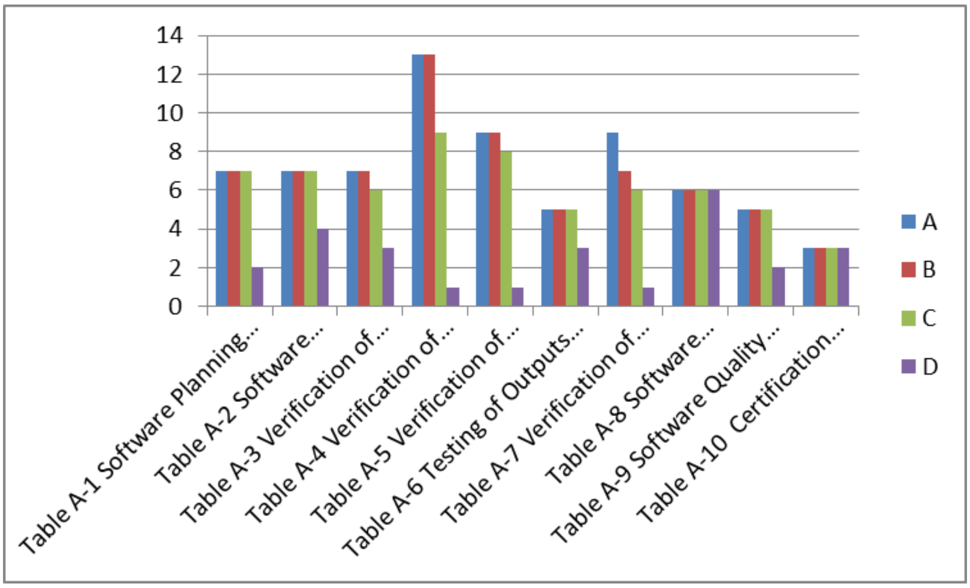


Figure 2. The comparison of applicable objectives in each Table of Annex A for different software levels

2.2. Analysis of Technical Focuses of DO-178C Process

The study on the DO-178C objectives and process can help software reviewers quickly locate the technical focuses and finding compliance. Table 3 is the analysis of the technologies based on DO-178C software life cycle processes. Each process of DO-178C may contain sub-process and components (RTCA, 2011a). The technical focus points are analysed based on each component covered and concerned during the software reviews. The technologies are from the research and analysis of the technical focus points, with most of them are described in DO-178C, and a few are from the industry practice, then they are compared with the CAST Paper research themes, finally re-analysed to ensure the completeness of the technology list.

Table 3. Qualitative analysis of technical focus points and related techniques of DO-178C

Source: (RTCA, 2011a) (FAA, 2003) (EASA, 2012) (FAA, 2004) (FAA, 2017) (CAST, 2002) (RTCA, 2011b)

Process	**Sub-process/Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
4.0 Software Planning	4.3 Software Plans	11.1 PSAC 11.2 SDP 11.3 SVP 11.4 SCMP 11.5 SQAP	1) Software DAL Determination 2) Partitioning 3) Multiple-Version Dissimilar Software 4) Safety Monitoring 5) PDI 6) User-Modifiable Software 7) COTS 8) Field-Loadable Software 9) Option-Selectable Software 10) Software Life Cycle Definition 11) Transition Criteria 12) Deactivated Code 13) PDS 14) Tool Qualification 15) Reuse of tool qualification data 16) Reuse of software life cycle data 17) Exhaustive Input Testing 18) Software Reliability Model 19) Product Service History 20) Database/PDI 21) Use of COTS Graphical Processor Unit (GPU) 22) Microprocessor 23) Multiple Core Processors 24) SEU (Single Event Upset) 25) Reverse engineering
	4.4 Software Life cycle Environment Planning	4.4.1 Software Development Environment 4.4.2 Language and Complier 4.4.3 Software Test Environment	
	4.5 Software Development Standards	11.6 Software Requirements Standards 11.7 Software Design Standards 11.8 Software Code standards	
5.0 Software Development	5.1 Software Requirements	11.9 Software Requirements Data	26) High-Level Requirements

Process	**Sub-process/Components**	**Technical Focus Points/Elements**	**Technologies ($T_{i,}$ i=1…n)**
		11.22 Parameter Data Item File	27) Derived requirements 28) Merging high-level requirements and low-level requirements
	5.2 Software Design	11.10 Design Description	29) Control Flow Design 30) Data Flow Design 31) Low-Level Requirements 32) PDI Design
	5.3 Software Coding	11.11 Source Code 11.22 Parameter Data Item File	33) C, Ada, Assembly languages 34) Auto code generation 35) MBD 36) OOT 37) Cache 38) Stack
	5.4 Integration	11.12 Executable Object Code	39) Compiling 40) Complier library 41) Software Integrity Check (e.g. Cyclic redundancy check, Checksum)
	5.5 Traceability	11.21 Trace Data	42) Traceability Tools (e.g. DOORS)
6.0 Software Verification	6.3 Software review and analysis	Review and analysis of Software Plans and standards 6.3.1 Review and analysis of Software High-Level Requirements (HLRs) 6.3.2 Review and analysis of Software Low-Level Requirements (LLRs) 6.3.3 Review and analysis of Software Architecture 6.3.4 Review and analysis of Source Code 6.3.5 Review and analysis of the Outputs of the Integration Process 6.4.5 Review and analysis of Test Cases, procedures, and results 6.6 Review and analysis of PDI File	43) Plans and Standards Review 44) HLR Review and Analysis 45) LLR Review and Analysis 46) Architecture Review and Analysis 47) Source Code Review and Analysis 48) Outputs of the Integration Process Review and Analysis 49) Test Cases Review and Analysis 50) PDI file Review and Analysis 51) Worst-Case Execution Time 52) Verification of Stack Usage 53) Model Review and Analysis 54) Verification of independence
	6.4 Software Testing	6.4.1 Test Environment 6.4.2,6.2.3	55) Hardware/Software Integration Testing 56) Software Integration

Process	**Sub-process/Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
		Requirements-Based Test 6.4.4 Test coverage Analysis	Testing 57) Low-Level Testing 58) Normal Range Test Cases Selection 59) Robustness Test Cases Selection 60) MCDC 61) Decision Coverage Analysis 62) Statement Coverage Analysis 63) Data Coupling 64) Control Coupling 65) DAL A additional verification (Whether Object Code can directly traceable to source code) 66) Extraneous Code Resolution 67) Deactivated Code Handle
	6.5 Traceability	11.21 Trace Data	
Integral Process	7.0 Software Configuration Management	7.2.1 Configuration Identification	68) Software part numbering
		7.2.2 Baselines and Traceability	69) Baseline Definition
		7.2.3 Problem Reporting	70) OPR Category Definition
		7.2.4 Change Control	71) Software Change Control
		7.2.5 Change Review	
		7.2.6 Configuration Status Accounting	
		7.2.7 Archive, Retrieval, and Release	72) Media Selection, Refreshing, Duplication 73) Data Retention
		7.3 Data Control Category	
		7.4 Software Load Control	74) Software Conformity Inspection
		7.5 Software Life Cycle Environment Control	
	8.0 Software Quality Assurance	8.2 Software Quality Assurance Activities	
		8.3 Software Conformity Review (SCR)	75) SCR 76) First Article Inspection (FAI)
	9.0 Certification Liaison	9.1 Means of Compliance and Planning (LOI, Milestones, and Issue Papers, etc.)	77) LOI Criteria
		9.2 SOI Reviews	78) SOI Review Strategy 79) Sampling Strategy
		9.3 Software	80) Software maturity

Process	**Sub-process/Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
		Approval, including approval of Software Configuration Index (SCI) and Software Accomplishment Summary (SAS)	evaluation for Type Inspection Authorization (TIA) 81) Open Problem Report (OPR) Evaluation 82) Software Change Impact Analysis (CIA) to determine Major or Minor Changes

Note: In addition to the description of the items in the first three columns, the chapter number of the referenced DO-178C is also listed, such as 6.0 Software Verification, where 6.0 refers to DO-178C Chapter 6. The verification process is one of the four integral processes listed separately in the table because it is highly related to the software product. Each technology is identified as T_i. For instance, T_{81} refers to item 81) ORP technology in this table.

This paper identified a total of 82 technologies based on the DO-178C software assurance benchmark. The same technology may be used in different processes, but the focus will be on different perspectives. For example, MBD (Model Based Development) may be used in planning, design, coding, and verification processes. The technology distribution statistics in each process are shown in Table 4 and Figure 3.

Table 4. Software Process Technology Coverage (PTC) matrix

DO-178C Process	**Technology Coverage**	**Amount**
Planning Process	T_1 ~T_{25}	25
Development Process	T_{26} ~T_{42}	17
Verification Process	T_{43}~ T_{67}	25
Configuration Process	T_{68}~ T_{74}	7
Quality Assurance Process	T_{75}~ T_{76}	2
Certification Liaison Process	T_{77} ~T_{82}	6

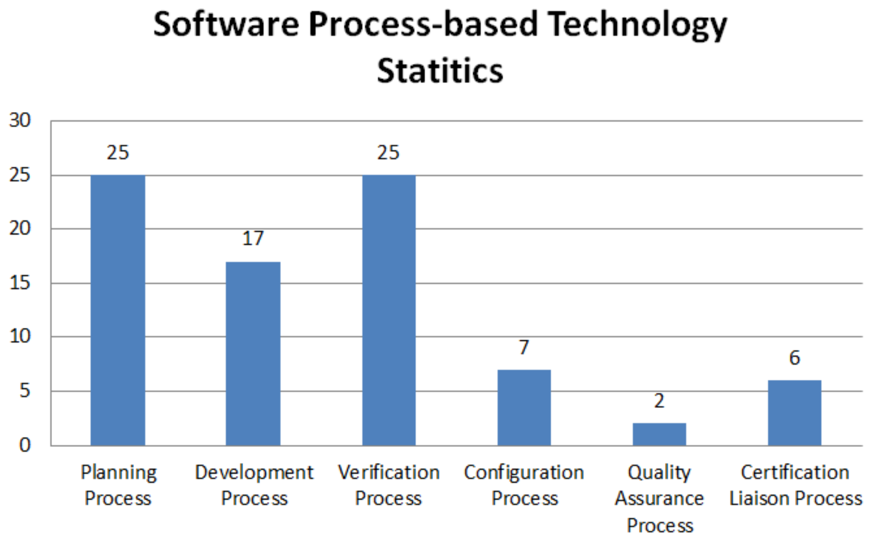


Figure 3. A quantitative analysis of the technology distribution of each process

3. ANALYSIS OF SOFTWARE REVIEWS AND SAMPLING STRATEGY

3.1. Analysis of SOI Reviews and LOI

SOI is a method of implementing process control on airborne software originally defined by the FAA in order to monitor the software life cycle process and assess compliance with the applicable objectives of DO-178B and related airworthiness requirements. According to FAA and EASA policy, four SOI reviews are defined, which are SOI#1 Software Planning Review, SOI#2 Software Development Review, SOI#3 Software Verification Review, and SOI#4 Final Certification Software Review (FAA, 2003) (EASA, 2012). The review aims to find systemic problems in the applicant's software developing processes and non-compliance issues with regulations and establish confidence in the software through the reviews (FAA, 2004). The purpose of software SOI review is to develop confidence in compliance with DO-178C objectives and other applicable software policy, guidance, and issue papers (FAA, 2018). The reviews can be conducted by a certification officer or delegated to a DER or ODA/DOA. The LOI depends on the project-specific conditions, which is executed through SOI. The main factors that can affect the SOI frequency are as follows:

a) the software category, which means PDS, COTS, new-developed software, TSOA software, libraries, RTOS, IMA hosted software, etc.,
b) the software DALs as determined by the system safety assessment process (EASA, 2012) (FAA, 2003),
c) the project characteristics, such as the tier of supplier-chain, the experience of the applicant, the complexity of the project, system functionality and novelty, software developing team human resources, and existence of issues associated with Section 12 of DO-178C (FAA, 2003),
d) the use of new technologies or unusual design features (EASA, 2012),
e) whether using alternative methods to show compliance,
f) the establishment and operation of the software assurance aspect of the applicant’s Design Assurance System (DAS), and
g) the amount of planning review activities of the delegation systems (e.g. DER or ODA) and the applicant’s self-monitoring status (EASA, 2012).

3.2. Analysis of SOI Review and Sampling Strategy

Studies indicate that developing a scientific and reasonable software review and sampling strategy, and mastering the technology related to each SOI review, especially the impact of this technology on software compliance verification, will facilitate the rapid identification of key clues during software reviews (Dodd & Habli, 2012). Each SOI review and sampling strategy and the applicable identified technologies for each SOI are analysed in the following sections.

3.2.1. SOI#1: Software Planning Review

The goal of SOI#1is to evaluate the compliance of the software planning with the applicable objectives of Table A-1 and A-8~A-10 of DO-178C Annex A (FAA, 2004). Review activities and review strategy of SOI#1 is suggested to firstly review the software interface with system development process, hardware design process, and system safety assessment process to assess the consistency among the plans and standards in compliance with the objectives of DO-178C Table A-1 (Chen, et al., 2015). Then review the verification results, the Software Quality Assurance (SQA) record, the Software Configuration Management (SCM) records, and the certification liaison process, and assess the compliance with the applicable objectives of DO-

178C Table A-8~A-10 (FAA, 2004). The important thing is to assess the consistency between software plans to determine that when the applicant follows their plans whether they will meet all applicable objectives of DO-178C and other applicable software policy or guidance (RTCA, 2011a). If tool qualification, MBD, OOT, or formal method is applied, the assessment could also cover additional aspects in RTCA/DO-330, DO-331,DO-332 and DO-333 (FAA, 2017).

3.2.2. SOI#2: Software Development Review.

The goal of SOI#2 is to assess whether the software plans and standards are effectively implemented and to evaluate the compliance of the software development process to the applicable objectives of DO-178C Table A-2~A-5, and A-8~A-10 (FAA, 2003). The review and sampling strategy is suggested to review the output of the software requirements process, design process, coding process, and integration process, and assess the compliance with applicable objectives of DO-178C Table A-2~A-5 through top-down and bottom-up thread review (illustrated in Figure 4) with the Risk-Based sampling strategy*(VanderLeest, 2013)*, by which the sampling covers each functional area until the reviewer has sufficient confidence in the software implementation of specific functional requirements set. It also need to assess the compliance of configuration management, quality assurance, and airworthiness liaison process with the applicable objectives of DO-178C A-8~A-10, and evaluate the closure status of review action items in SOI#1.

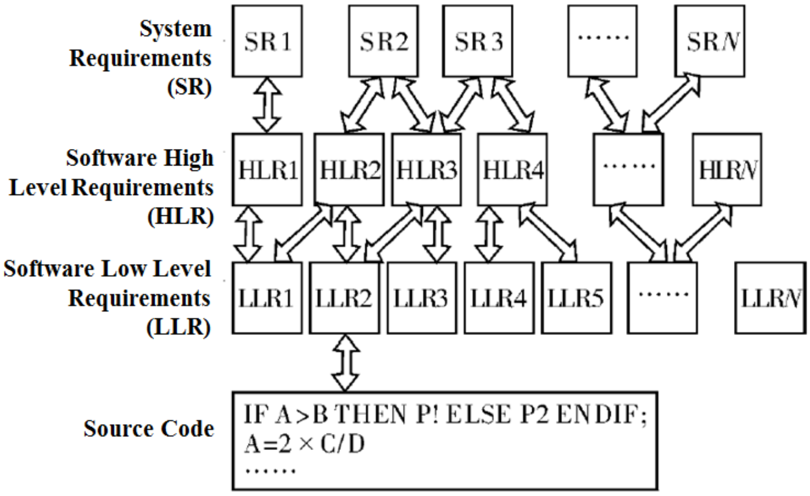


Figure 4. Illustration of top-down and bottom-up thread review

Source: (Xing & Mu, 2015)

3.2.3. SOI#3: Software Verification Review

The purpose of SOI#3 is to evaluate the compliance of the software verification process with the applicable objectives of DO-178C Table A-6, A-7, and A-8~A-10 to assess the effectiveness and implementation of verification plans and procedures (FAA, 2003). The SOI#3 software review and sampling strategy are suggested to be also risk-based to perform a delta review of the development data if there are major changes from the previous review, and to assess the test cases, test procedures, verification results, test coverage, and code structure coverage to the applicable objectives of DO-178C Table A-6 and A-7. The sampling strategy is the same as SOI#2.

Besides, it is recommended that reviewers pick some requirements that need additional considerations, for instance, if there are option selectable software or UMS, the verification of the protection mechanism should be reviewed. There are many cases, for example, the Fuel Quantity Indication Computer (FQIC) can use Litre or Kilogram as the unit of measurement, which is mainly configured through the pins of the computer. Some options are implemented by software design, for example, the B737Max MCAS system has two modes which are Auto Pilot (AP) and manual flight modes (Boeing, 2020), the mode selection is usually implemented by software logic with Case or If statement, according to such strategy, the reviewers is likely to pay attention to the protection mechanism to ensure there are no unintended behaviours and the system requirements and architecture has defined the mechanism (COMMITTEE, 2020).

3.2.4. SOI#4: Final Certification Software Review

The goal of SOI#4 is to determine compliance of the final software product with the appropriate objectives of RTCA/DO-178C and other applicable certification policies and guidance (FAA, 2003). The SOI#4 review strategy is suggested to evaluate the closure status of findings, observations, and action items of the previous reviews, to conduct a delta review of SOI#2 and SOI#3 when necessary if there are major changes or the reviewer does have sufficient confidence in the software product, to assess the OPRs(Open Problem Reports) to judge whether they can be deferred to post-TC, and to review the final SCI, SAS, tool qualification data, such as Tool Accomplishment Summary (TAS) if applicable, to judge whether the version of software product intended to be used in the certified system or equipment fully comply with all applicable DO-178C objectives, the policy, and guidance (FAA, 2004).

3.3. Quantitative Analysis of SOI Technology & Objective Coverage

Through the above analysis, it can be known that airborne software safety assurance can be achieved by a structured approach. Table 5 is the analysis result of the applicable technology and objectives of each SOI. The analysis approach and process are as follows:

a) Based on the analysis of the SOI review strategy in Section 4.3.2 of this paper, identify the appropriate technologies associated with each SOI by referring to the technology list in Table 3.
b) Based on DO-178C Annex A and the analysis of SOI review strategy in Section 4.3.2 of this paper, in conjunction with FAA Order 8110.49 Chapter 2 “Software Review Process” (FAA 2003), which was based on DO-178B, analyse the available data to identify applicable objectives for each SOI based on DO-178C.

Figure 5 is the quantitative analysis of the distribution of TOC of each SOI review, which demonstrated that 50% of the DO-178C objectives are assessed in SOI#2 review, with 35% of the technologies assessed. According to the number of objectives, SOI#3 is the second highest, with 32% of the objectives addressed, however, accounting for 26% of the technologies. SOI#1 accounts for 31% of the objectives, but covers 16%, of the technologies. Finally, SOI#4 objectives are at 2%, and technology accounts for 8%. SOI#4 is a review of the entire life cycle process. It is necessary to evaluate all previous SOI review opening items, non-conformance items, and observation items. Therefore, although the SOI#4 objectives are accounted for the least, it plays a very critical role in the entire software review process, as the reviewers will determine whether the software is in compliance with all the applicable objectives of DO-178C and whether it can obtain the final approval.

Table 5. TOC Matrix of each SOI.

Source: (FAA, 2004) (RTCA, 2011a)

SOI	**Technology**		**Objectives**	
	Identification	**Amount**	**Identification**	**Amount**
SOI#1	T_{1}~T_{25}, T_{43}, T_{68} ~ T_{69}, T_{77}~T_{78}	30	Table A-1: Objective1-7(All Objectives) Table A-8: Objective1-4 Table A-9: Objective 1 Table A-10: Objective1-2	14
SOI#2	T_{26} ~T_{42} T_{44}~ T_{54} T_{68}~T_{71} T_{78}~T_{79}	34	Table A-2: Objective 1-6 Table A-3: Objective1-7(All Objectives) Table A-4: Objective 1-13(All Objectives) Table A-5: Objective 1-6 Table A-8: Objective 1-4,6 Table A-9: Objective 1-4 Table A-10: Objective 1-2	43
SOI#3	T_{48}~T_{49}, T_{51}, T_{54} T_{55} ~ T_{67} T_{68}~ T_{71} T_{77}~T_{81}	26	Table A-5: Objective 7-9 Table A-6: Objective1-5(All Objectives) Table A-7: Objective 1-9(All Objectives) Table A-8: Objective1-6(All Objectives) Table A-9: Objective1-4 Table A-10: Objective 1-2	28
SOI#4	T_{75} ~ T_{82}	8	Table A-9: Objective 5 Table A-10: Objective 3	2

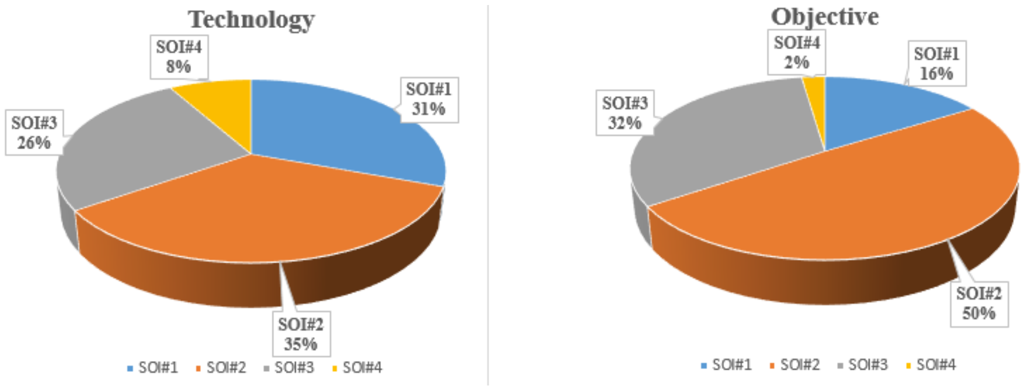


Figure 5. The TOC distribution of each SOI

4. Conclusions

Software reviews are always treated as a critical part of the system certification process, provided that it is conducted following each NAA's procedures and handbooks to finding compliance with the safety-related regulations § 25.1301 and § 25.1309. This research analysed regulation requirements and software review policies of the FAA, EASA, CASA, and CAAC using a comparative approach to establish the software certification basis and means of compliance. Given that the airborne software review is performed by people, the different working experiences, backgrounds, and technical capabilities of the reviewers may lead to different review conclusions.

An in-depth software review can discover the shortfalls existing in the design and potential risks to safety of aircraft design. This paper studied the technical focuses of airborne software review based on the DO-178C software life cycle process and identified 82 technology aspects through analysis of objectives and activities of each process. This paper also analysed the LOI impact factors of airborne software SOI review, and developed a set of Risk-based SOI reviews and sampling strategies, taking into account the applicable identified technologies and compliance objectives of DO-178C by developing the PTC and TOC matrixes. The study of this paper will help NAAs to maintain software expertise and formulate more effective software review procedures and guidance documents, and carry out corresponding technical research to ensure aircraft safety by conducting in-depth software reviews from a software certification perspective.

In the research process of this project, it was found that an Objective-oriented SOI review method based on DO-178C is meaningful. Software reviewers are required to apply their expertise and experience to judge compliance, while the software developer can provide effective assistance to demonstrate compliant evidence and perform software verification activities. This paper identified the necessity of future study to explore the applicable technical focuses and SOI review strategies for different DALs of airborne software based on each objective of DO-178C.

REFERENCES

[1] Boeing, 2020. *737Max Software Update.* [Online]
Available at: https://www.boeing.com/commercial/737max/737-max-software-updates.page
[Accessed 25 10 2020].

[2] CASA, 2014. *AC 21-50: Approval of software and electronic hardware parts,* Canberra: CASA.

[3] CAST, 2002. *CAST-10 "Literal" Interpretation of Decision Coverage Increases Rigor of Testing Requirements.* [Online]
Available at: https://www.rapitasystems.com/blog/cast-10-literal-interpretation-decision-coverage-increases-rigor-testing-requirements
[Accessed 6 July 2020].

[4] Chen, Y., Yan, L. & Sun, J., 2015. *Civil Aircraft Airborne Software Management.* Beijing: The Aviation Industry Press of China.

[5] COMMITTEE, T. H., 2020. *FINAL COMMITTEE REPORT: THE DESIGN, DEVELOPMENT & CERTIFICATION OF THE BOEING 737 MAX,* USA: THE House COMMITTEE on TRANSPORTATION AND INFRASTRUCTURE.

[6] Dodd, I. & Habli, I., 2012. Safety certification of airborne software: An empirical study. *Reliability Engineering & System Safety,* 98(1), pp. 7-23.

[7] EASA, 2012. *EASA CM – SWCEH – 002 Issue: 01 Revision: 01 Software Aspects of Certification.* [Online]
Available at: https://www.easa.europa.eu/sites/default/files/dfu/certification-docs-certification-memorandum-EASA-CM-SWCEH-002-Issue-01-Rev-01-Software-Aspects-of-Certification.pdf
[Accessed 19 Oct. 2020].

[8] FAA, 2003. *FAA Order8110.49: SOFTWARE APPROVAL GUIDELINES.* [Online]
Available at: https://www.faa.gov/documentLibrary/media/Order/FAA_Order_8110.49.pdf
[Accessed 19 Oct. 2020].

[9] FAA, 2004. *Job Aid: Conducting Software Reviews Prior to Certification.* [Online]
Available at: https://elsmar.com/elsmarqualityforum/attachments/jobaid-r1-1-pdf.14401/
[Accessed 19 Oct. 2020].

[10] FAA, 2011. *Order8110.49 Chg1: Software Approval Guidelines,* Washington: FAA.

[11] FAA, 2017. *AC 20-115D: Airborne Software Development Assurance Using EUROCAE ED-12( ) and RTCA DO-178( ),* Washington: FAA.

[12] FAA, 2018. *Order8110.49 A: Software Approval Guidelines,* Washington: FAA.

[13] Jimenez, J. A. et al., 2020. A Framework for Evaluating the Standards for the Production of Airborne and Ground Traffic Management Software. *IEEE Access,* 8(1), pp. 142-161.

[14] LU, Y. et al., 2011. Coverage analysis of airborne software testing based on DO178B standard. *Procedia Engineering,* I(17), pp. 480-488.

[15] Marques, J. & Yelisetty, S., 2019. AN ANALYSIS OF SOFTWARE REQUIREMENTS

SPECIFICATION CHARACTERISTICS IN REGULATED ENVIRONMENTS. *International Journal of Software Engineering & Applications (IJSEA),* 10(6), pp. 1-15.

[16] Mendis, K. S., 2008. Software Safety and Its Relation to Software Quality Assurance. In: G. G. Schulmeyer, ed. *Handbook of Software Quality Assurance.* Boston: ATECH HOUSE, p. 211.

[17] Rash, W., 2019. *eWEEK.* [Online]
Available at: https://www.eweek.com/mobile/how-software-can-make-an-airplane-crash
[Accessed 30 July 2020].

[18] Rierson, . L., 2013. *Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance.* 1 ed. Boca Raton: CRC Press.

[19] RTCA, 2011a. *DO-178C: Software Considerations in Airborne Systems and Equipment Certification,* Washington: RTCA, Inc.

[20] RTCA, 2011b. *DO-248C: Supporting Information for DO-178C and DO-278A,* Washington: RTCA, Inc.

[21] Xing, L. & Mu, M., 2015. Research On Airworthiness Standard DO-178B / C′s Object Analysis and Stage of Involvement Review in Airborne Software. *Aeronautical Computing Technique,* 45(5), pp. 97-101.

AUTHORS

Jinghua Sun : senior engineer, mainly study on airborne software, electronic hardware and system engineering areas. CAAC DER.

Samuel Edwards : DASA software specialist, mainly study on airborne software safety assurance and reviews.

Nic Connelly : RMIT Senior Lecturer, has more than 30 years' experience in the aviation industry, having ever worked for Air services and Virgin Australia.

Andrew Bridge : EASA software, electronic hardware and safety expert.

Lei Zhang : senior engineer, mainly study on process control and quality management.

Checklist Usage in Secure Software Development

Zhongwei Teng, Jacob Tate, William Nock, Carlos Olea, Jules White

Vanderbilt University

Abstract

Checklists have been used to increase safety in aviation and help prevent mistakes in surgeries. However, despite the success of checklists in many domains, checklists have not been universally successful in improving safety. A large volume of checklists is being published online for helping software developers produce more secure code and avoid mistakes that lead to cyber-security vulnerabilities. It is not clear if these secure development checklists are an effective method of teaching developers to avoid cyber-security mistakes and reducing coding errors that introduce vulnerabilities. This paper presents in-process research looking at the secure coding checklists available online, how they map to well-known checklist formats investigated in prior human factors research, and unique pitfalls that some secure development checklists exhibit related to decidability, abstraction, and reuse.

Keywords

Checklists, Cyber Security, Software Development

1. Introduction

Checklists have become common in industries, such as aviation, where human errors can lead to significant safety issues. For example, NASA has published detailed design guidance on creating and using checklists for aircraft operation [1]. The World Health Organization (WHO) has begun encouraging hospitals to adopt surgical-safety checklists [2], based on the results of a research study conducted by surgical staff in 8 hospitals from 2007 to 2008 which showed checklists reduced complications and mortality rates [3]. By decomposing a complex system or workflow into a set of deterministic items, checklists can help to avoid common errors and free up mental capacity for important cognitive tasks.

The promising achievements of checklists in other fields have motivated software developers to publish their own checklists for secure software development, which can be easily found online. A Google search for ”secure software development checklist” produces thousands of results. These checklists typically consist of items listing things that software developers should do to produce more secure software.

However, the security community needs to determine whether the benefits that checklists show in other domains carry over to secure software development, and if so, in what scenarios [4]. Moreover, the overall quality of the checklists that software engineers and their managers find online has not been assessed. Because of the sense of ”completeness” that checking off items on a list can instill, it is critical that checklists be of high quality. For example, a mismatch between a checklist's target domain and an individual application's domain can make developers assume that they have ”checked off all the critical security items,” when in fact, they are ignoring critical risks in their codebase that emerge from unique aspects of their application domain. Checklists have a connotation of ”complete-

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 283-293, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112322

Format	{Topic Keyword(s)} {Secondary Keyword(s)} security checklists		
Topic Keyword	**Secondary Keyword(s)**	**#Unique Checklists**	**#Tags in StackOverflow**

Web Development	—	9	492('web-applications'+'security')
			708('web'+'security')

Python	Flask	5	248('flask-security')
			79('flask'+'security')

Python	Django	7	367('django'+'security')
Python	Secure Coding	4	972('python'+'security')
Java	Spring	4	21773('spring-security')

Java	Android	4	430('android-security')
			1084('android'+'security')

Java	Secure Coding	6	311('java-security')
			5572('java'+'security')

Table 1: Search Terms for the Secure Coding Checklist Survey

ness" as shown in the Merriam Webster Dictionary definition of checklists:

```
a list of things to be checked or done
a pilot's checklist before takeoff
also : a comprehensive list
```

Online resources, such as StackOverflow, have become a commonly used resource for software developers to learn and gather information about best practices. Especially for small development teams that may not have substantial support from cybersecurity groups within their organization, checklists that developers or their managers find online may create a false sense of security. For example, when a checklist reminds web developers to avoid SQL injection attacks with a few examples, such as "Use prepared statements.", it is not guaranteed that readers of the checklist will fully understand the concept of prepared statements or how similar issues may arise when using object relational mapping systems. In this case, even though developers can show that their application passes a cybersecurity checklist, their underlying code may yet be highly insecure. Further, to outside non-experts, the knowledge that a checklist was followed (e.g., a secure process) may generate a more powerful belief in the security of the project than other artifacts.

In this paper, we explore whether or not checklists are truly effective in helping educate developers on how to produce more secure software. We explore the challenges of adopting security checklists in secure software development. In particular, we show that the scope of tasks where security checklists are effective in software development is much narrower than expected. This narrow scope is a result of significant differences in common tasks in software development and the types of tasks and contexts where checklists are known to be efficacious.

Paper organization. The remainder of this paper is organized as follows: Section introduces the survey that we conducted of online checklists. Section presents four key issues that we found when assessing a set of online secure development checklists. Section discusses the results of the survey and highlights key issues that were commonly found. Section introduces related work on checklists in other industries. Section provides concluding remarks and future work.

2. Survey of Cybersecurity Checklists found Online

Software developers looking to increase the security of their applications will often turn to the web for available resources and risks of seeking advice from web search are revealed by a survey [5]. Checklists can exist on programming-specific sites, blogs, or other sites. If a programmer is inspired to use a checklist to audit their work, these are the resources they will find. On a pragmatic level, this is the current state of cybersecurity checklists.

This survey provides initial data on the prevalence and potential failings of checklists related to secure coding and software implementation that a developer may find online. This analysis seeks to explore the extent to which checklists found online accommodate, or fall victim to, the proposed theoretical pitfalls described later in Section . The data should be considered a work in progress.

Using a variety of search keywords, which developers may use to address security concerns, secure software development checklists were located using the Google search engine. Searches were conducted using the incognito mode on Google Chrome, so that past searches did not influence results. The search queries for the survey were generated by concatenating a **Topic Keyword** & **Secondary Keywords**, with the term ”Security Checklists”. For example, as shown in Table 1, ”Python Django Security Checklists” was one such query. For each query, only links that were on the first page (excluding advertisements) were evaluated, since it is estimated that 95% of search engine visitors click through to a link on the first page of Google search results [6].

Based on choices of programming applications and trends on Stack Overflow, we chose two topic keywords, ”Python” and ”Java”, as well as corresponding programming frameworks as secondary keywords. The number of unique results, excluding ads, is shown in Table 1.

As shown, we analyzed 39 secure coding checklists in total. The application domain which we focused on was web application development. Selections of central keywords and auxiliary keywords are based on statistics of StackOverflow Trends. The checklists covered general security topics (web development security), two languages, including Python and Java, and four frameworks, such as DJango and Spring. General secure coding checklists for each topic keyword were also evaluated (e.g. ”Java Security Checklist”).

3. Challenges of Translating Checklists to Secure Software Development

It’s an open question whether the benefits seen in other domains, such as from the WHO’s surgery checklist, can be realized in cybersecurity to better educate software engineers. Despite this open question, there are a wide variety of cybersecurity checklists for software engineers available.

The most significant work on human factors considerations when designing checklists has come from the aviation and space domains [1, 7]. In general, checklist items are expected to fall into either a ”read-do” format [7] or a ”read-respond” (also known as ”challenge-response”) [1] format. Read-do items dictate actions that should be taken by the reader immediately after reading the item (e.g., circle incision site). Read-respond items are designed to be confirmed after reading (e.g., Is the incision site circled?).

We conducted research to understand the characteristics of the cybersecurity checklists for software engineers and understand how their items and scope compare to the basic best practices listed in prior research [1, 7]. Throughout the remainder of this section, we describe five key qualitative issues that many checklists suffer from and make creating clean ”read-do” or ”read-respond” items challenging in the secure coding domain. Read-do and read-respond items are the fundamental expected structure of checklist items. As described in human factors research on checklists in other domains, secure coding tasks that can't fit into a read-do or read-respond format aren't suited for a checklist. Secure software engineering checklists composed of items that fail to fit into the read-do or read-respond format should be approached with caution.

Based on observations of checklists from the survey conducted in Section , we identified repeated anti-patterns / challenges in secure coding checklists. Although they may appear sound, checklists that exhibit these patterns fundamentally violate past human factors research in best practices for creating usable checklists [1, 7]. We will discuss 4 main challenges for secure software development checklists throughout the remainder of this section.

3.1. Challenge: Non-deterministic Read-respond Items

Read-respond items in a checklist are meant to be deterministic in nature. Different users reading a "read-respond" item should be expected to always arrive at the same answer in the same situation. Read-respond items are constructed such that diverse users of the checklist interpret the item in the same manner, precisely and efficiently. For example, an aircraft maintenance checklist item that requires verifying that "the XYZ electrical lead wire does not exceed 30 metres in length", can be answered as "yes" or "no" in a consistent fashion. In contrast, if a checklist contains ambiguous items, such as "no electrical leads are too long", it's not precise enough in definition to be deterministically evaluated. One user may know the maximum length to be 30 meters, while another may be familiar with a different standard.

In many circumstances, however, security checklists in software development contain non-deterministic items, often statements warning developers to avoid a specific type of vulnerability, included with limited examples. For example, "Sensitive transactions require re authentication" [8] is a non-deterministic read-respond item. What is a "sensitive transaction"? Alternatively, the read-respond item "Session cookies are encrypted and have a length of at least 128 bits, are complex", provides concrete deterministic guidance on the encryption and length, but provides non-deterministic guidance on "complex". Two developers may arrive at different conclusions on cookie complexity.

When secure coding checklist items are non-deterministic, a number of problems arise. First, an item may be checked off simply because the developer doesn't really understand the original intent of the question. Second, different developers on the same team may apply varying security standards to their source code analysis / production, leading to inconsistent security. Finally, since checklists instill a sense of security / completeness, checking off items that can't consistently be answered / applied is potentially dangerous.

3.2. Challenge: Undecidable Read-respond Items

The Halting Problem and Rice's Theorem [9, 10] are well known theoretical proofs that inform the limits of what we can know about an arbitrary software application. Rice's Theorem shows that we cannot decide most semantic properties of arbitrary software systems, such as "their security." A fundamental problem in many secure coding checklists is that they use read-respond items that would violate Rice's Theorem or the Halting Problem if they could be answered for arbitrary software applications.

For example, a web development security checklist item warns developers "Do not leak session IDs" [11]. Given an arbitrary web application that uses session IDs, this question is likely undecidable. A similar read-respond item would be "do not create inputs that prevent the application from halting", which is likely not helpful to a software engineer.

Although these types of items appear as helpful reminders to developers, they are unhelpful when delivered in a checklist format. A web development security guide that explains what leaking a session ID is and provides educational examples is a better format for delivering this material. However, this type of material does not fit into a read-respond or read-do checklist item. Unfortunately, declarative statements that express what they want accomplished, but not how, tend to be undecidable in security checklists. To fit into a *decidable* checklist item, the specific steps on how to do what is stated need to be expressed and each step decidable in nature. Checklist items that state "don't introduce security vulnerability X" are generally undecidable due to Rice's Theorem.

3.3. Challenge: Failed Generalization for Reuse in Read-respond Questions

Checklists are built on the idea of summarizing only essential critical activities or information into a short and actionable list to make sure that people consistently and correctly perform these tasks. However, if we transfer an effective checklist from one context to a new context, due to differences in architecture, software stacks, etc., there is a significant risk of introducing non-determinism or irrelevant items. For example, checking for double freeing of memory may make sense in one programming language, but not make sense in another language, leading to the introduction of unnecessary items that distract the user.

In software development, however, it is uncommon to create a checklist for exactly one product – checklists are almost always abstracted and generalized like software to facilitate reuse. Checklists are usually designed for a domain, a framework, or a programming language, such as mobile application/Android/Java. These checklists are then implemented by developers appropriately within the new product's software architecture. The issue then arises: generalizing the use of these domain- or architecture-specific checklists relies on the assumption that their content will be transferable to other software domains.

There is a tension in developing an item for a cybersecurity checklist between the reusability of the checklist and the specificity of the items. Ideally, a checklist can be carefully crafted and reused, just like we do with software components. This desire for reuse leads checklist authors into the trap of creating non-deterministic read-respond items. The greater the number of non-deterministic read-respond items, the more likely that the checklist will be used incorrectly **AND** confer incorrect trust in the security of the system.

For example, the checklist item "Do not use eval() and similar functions" [11], is somewhat deterministic / decidable in the Javascript context, where eval() is a built-in function that source code can be checked for. However, the "similar functions" language is trying to facilitate use of this item in other contexts, possibly outside of Javascript – simply map eval() to the equivalent function in your context – which is not clearly deterministic or decidable. The goal of making the item reusable outside of Javascript with "similar functions" language creates non-determinism. Contrast this item with the checklist item "If I handle XML files, I disable entity and DTD processing" [12]. This checklist item provides more detail on what needs to be accomplished and but is still not necessarily decidable for an arbitrary language and set of frameworks.

3.4. Challenge: Missing Description of the Expected Checklist Context

Compared to daily-use checklists, such as shopping or errand lists, checklists for most non-trivial domains have explicit linkages between the items and domain concepts, such

Topic Keywords	Web development	Python	Java
%Determinism Issues	88.9	75	57.1
%Undecidable	100	87.5	57.1
%Improper Generalization	33.3	100	35.7
% Lacking Context	55.6	50	42.9

Table 2: Results from Secure Coding Checklist Survey

as aircraft wing flaps, ailerons, etc. Moreover, the checklists implicitly expect users to understand key concepts in the application domain, such as Australian aircraft maintenance workers being expected to understand the Australian Standard 2865[13]. In aviation, training programs are necessary for practitioners who use checklists to reduce risk and ensure that pilots know how to deterministically and correctly respond to or do what is asked. For example, pilots are trained on checklists specific to the aircraft they fly in order to ensure they understand how the checklist applies to their aircraft.

Many checklists in the secure coding domain, however, do not have a rigorously defined context where they are applicable nor an explicit set of concepts that the user should be familiar with. Without an explicit context in which the checklist is applicable, similar to a software design pattern's list of context, forces, etc; software engineers may incorrectly assume that the checklist is designed for their application domain / constraints. Developers that use Google to find and apply security checklists to their codebases, without sufficient knowledge of the assumptions made when the checklist was designed, will inevitably produce flawed evaluations of their software's security. When *completed*, these improper evaluations cause developers to boast a false sense of security to the desired users of their products.

In many other domains, particularly aircraft, the exact context that the checklist was designed for is well-understood. Pilots of a Boeing 777 don't have to wonder if the checklist they are using was actually designed for an Airbus a380. Not only is the domain of the checklist's applicability well-understood, but the training requirements to effectively use the checklist are as well. In contrast, there are often few expectations listed in the surrounding documentation of secure coding checklists to guide developers in determining if the checklist is applicable to their context. Was the checklist only designed for web applications built in Javascript with Express and NodeJS or is it equally applicable for .Net or PHP applications? We found many checklists fail to document this information in the materials accompanying the secure coding checklist.

4. Survey Results

Table 2 shows the results from our survey of 39 checklists from the Internet. Results in our survey **DO NOT represent human performance** of using the surveyed checklists. Our survey, however, identified the prevalence of the challenges from Section in the checklists we surveyed. For each checklist, if at least one item had demonstrated the challenge, it was added to the count of checklists demonstrating that issue. For example, if at least one item on the checklist was non-deterministic, the checklist was added to the count of checklists suffering from ”determinism issues.” The reason we allowed a single issue to add an item to the count is that security is inherently a weakest link game. We provide detailed per-checklist results in Tables II-V. The list of checklists and URLs is provided in the appendix.

[0] https://github.com/zhongwei-teng/security-checklists

Keywords	Determinism Issues	Undecidable	Improper Generalization	Lacking Context	Checklist
Python Django Security Checklist	2	1	1	2	C1
	2	1	1	2	C2
	1	1	1	0	C3
	2	2	1	0	C4
	1	1	1	0	C5
	2	2	1	2	C6
	0	0	0	0	C7
Python Flask Security Checklist	1	1	1	2	C8
	0	0	0	0	C9
	1	1	1	2	C10
	1	1	1	2	C11
	1	1	1	0	C12
Python Secure Coding Checklist	1	1	1	2	C13
	0	0	0	2	C14
	0	0	1	0	C15
	1	1	1	0	C16

Table 3: Secure Coding Checklist Results for Python

Keywords	Determinism Issues	Undecidable	Improper Generalization	Lacking Context	Checklist
Web Development Security Checklist	1	1	2	0	C17
	1	1	2	2	C18
	1	1	2	0	C19
	1	1	2	2	C20
	1	1	1	2	C21
	0	1	1	0	C22
	2	1	2	0	C23
	1	1	1	2	C24
	1	1	2	1	C25

Table 4: Secure Coding Checklist Results for Web Development

A single flaw is all that is needed – or a single misunderstanding of what a checklist item is asking for. If a checklist item is created, it should be deterministic, decidable, have proper generality, and an appropriate context. Otherwise, although it may communicate important information, it should be moved out of a checklist format and into a ”guide” or ”tutorial” where appropriate explanation can be given and there is no connotation of ”security completeness” due to the delivery format.

We observed some interesting variation in the results based on the programming language. Checklists for Java web applications exhibited the lowest percentage (25%) of determinism issues, that only 2 of 8 checklists have part of determinism issues. Their lower non-determinism was primarily attributable to the fact that the majority of the Java checklists in the top ten Google search results specifically targeted the Spring Framework and gave concrete guidance on configuration of Spring beans and the Spring Security Framework that could easily be checked. Secure coding checklist results found with the Spring keyword overall struck a fine balance between specificity, breadth, and relevance. Checklist items were confined to addressable, framework-specific programming prescriptions, and gave

Keywords	Determinism Issues	Undecidable	Improper Generalization	Lacking Context	Checklist
Java Security Checklist	1	1	2	0	C26
	1	1	2	0	C27
	2	1	2	2	C28
	2	2	2	2	C29
	2	1	2	2	C30
	2	2	2	2	C31
Java Spring Security	2	2	2	2	C32
	2	2	2	2	C33
Checklist Java Android Security Checklist	0	0	0	0	C34
	0	0	1	2	C35
	1	0	0	0	C36
	1	2	1	2	C37
	1	2	2	0	C38
	0	0	0	0	C39

Table 5: Secure Coding Checklist Results for Java

extensive context as to when the checklist is pertinent. Additionally, these checklists either provided proper code examples, or proposed existing third-party tools for identifying common mistakes. For example, in the blog [14] from top search results of "Java Spring Security Checklist", each checking item, which is usually a general security suggestion, are elaborated in the context of Java Spring with detailed code snippet, so that readers can easily check if their projects fit this secure coding rule. Besides regular security suggestions, they also recommend users to use OWASP ZAP [15], a security tool, to perform penetration testing in their Spring application.

Results for Android secure coding checklists, overall, scored most poorly out of all Java checklists, that 4 of 6 checklists have bad performance in our evaluation metrics. Though nearly every checklist highlighted the best-known Android vulnerabilities, most lacked any additional context. Furthermore, several checklists provided many non-deterministic items for developers to act on. For instance, a checklist mentions that developers should adopt Firewall to secure the server and API, while it doesn't either explain assessment of secure server or provide an external link as a suggestion. Checklists which were specific to Android often warned developers of broad classes of vulnerabilities to check for, absent of a decidable action for developers to determine if the vulnerability was actually present in their codebase.

Django, which is a Python web framework, provides an official security checklist and an automated checker to apply to your codebase – demonstrating the decidability and determinism of their items. Each item has a concrete instruction to check it. For example, instead of only emphasizing keeping secret keys properly, it provides two ways to keep secret keys(from environment variables or files) [16]. Further, it's notable that in the beginning of the official Django checklist, a link to key information that users need to know in order to understand the related security risks was provided. On the other hand, we found numerous personal blogs that provided Django security checklists that were not as carefully crafted and suffered from frequent determinism, generalization, and decidability issues. For instance, a checklist [17] mentions that outside data is not reliable and should be validated without providing readers with detailed examples. Users may also have trouble in deciding if all outside data in their application has been validated.

5. Related Work

Checklists have emerged as a popular decision aid in the last decade. Popular books, such as the best selling novel *The Checklist Manifesto* [18], have promoted their benefits. However, checklists do not always provide benefits and can be potentially detrimental if they create a false sense of cybersecurity. Even in aviation, one of the most successful fields where checklists are used, significant effort is dedicated to carefully designing checklists [1] and it would be unheard of to use a casually designed checklist downloaded from a blog.

Many domains have dedicated significant research to showing the contexts in which checklists provide benefit. For example, research [19] indicates that checklists can improve the efficiency of students and teachers in the classroom in certain scenarios. Using checklists for educational purposes combines the goal of ”providing a list of critical tasks” with the goal of ”educating users on what tasks they need to know.” However, there is conflicting guidance on whether or not checklists are valuable for educational purposes. Many resources, such as the ”Checklist for Checklists” published by Project Check [20] explicitly warns against using checklists to teach: ”A checklist is NOT a teaching tool or an algorithm”.

Application of checklists in healthcare has also not been universally successful. Given the complexity of patient care, checklists were proposed to aid in suppressing the frequency of a number of potentially life-threatening medical errors. For example, in one checklist study, infections fell to zero after a doctor at John Hopkins University implemented a checklist to remind doctors of the necessary steps to prevent infection during a number of procedures. A WHO pilot project to incorporate surgical checklists proved successful in a number of pilot hospitals. However, when the project was scaled up outside of the pilot hospitals, the original benefits were not seen. After evaluating the original checklist, a new 19-item checklist [21] was developed by WHO, which has demonstrated improved performance in ensuring surgical safety [22, 23].

In contrast to many other domains, however, secure software development checklists are not well-studied. The closest research is by Bellovin et al. [4], which investigated potential risks to using arbitrary checklists to enhance software cybersecurity. Much more additional research is necessary to help developers and organizations determine when checklists are appropriate in cybersecurity and the qualities that checklists need to be successful.

The importance of assessing the quality of online materials available for secure software development has been explored in other literature. Results from a survey illustrated that informal but more accessible online resources usually have negative effects on coding security [24], showing the necessity of paying more attention to the quality of online resources with respect to security. Our research is complementary to this prior work and specifically focuses on how well checklists are crafted to convey security information.

6. Conclusion

The domains where checklists have been successful are those where the most effort is put into careful design and testing of the checklists. Aircraft manufacturers put significant effort into designing checklists for take-off and other operations and have dedicated human factors experts evaluating them. In the domain of secure coding, however, there are a lot of checklists focused on helping developers produce secure software. Some of these checklists, such as the official checklist for Python’s Django web framework, are well-designed and offer concrete guidance.

At the same time, there are a large number of checklists available that are not nearly as well crafted. These checklists suffer from significant problems in their decidability, determinism, level of abstraction, and context communication. For example, many checklists include items of the form ”ensure you are not vulnerable to X”, with no specific guidance on how to *decide* if you are vulnerable. Often, determining if a codebase is vulnerable to ”X” is fundamentally undecidable.

Our survey of checklists taught us a number of valuable lessons regarding their design and usage in the cybersecurity domain:

- **Framework-specific checklists** provided by the framework developers themselves were of much higher quality than other checklists. As opposed to many other checklists, framework developers would create extremely specific checklist items that were deterministic and decidable. We assume that framework developers aren’t tempted to generalize, since they are interested in seeing increased adoption of their framework and not competing frameworks/approaches. Thus, they tend to produce very precise and actionable checklist items.
- **Business marketing content** formed some of the poorest quality checklists in our survey. We found many company blog posts are often structured as ”secure coding checklists” but are written to be as general as possible and attract a large number of readers. The checklist items inherently suffer from non-determinism and undecidability because they are too general to be useful.
- **Aiming for reuse** may end up being an anti-pattern in cybersecurity checklist design. The best checklists are highly specialized to the context / language / framework to ensure that items are deterministic, decidable, not overly broad, and clearly communicate what is expected of how they are used. Communicating key vulnerabilities and ideas in a reusable way is important – but should be done outside of the context of a checklist. Instead of using the ”checklist” term that connotes ”completeness”, lists should be used to communicate this type of information.

7. References

[1] A. Degani and E. L. Wiener, “Human factors of flight-deck checklists: the normal checklist,” 1991.

[2] WHO, “Who surgical safety checklist,” *https://www.who.int/teams/integrated-health-services/patient-safety/research/safe-surgery/tool-and-resources*, 2020.

[3] A. B. Haynes, T. G. Weiser, W. R. Berry, S. R. Lipsitz, A.-H. S. Breizat, E. P. Dellinger, T. Herbosa, S. Joseph, P. L. Kibatala, M. C. M. Lapitan *et al.*, “A surgical safety checklist to reduce morbidity and mortality in a global population,” *New England Journal of Medicine*, vol. 360, no. 5, pp. 491–499, 2009.

[4] S. Bellovin, “Security by checklist,” *IEEE Security & Privacy*, vol. 6, no. 2, pp. 88–88, 2008.

[5] Y. Acar, C. Stransky, D. Wermke, C. Weir, M. L. Mazurek, and S. Fahl, “Developers need support, too: A survey of security advice for software developers,” in *2017 IEEE Cybersecurity Development (SecDev)*. IEEE, 2017, pp. 22–26.

[6] M. Malik, “Search engine optimization seo: Business digital marketing success,” 2018.

[7] A. Degani and E. L. Wiener, “Cockpit checklists: Concepts, design, and use,” *Human factors*, vol. 35, no. 2, pp. 345–359, 1993.

[8] S. Secured, *Secure Code Review Checklist*, December 5, 2018. [Online]. Available: https://www.softwaresecured.com/secure-code-review-checklist/

[9] H. G. Rice, “Classes of recursively enumerable sets and their decision problems,” *Transactions of the American Mathematical Society*, vol. 74, no. 2, pp. 358–366, 1953.

[10] L. Burkholder, “The halting problem,” *ACM SIGACT News*, vol. 18, no. 3, pp. 48–60, 1987.

[11] WikiBooks, *Web Application Security Guide/Checklist*, November 26, 2011. [Online]. Available: https://en.wikibooks.org/wiki/Web_Application_Security_Guide/Checklist

[12] T. Mendo, “Web application security checklist,” *https://blog.probely.com/web-application-security-checklist-ee0479bf60c6*, December 6, 2018.

[13] A. B. Licence and I. S. (ABLIS), “Australian standard as 2865-2009: Confined spaces - western australia,” *https://ablis.business.gov.au/service/wa/australian-standard-as-2865-2009-confined-spaces/29626*, 2020.

[14] M. Raible, “10 excellent ways to secure your spring boot application,” *https://developer.okta.com/blog/2018/07/30/10-ways-to-secure-spring-boot*, 2018.

[15] OWASP, “Zap,” *https://www.zaproxy.org/*.

[16] D. S. Foundation, “Django deployment checklist,” *https://docs.djangoproject.com/en/3.0/howto/deployment/checklist/*, December 6, 2018.

[17] O’Reilly, “A handy security checklist,” *https://www.oreilly.com/library/view/django-web-development/9781787121386/ch33s02.html*.

[18] A. Gawande, *Checklist manifesto, the (HB)*. Penguin Books India, 2010.

[19] K. D. Rowlands, “Check it out! using checklists to support student learning,” *English Journal*, pp. 61–66, 2007.

[20] P. Check, *A Checklist for Checklists*, January 14, 2010. [Online]. Available: https://www.projectcheck.org/uploads/1/0/9/0/1090835/checklist_for_checklists_final_10.3.pdf

[21] W. H. O. (WHO), “Who surgical safety checklist,” *https://www.who.int/patientsafety/ safesurgery/checklist/en/*, 2017.

[22] T. G. Weiser, A. B. Haynes, G. Dziekan, W. R. Berry, S. R. Lipsitz, A. A. Gawande *et al.*, “Effect of a 19-item surgical safety checklist during urgent operations in a global patient population,” *Annals of surgery*, vol. 251, no. 5, pp. 976–980, 2010.

[23] A. Fudickar, K. Hörle, J. Wiltfang, and B. Bein, “The effect of the who surgical safety checklist on complication rate and communication,” *Deutsches Ärzteblatt International*, vol. 109, no. 42, p. 695, 2012.

[24] Y. Acar, M. Backes, S. Fahl, D. Kim, M. L. Mazurek, and C. Stransky, “You get where you’re looking for: The impact of information sources on code security,” in *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2016, pp. 289–305.

CRYSTAL: A PRIVACY-PRESERVING DISTRIBUTED REPUTATION MANAGEMENT

Ngoc Hong Tran[1], Tri Nguyen[2], Quoc Binh Nguyen[3],
Susanna Pirttikangas[2], M-Tahar Kechadi[4,5]

[1]Vietnamese-German University, Vietnam
ngoc.th@vgu.edu.vn
[2]Center for Ubiquitous Computing, University of Oulu, Finland
tri.nguyen@oulu.fi
susanna.pirttikangas@oulu.fi
[3]Ton Duc Thang University, Vietnam
nguyenquocbinh@tdtu.edu.vn
[4]School of Computer Science, University College Dublin, Ireland
[5]Insight Centre for Data Analytics, University College Dublin
tahar.kechadi@ucd.ie

ABSTRACT

This paper investigates the situation in which exists the unshared Internet in specific areas while users in there need instant advice from others nearby. Hence, a peer-to-peer network is necessary and established by connecting all neighbouring mobile devices so that they can exchange questions and recommendations. However, not all received recommendations are reliable as users may be unknown to each other. Therefore, the trustworthiness of advice is evaluated based on the advisor's reputation score. The reputation score is locally stored in the user's mobile device. It is not completely guaranteed that the reputation score is trustful if its owner uses it for a wrong intention. In addition, another privacy problem is about honestly auditing the reputation score on the advising user by the questioning user. Therefore, this work proposes a security model, namely Crystal, for securely managing distributed reputation scores and for preserving user privacy. Crystal ensures that the reputation score can be verified, computed and audited in a secret way. Another significant point is that the device in the peer-to-peer network has limits in physical resources such as bandwidth, power and memory. For this issue, Crystal applies lightweight Elliptic Curve Cryptographic algorithms so that Crystal consumes less the physical resources of devices. The experimental results prove that our proposed model performance is promising.

KEYWORDS

Reputation, peer to peer, privacy, security, homomorphic encryption, decentralized network.

1. INTRODUCTION

Nowadays the Internet covers almost everywhere, which eases people in communication. However, the Internet is not completely accessible freely through Wi-Fi in a number of places. Whenever someone wants to use the Internet at a certain place for free to ask for some urgent information, (s)he has to be their customer, and the Internet connection owns limits. Moreover, the Internet fee is still not trivial to everyone. The other cases in which people cannot access the Internet are many. For instance, people forget to renew the mobile Internet payment before going out, or just arrive at an airport in a new country without a new sim card, etc. Therefore, there is really a need for a peer-to-peer network so that people in the aforementioned situations can send questions and recommendations, especially for an emergency. In the meanwhile, the swift growth of mobile devices today, in both quality and quantity, requires a specific type of network for exploiting their emergence that is to comfort users by the availability of services.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 295-308, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112323

One of those networks can be listed as Near-me Area Network (NAN) by Wong [1]. In NAN, the connection among nearby wireless devices (i.e., smartphones) is deployed instead of using the Internet to connect different LANs or cellular networks. Nodes in NAN can freely exchange or provide information to each other without the need for the Internet. This type of network properly fits the aforementioned lacking Internet but needing connection cases. Thus, it is necessary to apply this decentralized network architecture to allow people in an area to communicate together without the Internet.

However, security and privacy issues raise when we apply this decentralized network to those users (or, nodes) who are strangers. Nodes concern the reliability of the ones who are communicating to them. Hence, there is a need for a tool measuring the trustworthiness of each node. For that purpose, the reputation system [2] has been a competent candidate for this issue. In particular, each user holds its own reputation value based on previous feedbacks by the others. This reputation value indicates the trust level of each node. However, due to the fear of retaliation [3], the users can avoid giving correct feedback. Thus, privacy including confidentiality of feedback [4] is an important feature of reputation systems. Additionally, the reputation systems [5, 6] considers that each node needs to trust several nodes before joining the network. Meanwhile, Petrlic et al.'s system [7] requires a powerful computation with a use of Paillier cryptosystem [8]. Hence, this idea cannot apply directly to mobile carrier networks as the public key scheme consumes much time and computing performance.

To cope with the drawbacks of the related studies, our work contributes a privacy-aware protocol, namely Crystal, with its services as in Table 1. More specifically, for Crystal, we propose a reputation system basis providing the core activities of storing, computing, evaluating reputations scores of advising users, and propose a secure model of computing and evaluating reputation scores without allowing the both questioning user and advising user to intervene the reputation grading process or to learn anything from that process. Nodes in Crystal model are mobile devices with a limit in bandwidth, power and memory. Therefore, the cryptographic algorithms that are selected for Crystal are the lightweight ones so that they cannot exceed the restricted physical resources of Crystal nodes. More specifically, Crystal model leverages homomorphic Elliptic Curve Cryptography (ECC) algorithm as a lightweight encryption to optimize computing performance in comparison with [7]. Moreover, Crystal's participants do not require the knowledge of others prior to accessing.

Table 1. Services of reputation management system and its privacy-preserving version.

Reputation Management	**Privacy Preserving**
Store reputation score	Computing the encryption - Secure arithmetic operators - Secure comparative operators
Convert reputation score to reputation level	
Compute reputation score	
Evaluate reputation score	Evaluating the encryption
Update the reputation score	Protecting the integrity

The rest of this paper is organized as follows. Section 2 is about current privacy-preserving reputation managements in decentralized networks. A proposed reputation management protocol is described in Section 3. The privacy issues and the privacy-preserving protocol is presented in Sec 4. The experiment is described in Section 5. Eventually, Section 6 concludes the work.

2. Related works

A study related to decentralized privacy-preserving reputation was proposed by Hasan et al. [5, 6] through the utilization of set technologies including set-membership, plain-text equality, non-interactive zero-knowledge proof and additive homomorphic cryptosystem. Since the use of a sub-set of each member in the network for communications, the authors mentioned the fewer number of messages transmitted in the network than a previous study by Gudes et al. [9]; however, in [5, 6] each node needs to trust in a small set of participants before proving them about its correct shares as feedback based on zero-knowledge proof. Note that the correct shares from the node are not directly sent to its trusted nodes, they are forwarded by several middle nodes before obtaining the trusted nodes instead. Once receiving the messages, those middle nodes collect and add that information to the knowledge. As a consequence, those reputation systems focus on confidentiality-preserving more than privacy where the list of users joining the rating is not hidden.

Another study in [16] deploys data anonymization techniques to conceal the identity of rating users. Another work in [17], the authors used a pseudonym-based scheme to protect the user's reputation that is simply the number of cryptographic "votes". Although anonymization and pseudonymity are still effective in specific applications, however, in our work, the protocol is more complicated to use those techniques. Moreover, recently some works [18, 19] deploying the blockchain to protect the user identity from their used web services. Some other works in [20-22] create a tamper-proof ledger that is delivered among all participants. This technique can the data recorded in the ledger safe from illegally changing by the other unauthorized users and is applied in a diversity of contexts as in voting system [24], machine-to-machine environment [23], and identifier systems [25]. Another work that is the closest to our work is [7]. Petrlic et al. [7] showed a study on the privacy-preserving reputation management. Both reliability and privacy protection are based on cryptographic algorithms Paillier cryptosystem [8], a public-key encryption scheme, and the zero-knowledge proof [10]. However, this approach allows a reputation provider (RP) who manages providers' reputation value rated by users. In detail, when a user desires to rate any service provider, she/he is to be allowed by the RP. To rate service providers, users utilize a public key scheme to sign and prove with RP that the message is correct exploiting the zero-knowledge proof. Using Paillier scheme leads to a requirement of time and powerful computation. Therefore, in our work, as we address the resource-related issues in Section 1, we apply the lightweight ECC algorithm to protect the data and make the cost low to satisfy the restriction of the node's physical resources.

3. Crystal - Reputation Management Model

Let us consider a specific scenario that includes many users in a close area. Each user keeps a mobile device installed with *Crystal application* denoted as *app*. It is assumed that each user has a different experience about places surrounding their position. For example, each user knows different money exchange agencies with the best price. As a user sends a request for advice through the app, we call this user *requester*. If the other peers receive the request and know the answer, they will make a recommendation to the requester. The answering user is called *advisor*. Actually, in a popular communication way, Crystal users can communicate with each other using the Internet. However, in the specific case when the area is uncovered with the Internet, each node[1] (i.e., mobile device) needs Bluetooth or direct Wi-Fi. When all mobile devices are connected, they create a peer-to-peer (P2P) Crystal network (see Figure 1). The nodes contact together inside their physical range. For example, Bluetooth V2.1 has a 10 - 100 meter physical range on theory, whereas direct Wi-Fi has a 45-200 meter physical range [12]. When a peer enters the physical range of another peer's, they both can send messages to each

[1]The three terms node, user and mobile device are used interchangeably. This work focuses on the application layer, not on the network layer.

other. For example, in Figure 1, there are three overlapping P2P networks. Network 1 (i.e., green dashed circle) involves $\{u_0, u_1, u_2\}$ while Network 2 (i.e., yellow dashed circle) involves $\{u_1, u_5\}$, and and Network 3 (i.e., purple dashed circle) involves $\{u_3, u_4, u_5\}$. Node u_1 is in the network 1 and 2. Node u_5 is in networks 2 and 3. When a node moves out of one network, it can enter the other networks. Each user has his/her own memory which summarizes the most frequently used data, the remaining data is saved in the cloud storage.

Moreover, not all Crystal users know each other. Although a requester receives advice from their peers, the advice is not trustworthy enough for the requester until it is evaluated. However, the advice itself cannot contain adequate proof of the honesty of its owner. Therefore, the advisor has to give his/her requester the evidence to prove that their recommendation is trustworthy. For that purpose, in a P2P network, a user can ask the neighbours for the reputation score of the advisor. However, there is a case that the neighbours do not have any information of that advisor, or in case they compromise to cheat on the requester.

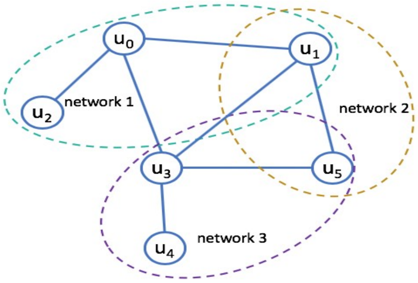


Figure 1: P2P Communication among Nodes in Crystal

In this work, each user keeps his reputation score and provides this score to his requester. This method can violate the reputation score if the owner is not honest. More details of how to protect the reputation score's integrity are discussed in Section 4. Then, the requester can evaluate advisor's reputation level. More specifically, the way to calculate a reputation score is defined as follows. Note that in this paper, we denote the requester as v and the advisor as u.

Definition 1 (Reputation Score). *Let u be an advising user who receives the grades from its requesters. Let N be the number of users attending to grade the reputation of u. Let* P_i*, with* $i = 0, \cdots, (N-1)$ *and* $P_i \in [0, 100]$*, be the mark which u is graded by user i. The reputation score of u denoted as* R_u *that is computed as follows:*

$$R_u = \frac{\Sigma_{i=0}^{(N-1)} P_i}{N}$$

Example 1. Consider the case of $N = 5$ users grading user u sequentially with {50, 60, 30, 70, 20}, respectively. Therefore, $R_u = \frac{(50+60+30+70+20)}{5} = \frac{230}{5} = 46$.

Commonly, the reputation score that is used for evaluation contains only positive scores made by the others. Hence, it is not fair enough as aforementioned. A reputation evaluation is fairer when each user does not only keep the positive scores (i.e., $R_{V_{3pos} \rightarrow u}$) made by his past requesters, but also the negative scores (i.e., $R_{V_{2neg} \rightarrow u}$) they made for him. The reputation grades, which the user u made for his past advisors, are also needed for computing the reputation level, i.e., $R_{u \rightarrow neg\, V_0}$ and $R_{u \rightarrow pos\, V_1}$. Notice that a requester cannot give positive and negative grades at the same time for the same advisor after using his advice. Remarkably, whenever the users connect to the Internet, for user u to get an honesty evaluation for itself from the server, these values $R_{u \to neg\, V_0}$ and $R_{u \to pos\, V_1}$ are requested. They are saved in the local memory when *u* grades his advisor. Therefore, in this work, we propose that a user always keeps a tuple of four reputation scores as in Definition 1.

Definition 2 (Reputation Score Tuple). *Let u be the advisor. Let V_0, V_1 be two sets of u's past advisors. Let V_2, V_3 be two sets of u's past requesters. Let RT_u be a quadruple of reputation scores of user u in Crystal model. More formally,*

$$RT_u = [R_{u \to negV_0} | R_{u \to posV_1} | R_{V_2 neg \to u} | R_{V_3 pos \to u}]$$

where $R_{u \to neg\, V_0}$ is the total negative reputation score which u grades his past advisors v, $R_{u \to pos\, V_1}$ is the final positive reputation score which u reduces the scores of the others v, $R_{V_{2neg} \to u}$ is u's reputation score which u reduces the scores of the others v, $R_{V_{3pos} \to u}$ is u's reputation score which u reduces the scores of the others v. Moreover, each element in RT_u is defined in Definition 1.

Example 2 Let users u_0 be an advisor. Let V_0 = {u_1, u_2, u_3} be the set of users whom u_0 negatively evaluates, V_1 = {u_3, u_4} be the set of users whom u_0 positively evaluates, V_2 = {u_3, u_5} be the set of users negatively evaluating u_0, and V_3 = {u_1, u_2, u_4} be the set of users positively evaluating u_0. Assume that we have $R_{u_0 \to neg\, V_0} = 60$, $R_{u_0 \to pos\, V_1} = 40$, $R_{V_{2neg} \to u_0} = 40$, and $R_{V_{3pos} \to u_0} = 60$. Thus, as in Definition 2, we have a reputation score tuple RT_u with RT_u = [60, 40, 40, 60].

When *v* receives the reputation score tuple from *u*, *v* retrieves the reputation level of *u* before following *u*'s advice. The reputation level is defined as follows.

Definition 3 (Reputation Level). *Let u be the advisor. Let R_u be the reputation score of user u as in Definition 1. Let L_u be the reputation level of u. The reputation level of u is classified in different ranges of scores as below.*

$$f_L(R_u) = \begin{cases} Low & if\ \ L_1 \le R_u < L_2 \\ Average & if\ \ L_2 \le R_u \le H_1 \\ High & if\ \ H_1 < R_u \le H_2 \end{cases}$$

where L_1, L_2 are the minimum and maximum values of the level Low, H_1 and H_2 are the minimum and maximum values of the level High.

Figure 2 illustrates Definition 2. L_1, L_2, H_1, H_2 can be customized depending on each specific system.

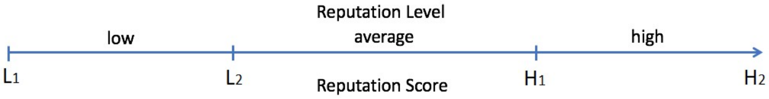
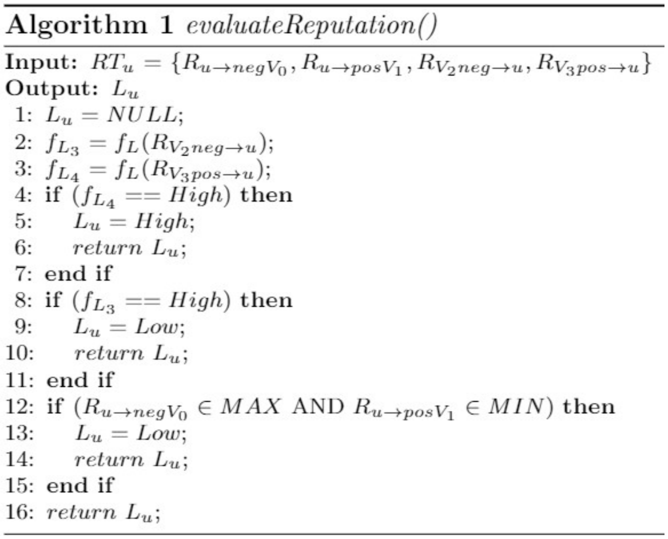


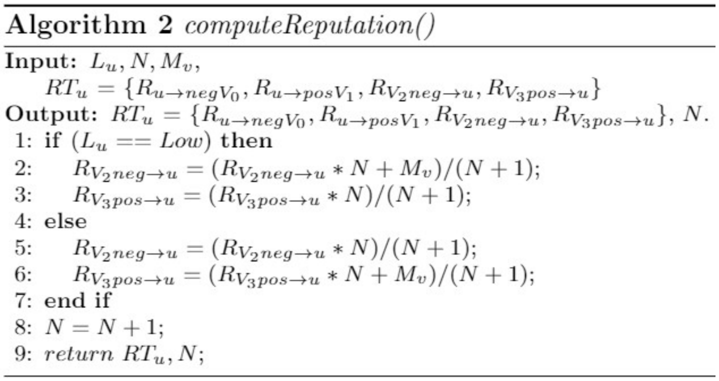
Figure 2: Reputation Level vs Reputation Score.

Evaluating Reputation. To convert a given reputation score tuple to the reputation level, *v* applies Algorithm 1. The input data of Algorithm 1 is the reputation score tuple RT_u as in Definition 2 and the output is the reputation level of *u*, i.e., L_u as in Definition 3. The reputation level is initialized with NULL as it does not contain any reputation level of *u* (line 1). Then, the levels of each reputation score, which the other users grade *u*, are converted based on Definition 2 (lines 2, 3). If the reputation level of the score which the other users positively grade *u* (i.e., f_{L_4}), is high, it means that the reputation level of *u* is high (lines 4, 5, 6). If the reputation level of the score which the other users negatively grade u (i.e., f_{L_3}) is high (lines 8, 9, 10), it means that the reputation level of *u* is low. Moreover, in the other case when the reputation evaluation of u for his past advisors, i.e., $R_{u \to neg\, V_0}$ is extremely high in the range MAX (e.g., MAX = [95%, 100%]) while $R_{u \to pos\, V_1}$ is extremely low in the range MIN (e.g., MIN = [0%, 5%]), the reputation level of *u* is low (lines 12, 13, 14). This case happens to avoid that the requesting user's grading is not correct to degrade the others, as it is not reasonable when most of advisors give bad recommendations to decrease their reputation. The other cases can be covered (line 16) since the positive and the negative evaluation percentages of the same type are complemented, which means their addition is 100%, and one user cannot grade an advisor negatively and positively for the same advice.



```
Algorithm 1 evaluateReputation()
Input: $RT_u = \{R_{u \to negV_0}, R_{u \to posV_1}, R_{V_2neg \to u}, R_{V_3pos \to u}\}$
Output: $L_u$
 1: $L_u = NULL$;
 2: $f_{L_3} = f_L(R_{V_2neg \to u})$;
 3: $f_{L_4} = f_L(R_{V_3pos \to u})$;
 4: if ($f_{L_4} == High$) then
 5:     $L_u = High$;
 6:     return $L_u$;
 7: end if
 8: if ($f_{L_3} == High$) then
 9:     $L_u = Low$;
10:     return $L_u$;
11: end if
12: if ($R_{u \to negV_0} \in MAX$ AND $R_{u \to posV_1} \in MIN$) then
13:     $L_u = Low$;
14:     return $L_u$;
15: end if
16: return $L_u$;
```

Computing Reputation. After evaluating the reputation level of *u*, *v* then decides if *v* follows the recommendation of *u*. Then, *v* can evaluate the advisor by adding or subtracting the reputation scores by a point denoted as M_v, depending on how successful v finds from the advice of *u*. As in Algorithm 2, we have *N* as the number of requesters evaluating *u*. if L_u is low, which means that the reputation of *u* is low (line 1), *v* then increases the negative points for *u* by M_v (line 2) and updates the positive points for *u* as well (line 3). In case L_u is high, *v* updates the negative points for *u* (line 5), and increases the positive points by M_v (line 6). After all, the number of requesters *N* is increased by one (line 8). The results are returned including the updated reputation score tuple RT_u and the number of requester *N*.



```
Algorithm 2 computeReputation()
Input: $L_u, N, M_v$,
    $RT_u = \{R_{u \to negV_0}, R_{u \to posV_1}, R_{V_2neg \to u}, R_{V_3pos \to u}\}$
Output: $RT_u = \{R_{u \to negV_0}, R_{u \to posV_1}, R_{V_2neg \to u}, R_{V_3pos \to u}\}$, $N$.
 1: if ($L_u == Low$) then
 2:     $R_{V_2neg \to u} = (R_{V_2neg \to u} * N + M_v)/(N + 1)$;
 3:     $R_{V_3pos \to u} = (R_{V_3pos \to u} * N)/(N + 1)$;
 4: else
 5:     $R_{V_2neg \to u} = (R_{V_2neg \to u} * N)/(N + 1)$;
 6:     $R_{V_3pos \to u} = (R_{V_3pos \to u} * N + M_v)/(N + 1)$;
 7: end if
 8: $N = N + 1$;
 9: return $RT_u, N$;
```

Updating Reputation Score. New scores are then updated directly if *u* and *v* are still in the radio range of each other. In this work, we focus on this case. Otherwise, another case is that the scores are modified when they connect to the Internet. Then, the requester *v* moves to a new position, it may leave their network, and create a new network with a sum of neighbours which may be old or new to *v*. In case, a new network includes a few old nodes with *u*, *v* can send a request to the whole network to ask for the experience with the advice of u. Based on that, *v* can decide how much trust v has for *u*. In this work, we do not focus on this direction. Our work focuses on the case that no node connecting the advisor in the previous transactions, the requester then computes the reputation score based on the data which the advisor gives it. However, a trust question happens as the advisor's reputation scores given to the requester can be forged by the advisor.

4. CRYSTAL - PRIVACY PRESERVING REPUTATION PROTOCOL

4.1. Privacy Requirements

When a requester *v* requests for the information from an advisor *u*, *u* sends *v* its reputation score tuple (see Definition 2) so that *v* can evaluate the reputation of *u* before sending *v* more information of the answer. In this situation, leaking data can happen, so the privacy requirements are needed.

(1) *Non-violated Reputation Scores against Requester*: This issue focuses on the requesters in case they try to learn the reputation scores received from the advisor. The requester *v* can misuse the received reputation score of *u*'s, e.g., impersonating the owner's received reputation score. Hence, there is a need to protect the reputation score computation.

(2) *Reputation Score Integrity against Advisor*: This privacy issue focuses on the advisor *u* in case they counterfeit their reputation scores before sending them to the requester *v*. So *v* needs to have a solution to check if the received reputation score is integrity.

In this work, we focus on proposing a secure solution to avoid the two above issues.

4.2. Homomorphic Elliptic Curve Cryptography (ECC)

ECC is a public schema key system based on the elliptic curve that is defined by an equation of the form $y^3 = x^3 + ax + b$. In ECC, the logical operators, such as Exclusive OR (+), are used to speed up the computation. As a quick view, assume that Bob wants to send message M to Alice. She generates a pair of keys including a public key *k.B* and a private key *k*, where the operator "." implies the scalar point multiplication, and B is a base point. Then, *k.B* is shared with Bob so that he can use it to encrypt *M* into a ciphertext; $Enc_{k.B}(M)$. We have $Enc_{k.B}(M) = (r.B, M + r.k.B)$ where *r* is a random number generated by Bob. Once Alice receives the encryption from Bob, she uses her private key to read the message *M* without requesting *r* from Bob, $Dec(Enc_{k.B}(M)) = M + r.k.B + r.B.k = M$.

4.3. Encryption-driven Crystal Protocol

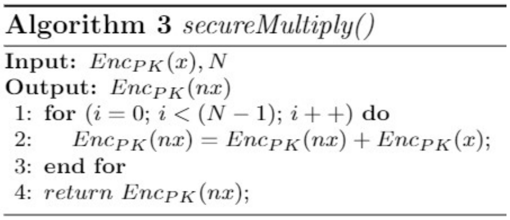
In this section, the Crystal protocol operates as presented in Section 3 but in a way that the Crystal protocol is protected with the encryption algorithms against the privacy issues addressed in Section 4.1. In this work, we accept a server which is only responsible for storing data of all nodes, providing the keys for security purposes, and evaluating the reputation of all nodes when the nodes connect to it through the Internet. The last purpose aims that the Crystal server bans or reduces the reputation score of the node that is identified to be dishonest or malicious.

a) For the first privacy issue caused by the dishonest requester. Let (PK, SK) be a pair of public and private keys, respectively, of the Crystal server. The reputation scores are encrypted by a public key of the server, i.e., PK. No users can read the plain text encapsulated inside the encryption. Therefore, all computations at the users are performed in a blind way, i.e., in a secret and secure way. For this goal, we apply the ECC algorithm (see Section 4.2) to encrypt the reputation scores to gain the secure Crystal reputation score as in Definition 3.

Definition 4 (Crystal Reputation Score). *Let u be the advisor. Let PK be the public key of Crystal server. Let RT_u be the reputation score tuple as in Definition 2. The Crystal reputation score tuple $Enc_{PK}(RT_u)$ is a set of encryptions of each element in the tuple. More formally,*

$$Enc_{PK}(RT_u) = [Enc_{PK}(R_{u\to negV_0})|Enc_{PK}(R_{u\to posV_1})$$
$$|Enc_{PK}(R_{V_2neg\to u})|Enc_{PK}(R_{V_3pos\to u})]$$

where V_0, V_1, V_2, V_3, $R_{u\to neg\,V_0}$, $R_{u\to pos\,V_1}$, $R_{V_{2neg}\to u}$, $R_{V_{3pos}\to u}$ are defined as in Definition 2.



```
Algorithm 3 secureMultiply()
Input: $Enc_{PK}(x), N$
Output: $Enc_{PK}(nx)$
1: for $(i = 0; i < (N-1); i++)$ do
2:     $Enc_{PK}(nx) = Enc_{PK}(nx) + Enc_{PK}(x)$;
3: end for
4: return $Enc_{PK}(nx)$;
```

There are two types of computing operators in Crystal, that is, arithmetic and comparative operators (see Algorithms 1 and 2).

Secure Arithmetic operators. The arithmetic operators such as + (addition), - (subtraction), * (multiply), / (division) mostly used in Algorithm 2. The reputation scores are encrypted by ECC (see Definition 4) with the public key provided by Crystal server *PK*. Operators + and – are can be enforced easily by ECC as they are basic operators of ECC. For example with operator +, the addition encryption of the two values x_1 and x_2 is performed as follows: let $Enc_{PK}(x_1)$ and $Enc_{PK}(x_2)$ be encryptions of x_1 and x_2 responsively. The encryption of the addition of x_1 and x_2 is $Enc_{PK}(x_1+x_2) = Enc_{PK}(x_1) + Enc_{PK}(x_2) = (r_1B, x_1.r_1.PK.B) + (r_2B, x_2.r_2.PK.B) = (B(r_1 + r_2), (x_1 + x_2 + PK.B.(r_1 + r_2))$ (refer Section 4.2). Then, to execute the operator * that is the multiplication of encryptions, we can exploit the operators +. The operator * of two operands $Enc_{PK}(x)$ and *N* are executed using the operator + as in Algorithm 3. There is a loop of $(N - 1)$ times (line 1) of adding $Enc_{PK}(x)$ into the output data $Enc_{PK}(nx)$ (line 2). Hence, we have $\Sigma_{i=0}^{(N-1)} Enc_{PK}(x) = Enc_{PK}(x + \cdots + x) = Enc_{PK}(Nx)$. For the operator / of $Enc_{PK}(x)$ and *N*, we need to calculate the inverse value of the denominator, denoted N^{-1}, by a binary polynomial inversion calculation. The ECC operators are used for enforcing the operators in Algorithm 2.

Moreover, Crystal executes the comparison operations on the encrypted reputation scores (see Definition 4) for evaluating the reputation level as in Definition 3 or for checking conditions to re-compute the reputation scores (see Algorithm 2).

Secure comparative operators. For this type of comparative operator, we adopt the framework supporting the secure comparison in [13]. This protocol leverages the asymmetric encryption to secretly evaluate if an encryption satisfies a condition in the form of *cond = (encrypted variable, operator, threshold)*, where the operators include: <, >, ≤, ≥, =, ≠. Specifically, this protocol generates a token T for each condition. This is done by using the *T = GenToken(SK,< cond >)* function defined in [13], where SK is the Crystal server's private key generated according to [13]. Crystal server's SK is used to ensure that only Crystal server can generate tokens containing the secret conditions unknown to the unauthorized nodes. In order to evaluate whether the encryption of value *x* with the corresponding public key PK, i.e., $Enc_{PK}(x)$, satisfies *cond*, we make use of the *Query()* function defined in [13]. This takes as input the encryption $Enc_{PK}(x)$ and the token T generated for cond and returns a predefined message M, if cond is satisfied, ⊥, otherwise. As in Algorithms 2, a condition for if statement is *(L_u, =, "Low")*, and as in Algorithm 1, four conditions for if statement are *(f_{L_4}, =, "High")*, *(f_{L_3}, =, "High")*, *($R_{u\to neg\,V_0}$, =, 100)*, and *($R_{u\to pos\,V_1}$, =, 0)*. In order to use the function *Query()* for the above conditions, the variables in the conditions are encrypted. Therefore, we have *($Enc_{SK}(Lu)$, =, "Low")*, *($Enc_{SK}(f_{L_4})$, =, "High")*, *($Enc_{SK}(f_{L_3})$, =, "High")*, *($Enc_{SK}(R_{u\to neg\,V_0})$, =, 100)*, and

(EncSK($R_{u \rightarrow pos\, V_1}$*), =, 0)*. The returned result can be of these conditions are "true" or "false". Moreover, we have three other conditions in Definition 3, that is, *(*$Enc_{SK}(R_u)$*, <, 30)* returns "Low" or ⊥, while *(*$Enc_{SK}(R_u)$*, >, 70)* and *(*$Enc_{SK}(R_u)$*, ≤, 100)* return "High" or ⊥.

By this cryptographic method, we can achieve the first privacy requirement (see Section 4.1) in *protecting the reputation score against dishonest requesters.*

b) *For the second privacy issue of data integrity against dishonest advisor*. We adopt the digital signature mechanism, specifically, that is the Elliptic Curve Digital Signature Algorithm (ECDSA) [14]. Nodes receive or update the public keys of the other peers using the Crystal application periodically from the Crystal server when they can access the Internet. Each node has its own private key called SK_v. This private key is used for generating the digital signature of the secure reputation score tuple (see Definition 4) after the requester *v* uses *u*'s advice and updates *u*'s reputation scores. However, to make the following requester evaluate the integrity of the reputation scores, *v* needs to add its ID after the digital signature of the secure reputation score tuple as well, so that the following requester can obtain *v*'s public key from *v*'s ID and verify the digital signature to ensure that it was made by a requester different from *u*. We denote the data structure, consisting of the secure reputation score tuple, digital signature and *v*'s ID, as reputation score of integrity. More formally,

Definition 5 (Reputation Score of Integrity). *Let u, v be the advisor and the requester, respectively. Let PK, SK_v be the public key of Crystal server and the private key of the requester v, respectively. Let $Enc_{PK}(R_u)$ be the secure reputation score tuple as in Definition 4. Let I_{R_u} be the reputation score of integrity of u. More formally,*

$$I_{R_u} = [Enc_{PK}(R_u) \| Sign_{ECDSA}(Hash(Enc_{PK}(R_u))) \| ID_v]$$

This reputation score of integrity I_{R_u} is used instead of the secure reputation score tuple. Only *v* has its own private key to make the digital signature, so the other nodes cannot know as well as cannot forge v's private key. Let w be the following requester. When w receives I_{R_u} from *u*, *w* extracts ID_v and checks if that is *u*'s ID. If it is true, the data integrity is not ensured. Otherwise, *w* searches the public key of *v*, that is, PK_v. Then, *w* decrypts $Sign_{ECDSA}(Hash(Enc_{PK}(R_u)))$ by PK_v to get $h_1 = Hash(Enc_{PK}(R_u))$. Then, *w* hashes $h_2 = Enc_{PK}(R_u)$ to compare if $h_1 = h_2$. If they are equal, the data integrity is maintained. Otherwise, it is not. Then, the Crystal protocol is enforced and *w* updates the reputation scores, then generates the signature with its private key and attaches its ID_w into I_{R_u}.

5. EXPERIMENTATION

We compare our work with the work in [7] as both leverage the cryptographic algorithms although we do not solve the same problem. The difference in our approaches is that in [7] the authors use Paillier algorithm while we focus on the Elliptic Curve cryptographic algorithms.

Parameters. We perform several experiments using the cryptographic algorithms ECC, ECDSA and Paillier as well as their different key sizes. The experiments are performed on one PC of Windows 10 Professional, 8 GB RAM, CPU i7 1.80GHz. Each value in the experiments is an average of 20 execution times. The Koblitz Elliptic Curves, which are one kind of 15 recommended elliptic curves analysed in [15], are used. Their key sizes are variant in {233, 283, 409, 571} bits. SHA-2 key sizes are in {224, 256, 384, 512} bits. ECDSA key sizes are in {384, 521} bits. Paillier key sizes are in {1024, 2048, 3072, 4096} bits. Currently, these key sizes satisfy the NIST's conditions against the analysis attacks[2].

[2] https://www.keylength.com/en/4/

Time cost. We create 8 combinations of cryptographic algorithms' key sizes, that is, SHA key size, ECDSA key size = {(224, 384), (256, 384), (384, 384), (512, 384), (224, 521), (256, 521), (384, 521), (512, 521)}. For each pair of SHA-2 and ECDSA keys, the key sizes of ECC and Paillier are variant. So, we have 8 different values for each pair of SHA-2 and ECDSA keys. We compare the computing time in generating one reputation score of integrity for each case of key sizes of {SHA-2, ECDSA, ECC} and {SHA-2, ECDSA, Paillier} as in Figure 3. In all cases of using ECC, the computing time costs are always much less than using Paillier. In the smallest key sizes, of {SHA-2, ECDSA, ECC} = {224, 384, 233} and {SHA-2, ECDSA, Paillier} = {224, 384, 1024}, the combination of key sizes containing ECC costs 36,1ms while the combination of key sizes containing Paillier costs 359.9ms.In the highest key sizes of {SHA-2, ECDSA, ECC} = {512, 521, 571} and {SHA-2, ECDSA, Paillier} = {512, 521, 4096}, the combination of key sizes containing ECC costs 69,98ms while the combination of key sizes containing Paillier costs 390.5ms.

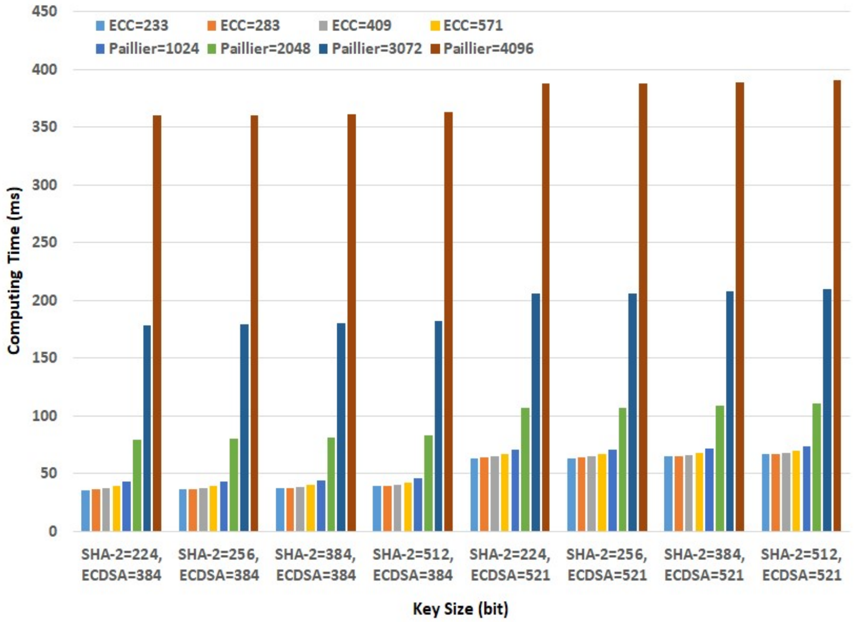


Figure 3: Time cost comparison.

Data Load. We create 8 combinations of key sizes of ECC and ECDSA, that is, (ECC, ECDSA) = {(233, 384), (283, 384), (409, 384), (571,384), (233, 521), (283, 521), (409, 521), (571,521)}, and 8 combinations of key sizes of Paillier and ECDSA, that is, (Pailler, ECDSA) = {(1024, 384), (2048, 384), (3072, 384), (4096,384), (1024, 521), (2048, 521), (3072, 521), (4096, 521)}. We compute and compare the data payloads of the reputation score of integrity with different combinations of key sizes as in Figure 4. In the case of the smallest key sizes of 384-bit ECDSA, 233-bit ECC, 1024-bit Paillier, the payload created from ECC is 446 bytes while the one created from Pailler is 675.2 bytes. In the case of the largest key sizes of 521-bit ECDSA, 571-bit ECC, 4096-bit Paillier, the payload created from ECC is 2248 bytes while the one created from Pailler is 2283 bytes. Therefore, the data payload created by ECC is always smaller than the one created by Paillier.

6. Conclusions

We investigate a scenario of a decentralized close area where users cannot access the Internet, and propose a privacy-preserving protocol supporting the requesting user to evaluate the reputation level of an advising user before following his recommendation. Additionally, the proposal presents the secure process of reputation evaluation and computation against the privacy violation of the requester and the advisor. The future works will cope with the authentication issues for nodes as well as the privacy issues against the Crystal server.

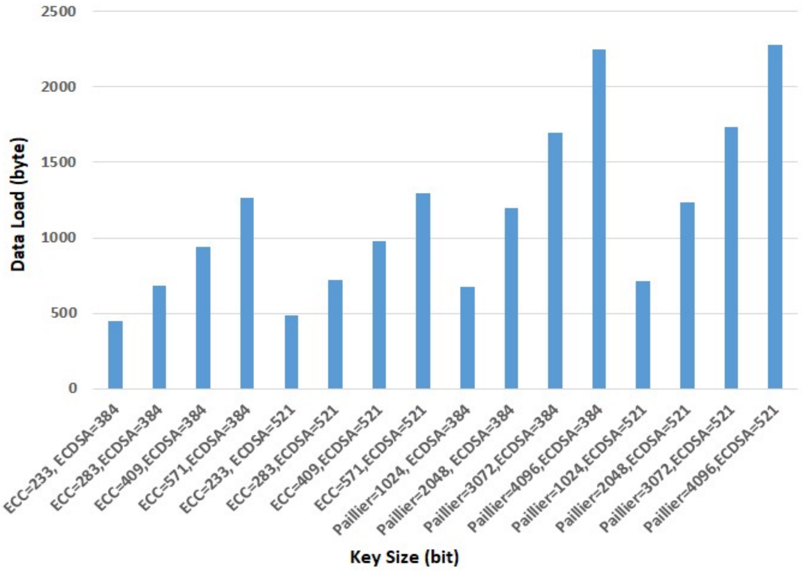


Figure 4: Data load comparison.

Acknowledgements

This work is partially supported by TrustedMaaS project - the Infotech institute of the University of Oulu, Science, and Academy of Finland 6Genesis Flagship (Grant 318927).

References

[1]. A. K. Wong, "The near-me area network," *IEEE internet computing*, vol. 14, no. 2, pp. 74–77, 2010.

[2]. P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Communications of the ACM*, vol. 43, no. 12, pp. 45–48, 2000.

[3]. P. Resnick and R. Zeckhauser, "Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system," in *The Economics of the Internet and E-commerce*, pp. 127–157, Emerald Group Publishing Limited, 2002.

[4]. O. Hasan, "A Survey of Privacy Preserving Reputation Systems" [Technical Report] LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/École Centrale de Lyon. ffhal-01635314f, 2017.

[5]. O. Hasan, L. Brunie, and E. Bertino, "Preserving privacy of feedback providers in decentralized reputation systems," *Computers & Security*, vol. 31, no. 7, pp. 816–826, 2012.

[6]. O. Hasan, L. Brunie, E. Bertino, and N. Shang, "A decentralized privacy preserving reputation protocol for the malicious adversarial model," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 949– 962, 2013.

[7]. R. Petrlic, S. Lutters, and C. Sorge, "Privacy-preserving reputation management," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 1712–1718, ACM, 2014.

[8]. P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 223–238, Springer, 1999.

[9]. E. Gudes, N. Gal-Oz, and A. Grubshtein, "Methods for computing trust and reputation while preserving privacy," in *IFIP Annual Conference on Data and Applications Security and Privacy*, pp. 291–298, Springer, 2009.

[10]. O. Baudron, P.-A. Fouque, D. Pointcheval, J. Stern, and G. Poupard, "Practical multi-candidate election system," in *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing*, pp. 274– 283, ACM, 2001.

[11]. J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach. USA: Addison-Wesley Publishing Company*, 5th ed., 2009.

[12]. Z. Pei, Z. Deng, B. Yang, and X. Cheng, "Application-oriented wireless sensor network communication protocols and hardware platforms: A survey," in *2008 IEEE International Conference on Industrial Technology*, pp. 1–6, IEEE, 2008.

[13]. D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Theory of cryptography*, pp. 535–554, Springer, 2007.

[14]. D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ecdsa)," *International Journal of Information Security*, vol. 1, no. 1, pp. 36–63, 2001.

[15]. H. D. and M. A., "Chapter nist elliptic curves, encyclopedia of cryptography and security (2nd ed.)," 2011.

[16]. E. Zhai, D. I. Wolinsky, R. Chen, E. Syta, C. Teng, and B. Ford. "Anonrep: Towards tracking-resistant anonymous reputation, 13th Usenix Conference on Networked Systems Design and Implementation, pp. 583–596, 2016.

[17]. T. Minkus and K. W. Ross. "I know what you're buying: Privacy breaches on ebay", 14th International Symposium Privacy Enhancing Technologies, pp. 164–183, 2014.

[18]. D. Maram, H. Malvai, F. Zhang, N. Jean-Louis, A. Frolov, T. Kell, T. Lobban, C. Moy, A. Juels, A. Miller, "CanDID: Can-Do Decentralized Identity with Legacy Compatibility, Sybil-Resistance, and Accountability", IACR Cryptology ePrint Archive, Report 2020/934, 2020.

[19]. F. Zhang, S. Krishna, D. Maram, H. Malvai, S. Goldfeder, A. Juels, "DECO: Liberating web data using decentralized oracles for TLS", ACM Conference on Computer and Communications Security, 2020.

[20]. M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli and M. H. Rehmani, "Applications of blockchains in the Internet of Things: A comprehensive survey," IEEE Communication Surveys Tuts., vol. 21, no. 2, pp. 1676–1717, 2019.

[21]. S. Brotsis, N. Kolokotronis, K. Limniotis, S. Shiaeles, D. Kavallieros, E. Bellini, C. Pavué, "Blockchain solutions for forensic evidence preservation in IoT environments", IEEE Network Softwarization (NetSoft), pp. 110–114, 2019.

[22]. A. Bellini, E. Bellini, M. Gherardelli, and F. Pirri, ''Enhancing IoT data dependability through a blockchain mirror model", Future Internet, vol. 11, no. 5, p. 117, 2019.

[23]. B. Shala, U. Trick, A. Lehmann, B. Ghita, and S. Shiaeles, ''Blockchain based trust communities for decentralized M2M application services", Advances on P2P, Parallel, Grid, Cloud and Internet Computing, vol. 24, 2018.

[24]. E. Bellini, P. Ceravolo, and E. Damiani, ‘‘Blockchain-based e-vote-as-a-service”, 12th IEEE Conference on Cloud Computing (CLOUD), pp. 484–486, 2019.

[25]. E. Bellini, ‘‘A blockchain based trusted persistent identifier system for big data in science”, Foundation Computing Decision Science, vol. 44, no. 4, pp. 351–377, 2019.

Authors

Ngoc Hong Tran is a lecturer at the Vietnamese German University, Vietnam, since 2016. She obtained a Bachelor Degree in Information Technology, and a Master Degree in Computer Science, from University of Science, Vietnam National University - Ho Chi Minh City. She achieved a PhD Diploma in Computer Science from University of Insubria, Italy (2016). Moreover, she had been a lecturer at the University of Science, VNU-HCM. She also worked in National Institute of Informatics, Tokyo, Japan, in 2009, and was an exchange scholar in Portland State University (PSU), Portland City, Oregon State, USA, in 2010. She worked in LAAS-CNRS, Toulouse, France, in 2012. She was a researcher in Singapore University of Technology and Design, Singapore, from March to April 2017, and was a postdoctoral researcher in University College Dublin, Ireland, 2018-2020. She has been a reviewer of several international conferences. Her research interests are security and privacy in a variety of scenarios, as in healthcare, data mining, block chain, 6G, IoT, distributed collaboration, mobile social network, web composition, network coding.

Tri Nguyen was born in Ho Chi Minh, Vietnam, in 1993. He received the B.Sc. degree in computer science from the University of Information Technology - Vietnam National University, Vietnam in 2015, and the M.Sc. degree in computer science from the University of Pisa, Italy, in 2018. Since 2018, he has been a doctoral student in the Center for Ubiquitous Computing, University of Oulu. His research interests include distributed systems, blockchain technology, and information security.

Quoc-Binh Nguyen is currently working as a lecturer of Faculty of Information Technology of Ton Duc Thang University. He obtained the Bachelor Degree in Information Technology in 2009 and Master Degree in 2012 at University of Science, VNU of Ho Chi Minh City. His research interest is security and privacy.

Susanna Pirttikangas was born in Kemi, Finland in 1973. She received the M.Sc. degree in theoretical mathematics from the University of Oulu, in 1998 and the D.Sc. (tech.) degree in embedded systems in University of Oulu in 2004.

Dr. Pirttikangas works as a research director in the Center for Ubiquitous Computing at the University of Oulu. She made her postdoctoral visits to Waseda University, Japan (2005-2006), Tokyo Denki University, Japan (2008) and Tsinghua University, China (2011). Her research team Interactive Edge develops adaptive, realiable and trusted edge computing. She is an active member of the international research community as a workshop and conference organizer, as well as serving as a reviewer and PC member in top journals and conferences in her field. Dr. Pirttikangas also works as a freelance lead AI scientist in a Finnish company Silo.AI.

M-Tahar Kechadi was awarded PhD and Masters degree - in Computer Science from University of Lille 1, France. He joined the UCD School of Computer Science (CS) in 1999. He is currently Professor of Computer Science at CS, UCD. His research interests span the areas of Data Mining, distributed data mining heterogeneous distributed systems, Grid and Cloud Computing, and digital forensics and cyber-crime investigations. Prof Kechadi has published over 260 research articles in refereed journals and conferences. He serves on the scientific committees for a number of international conferences and he organised and hosted one of the leading conferences in his area. He is currently an editorial board member of the Journal of Future Generation of Computer Systems and of IST Transactions of Applied Mathematics-Modelling and Simulation. He is a member of the communication of the ACM journal and IEEE computer society. He is regularly invited as a keynote speaker in international conferences or to give a seminar series in some Universities worldwide.

INTERNET OF THINGS (IOT): DATA SECURITY AND PRIVACY CONCERNS UNDER THE GENERAL DATA PROTECTION REGULATION (GDPR)

Olumide Babalola

School of Law, University of Reading,
Whiteknights, Reading, United Kingdom

ABSTRACT

Internet of Things (IoT) refers to the seamless communication and interconnectivity of multiple devices within a certain network enabled by sensors and other technologies facilitating unusual processing of personal data for the performance of a certain goal. This article examines the various definitions of the IoT from technical and socio-technical perspectives and goes ahead to describe some practical examples of IoT by demonstrating their functionalities vis a vis the anticipated privacy and information security implications. Predominantly, the article discusses the information security and privacy risks posed by the operationality of IoT as envisaged under the EU GDPR and makes a few recommendations on how to address the risks.

KEYWORDS

Data Protection, GDPR, Information Security, Internet of Things, Privacy.

1. INTRODUCTION

In its simplest form, Internet of Things (IoT) connotes the seamless interconnectivity, inter-relativity, and interaction of animate and inanimate objects towards the performance of specific tasks. The concept or technology enables the smart communication of two or more objects co-existing digitally or otherwise for a pre-determined or anticipated outcome or set of outcomes.

The IoT - a coinage by Kevin Ashton in 1999, a British technocrat who co-created a global standard for radio-frequency identification (RFID) - has become a household name to describe the functionality of artificial intelligence (AI) deployed to initiate and consummate a wide variety of human related activities or provision of services.[1] The notion of IoT surfaced along with the invention of the worldwide web but was used for the first time in 1999 with the principal objective of developing technologies that would enable the cross communication and interconnectivity of remote digital devices as part of the 'embedded computer system.'[2] Porras, et al however conversely argue that the first modern notion of IoT was rather introduced by Mark Weiser in his 1999 article where he mused about 'interconnected devices that disappear into the background of our everyday lives.'[3]

This article first introduces IoT as a relatively new technology enabling inter-relativity of multiple devices through connectivity-enhancing sensors and control systems while the second part reproduces the various definitions of the concept from academic and technical perspectives and the third describes some practical examples of IoT and the fourth part analyses the data

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 309-320, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112324

security issues plaguing the functionality of IoT whereas the fifth part analyses the privacy concerns in IoT and then then the sixth provides recommendation on solutions to the issues while the last part concludes with a recap of the issues discussed.

2. CONCEPTUAL DEFINITIONS

The various definitions of IoT are coloured by origins and vision and sometimes the perspectives of the author making such attempt. The concept has been interchangeably referred to or conflated with terms like Internet of Everything (IoE)[4], Machine to Machine (M2M),[5] Cloud of Things, (CoT),[6] Internet of People (IoP)[7] and Web of Things (WoT)[8] which terms have been given similar or divergent connotations with the IoT.[9]

However, a number of authors and stakeholders have attempted defining IoT along the line of divergent interests and proclivities. The International Telecommunications Union (ITU) views IoT as ‘a global infrastructure for the information society enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies (ICT)’[10] and in a similar but not identical attempt, the Internet Oriented Vision defines IoT as ‘a global infrastructure that enables connectivity between both virtual and physical object.[11]

Some definitions however compartmentalize IoT only in relation to physical objects as opposed to the inter-relativity between animate and inanimate entities. For example, Al- Fagaha, et al argue that IoT is a technology that allows physical objects to perceive, hear, see, analyse and undertake tasks by having them interact to exchange data and, ‘relay information to one another, process the information collaboratively, and take action automatically.’[13] This attempt however limits the operationality of IoT to the Internet thereby disregarding the workability of offline digital platforms and their functionalities. In another attempt that overlooks the (human) users of IoT, Whitmore, et al however view IoT as ‘a paradigm where everyday objects can be equipped with identifying, sensing, networking and processing capabilities that will allow them to communicate with one another and with other devices and services over the internet to accomplish some objectives.’[14]

In creating a basis for researchers and academics to further expound on the definitions of IoT, one of the European Commission’s intervention initiatives defined the concept as a universally connected network of infrastructure ‘linking physical and virtual objects through the exploitation of data capture and communication capabilities.’[15] Flowing from this, Weyrich and Ebert associate IoT with ‘innovative functionality and better productivity by seamlessly connecting devices’[16] but in a more elaborate approach, Tarkoma and Katasoner define the concept as ‘a global network and service infrastructure of variable density and connectivity with self-configuring capabilities based on standard and interoperable protocols and formats (which) consists of heterogenous things that have identities, physical and virtual attributes and are seamlessly and securely integrated into the internet for clarity.’[17] This definition aligns with notion that socio- technical dimensions to IoT envisages the interaction of the mechanical components with their non-technical counterparts within the same artwork.[18]

3. EXAMPLES OF IOT

Superficially, from the preceding definitions, the concept of IoT appears abstract but with the paradoxical [36] intrusion of technology into homes and private affairs, IoT lives with an average human that envisaged. In this part, I will briefly discuss some contemporary examples or manifestations of IoT around.

3.1. Smart homes or automated homes

These are houses or living environments where technology is used to monitor or control the home appliances remotely in two folds: one consists of the automated home devices and the other relates to their interface, processing and intercommunication.[37] The introduction of IoT into homes remotely controls and coordinates the occupants' individual or joint security needs, medical needs, entertainment preferences, business services, occupational needs and other living needs.[38]

Since smart homes are equipped with ICT which anticipates and responds to the needs of occupants of a house, they necessarily perform their functions after analysing the users' personal information in relation to those needs and the repeated processing activities outside occupants' control raise presumptions of privacy invasion and misuse of such personal data.[39] Within the IoT and Smart homes network, personal data are collected, shared, exchanged and transmitted between several exposed platforms in a manner that robs the users of reasonable control over such personal information and thereby puts them in imminent and imagined risks of privacy violation.[40]

3.2. Wearable devices

Wearable devices are electronic or digital gadgets and software integrated into clothing or worn as accessories for processing information from time to time.

They are manufactured with in-built sensors that enable them track day to day activities of users by syncing them with remote mobile devices. These devices by their operational nature periodically collect users' personal data, share them with other remotely connected devices and ultimately store them in clouds making them vulnerable to attacks, data leakages and breaches with the ultimate end result of privacy invasion. Wearable device like smart bracelets or smart glasses utilize sensors to capture users' sensitive data like pulse, heart rate, blood lipid, blood pressure and other health data and synchronized with health centres' devices to detect early symptoms or supervise health status.[41]

The privacy gaps in the processing activities undertaken by the operators of the wearable devices are accentuated by lack of uniform industry regulation on their transmission formats, encryption and confidentiality especially regarding the further use or indefinite storage of the personal data collected on daily basis. [42]

3.3. Automated vehicles (AV)

Automated vehicles are also referred to as 'fully automated vehicles' or self-driving cars' or 'driver-less cars.'[43] These vehicles are automated to function without human drivers but their navigation is aided by algorithm and sensors using cameras, imaging technology and location-sensitive chips to gather information about the vehicle's location and other information which impart the vehicle owners' expectation of privacy.[44] Most personal data processed during the operationality of AVs are stored in the cloud outside the control of users within the custody of third parties who do not have direct contact with users and provide no guarantees against misuse of such sensitive personal data.

4. INFORMATION SECURITY IN IOT

The ubiquity and dynamism of IoT explicably exposes the technology to a wide array of data security issues.[45] Porras identifies nine primary categories of security concerns raised within IoT as: environmental constraints, vulnerable devices, data security, functional constraints, enforcement mechanism, cross device dependencies, identification, authentication and authorization, control legislation and attacks- threats, modes. [46] However, the concern of this borders on data/ information security – a term often conflated or confused with cybersecurity. While cybersecurity refers to the 'collection of tools, security concepts, security safeguards, guidelines, risk management approaches, actions, training, best practices, assurance and technologies that can be used to protect the cyber environment and organization and users' assets, [47] information security on the other hand is 'the protection of information and its critical elements including the systems and hardware that use store and transmit that information.[48]

In the context of IoT, data security borders on confidentiality, integrity, availability, accuracy, authenticity, utility, and possession of personal information within the confines of the relevant data privacy laws applicable in the respective jurisdictions of the IoT concerned. I shall consider some of this in turn.

4.1. Data confidentiality and integrity

Confidentiality and integrity are one of the universal principles of data protection. The EU General Data Protection Regulation (GDPR) mandates data controllers to ensure appropriate security of personal data against accidental loss, destruction or damage through formidable technical or organization measures. [49] By the nature of IoT, massive complex processing activities take place simultaneously necessitating appropriate data confidentiality and integrity mechanisms to prevent loss or destruction of personal information. Contemporary and modern techniques must be employed to shield access to personal data from authorized third parties or malicious destruction. In this light, Chanal suggests some confidentiality-preservatives like Message-Digest (M05), Ellipte Curve Cryptography, Advanced Encryption Standard (AES) and Algorithms for efficient communication in IoT without the fear of eavesdropping, theft of personal data or compromise of the information in any form. [50]

While access control and cryptography have been suggested as mechanisms that reduce the risk of data manipulation, unauthorized access or misappropriation in IoT, their protective coverage do not extend to already disseminated or transmitted personal data but Minch alternatively advises the use of confidential policy in IoT to analyze information flow which ought to culminate in information policies for the networked systems.[51]

4.2. Data accuracy

This is another principle of data protection deeply rooted in the OECD principle of data quality.[52] It stipulates that personal data stored by entities must reflect the true and correct information of data subjects and where they are outdated, such data must be updated or deleted completely. For the IoT, data accuracy is regulated by the source of collection of data and ultimately the storage mechanism which ought to facilitate periodic and necessary updates.

Personal data is the lifeblood of IoT, because they provide the link between the connected devices on one hand and clarity on the nature of expected outcome via the intercommunication of the entities involved, hence, the quality and accuracy of the personal information transmitted within the interconnected entities must not only be verified but sustained.

Karkouch however notes that while data quality or accuracy in IoT is vitiated by: deployment scale, sensors, constrained resources and intermitted loss of connection, these negative effects can be cured by various relevant data cleaning techniques. [53] Inaccurate (personal) data processed within the IoT system does not only violate data protection principles and users' rights, it compromises the objectives and outcomes of the IoT processing activities making it unreliable of unfit for purpose. In exercise of the right of access [54] to their personal information processed in IoT, users can request from the operators of such technologies, copies of their personal data processed to ensure accuracy of data and as well as ensuring transparency of processing activities involved in the IoT ecosystem when it is ultimately, considered that, IoT could constitute problems to their operators or users where personal data used are inaccurate or outdated.

4.3. Misuse or unauthorised possession of (personal) data

The main objective of data security is prevention of data breach in the form of data loss-(availability breach) or misuse of personal data (utility breach). The risks of data breach vary for different kinds of devices in a IoT network, hence, the need for appropriate and befitting IoT security measures for the respective systems. IoT security is 'a technology area that addresses the protection of the security and privacy of data and information in the physical world as well as in the digital world.' [55]

The IoT functionality involves some external and exposed cross-transmission of personal data on various platforms which may be intercepted by middlemen and third parties through the use of sniffing stations.[56] Other security issues such as robustness, reliability, safety, resilience, performability and survivability may also plague data IoT but it must however be noted that while all these issues impact vehicular data, they do not all relate to personal data as far as IoT security is concerned.[57]

5. (INFORMATION) PRIVACY IN IOT

The functionality of IoT thrives in a multitude of data processing activities. Personal data are used to assess users' preferences, lifestyle, social activities and to ultimately create a profile for marketing or other purposes. Informational privacy is the shade of privacy that interplays with IoT when users' information are shared between several interconnected devices to provide certain services, thereby exposing the users to privacy risks. While one concedes that the IoT's utility of personal data ultimately improves service delivery by making them unusually seamless, however this advantage ought to be balanced against the essentiality of right to privacy especially where the data are amassed without (informed) consent or legal basis. [58]

Consent, in this context is, any freely given, specific, informed, and unambiguous indication of the data subject's wishes by which he or she, by a statement or by a clear affirmative action, signifies agreement to the processing of personal data relating to him or her [59] while informed consent in IoT refers to 'the process by which a fully informed user participates in decisions about his or her personal data.' [60] Like found in some online transactions, the ubiquity of IoT however sometimes makes it impracticable for informed consent to be sought and obtained especially since consent envisages an affirmative indication of agreement to surrender data for certain purposes.

Privacy of IoT always raises the issue of trust since the network of devices utilize massive personal data which processing ought to align with users' expectations of privacy and other freedoms. In IoT, the personal data processed may reveal information on users' location, financial data, health data, home, family, sexuality etc. hence the paradigm requires protection of these sensitive data to guarantee users privacy even when exchanged on varying platforms over which the users do not have reasonable control. [61] Ziegeldorf et al argue that ‘the increasingly invisible dense and pervasive collection, processing and disseminating’ of users’ personal data raise serious privacy concerns for IoT enablers like RFID, wireless sensor networks (WSN), web personalization and mobile application platforms. [62]

Privacy in IoT guarantees tripod protection for users to wit: (a) transparency risks posed by AI in IoT (b) personal autonomy over personal data collected and (c) knowledge and control of future utility of personal data, all through five different type of information flows of interaction, collection phase, processing, dissemination, and presentation phases. [63] Even though all the phases technically constitute data sharing and processing phases, they depict the IoT’s cycle of personal data handling in the light of magnitude of personal data surrendered by users’ to IoT and their consequential exposure to privacy risks.

6. Managing Privacy and Security Risks in IoT under the GDPR

Many authors [64] have proffered technical solutions for addressing security and privacy risks in IoT but our concern here relates to the legal or regulatory management of information privacy and security in the networked technology. The GDPR which now represents the global benchmark for data privacy [65] is applicable to IoT in so far as it processes the personal data of EU residents or operated by an EU-based entity or targeted at EU customers.[66] Operators or manufacturers of IoT (as data controllers) are obligated under the GDPR to take certain measures to ensure data security and privacy of users of their products.

6.1. Identifying the controller(s) in IoT systems

The whole essence of data protection laws especially the GDPR is the apportionment of liabilities and responsibilities to the stakeholders in every relevant data processing eco-system. Under the GDPR, while the ‘controllers’ determine the purpose(s) and means of processing personal data, the ‘joint controllers’ are two or multiple controllers that jointly determine means and purpose of processing and then ‘processors’ are engaged under contract with definitive terms to process personal data on behalf of controllers, while ‘recipients’ are either employees or other entities to which personal data are disclosed and ‘third parties’ are entities who does not qualify as any of the preceding parties listed here.[67]

In IoT systems a decision on the party responsible for ensuring privacy and data security must necessarily begin with an inquiry into whether or not the developer or manufacturer or seller of IoT is the controller, joint controller or processor. Recital 78 and article 25 GDPR requires a controller to consider and implement the principles of data protection at the point of determination of the means of processing and its implementation and with respect to processors, article 28(1) (b) also requires the implementation of ‘appropriate technical and organizational measures’ to protect personal data.

The complex nature of processing activities undertaken by independent developers of certain components [68] of IoT systems render them liable to qualify as joint data controllers or independent controllers except in rare cases where they may not process personal data in the developmental stages.[69] Hadzovic however identifies a host of other players in the IoT network with varying data processing roles to wit: data manager, service providers, IoT data provider, IoT framework provider, IoT data application provider, IoT data carrier etc.[70] Ultimately, every developer of a component in an IoT network is independently or jointly responsible for ensuring protection of privacy by implementing the GDPR principles except personal data was not processed during developmental stages. The table below illustrates the apportionment of responsibilities under the GDPR but for this purpose of this paper, we are only concerned with the systems developers, components developer, IoT users and IoT managers.

Actors	Designation undo GDPR	Responsibility under the GDPR
IoT systems developer	Controllers	Design IoT to incorporate data protection principles like data minimization storage limitation, lawfulness of processing & transparency, confidentiality and integrity etc. (GDPR, art.24)
IoT component developer	Controllers, joint controllers, or processors (depending on terms of engagement)	Comply with the obligations of controllers, and/or as processor provide sufficient guarantees to implement appropriate threshold and organizational measures (GDPR, art. 28)
IoT Users/ consumers	Joint controllers/ independent controllers	Ensure the utility of IoT does not violate others' data privacy rights and fulfil obligations under the GDPR.
IoT managers	Controllers/processors. (Depending on the stage they come into the picture.)	Ascertain the compliance of IoT with GDPR principles and ensure the regulatory measures are implemented to minimize risk of data privacy rights violation.

6.2. Use of privacy statements

IoT collect information round the clock for intermittent use and sometimes store them indefinitely hence, users of IoT platforms are entitled to information on how, why and when their personal data are collected, stored and used. Article 13 GDPR guarantees data subjects right to information on: (a) identity (b) identify of controller, details of data protection office, purpose of process (c) legitimate interest (d) recipients etc.

Operators of IoT can fulfil their obligation to provide information by utilizing privacy statements. These are either designated as privacy notice or policies that fully explain entities' collection purpose, use, storage, and overall management of personal data and therein giving the users a choice over their preference for processing activities on their personal data. These statements enable users develop trust in the IoT systems and reduce the apprehension of privacy risks more so through the data controllers' transparency as clearly spelt out in the statements. Ultimately, to achieve optimum result, privacy policies/notices in IoT ought to be summarized to aid comprehension, categorized, well organised, and automated into the respective systems.[71]

6.3. Data protection by design and default

This is one of the hallmarks of the GDPR which introduced an additional obligation on data controllers (IoT manufacturers in this context) to integrate, at the point of construction 'Privacy-Enhancing- Technologies' (PETs) and throughout the life cycles of the IoT.[72] Article 25 GDPR imposes a duty on IoT manufacturers and operators to integrate technical and organizational measures in the system, to fulfil the data protection of personal data and privacy of users.

In integrating privacy by default and design in IoT, manufacturers must identify and ascertain the legal basis to process users' information, ensure security of information collected and prevent misuse of such personal data which must not be stored for more than necessary and minimize the quantum of irrelevant data collected by the devices in the network. Article 29 Working Party recommends the utility of 'shielding techniques' or 'kill commands' to address unauthorized or unanticipated tracking of personal data belonging to users of IoT. [73] For enhancement of privacy in IoT, Larrieux suggests the use of other PETs like 'thresholding the transmission of information based on signal strength, protecting passwords and using hash-locks or metalDs.'[74]

7. Conclusion

With the rise in human dependence on IoT comes privacy and security challenges associated with the intrusive and invasive tendencies of the ubiquitous technology. Of all issues militating against IoT, privacy and security concerns rank top in spite of the seaming wilful surrender of personal information by users – this underscores the privacy paradox of IoT platforms.

In this article, I have briefly discussed the origin of IoT as well as the various academic definitions of the concept to show its nature, objectives and nuances. I have also analysed how privacy and data security constinue to pose threats to the seamless utility and operationalism of IoT and here, I have also proffered some quick or long fixes to the process.

References

[1] Keyul K. Patel & Sunil Patel, (2016) 'Internet of Things - IoT: Definitions, Characteristics, Architecture Enabling Technologies, Application and Future challenges' 6(5) IJESC, VOL. 6 NO. 5, p 6122.

[2] Jorge E. Ibara-Esquer et al, (2017) 'Tracking the Evolution of Internet of Things Concept Across Different Application Domains' Sensors Journal, Vol. 17, p1.

[3] Jari Porras, et al, (2018) 'Security in the Internet of Things- A Systematic Mapping Study' Proceedings of the 51st Hawaii International Conference on System sciences.

[4] Langley et al argue that IoE is an expanded and broadened version of IoT by throwing people, business and other processes into the mix. They describe IoE as 'a network of connections between smart things, people, processes, and data with real-time data/information flows between them.' See David Langley, (2021) 'The Internet of Everything: Smart Things and their Impact on Business Models' Journal of Business Research, Vol. 122, pp853 – 863.

[5] Kalyani et al view M2M as an application of IoT which utilizes sensors to enable communication between devices of same type. In other words, the IoT's functionality is aided by M2M via merger of wireless technologies and smart sensors. See Vijay Laxmi Kalyani et al, (2015) 'IoT: 'Machine to Machine' Application A Future Vision' Journal of Management Engineering and Information Technology, Vol. 2 No. 4, p15. Chen emphatically says, IoT is also known as M2M. See Yen-Kuang Chen, (2012) 'Challenges and Opportunities of Internet of Things' 7th Asia and South Pacific Design Automation Conference, pp. 383-388, doi: 10.1109/ASPDAC.2012.6164978.

[6] This is the integration of cloud computing and IoT. See D. Vaishnavi, (2018) 'Towards Cloud of Things from Internet of Things' International Journal of Engineering & Technology, Vol. 7 No. 4, p112-116.

[7] Proposed as 'a radically new human-centric approach to Internet data and knowledge management' with emphatic focus of the users of the paradigm as opposed to the orthodox IoT that is quite distant from humans but more fixated on the inanimate players in the network. See Marco Conti and Andrea Passarella, (2018) 'The Internet of People: A Human and Data-Centric Paradigm For the Next Generation Internet' Computer Communications, Vol. 131, p51 – 65.

[8] This ‘invention’ enables ‘physical devices to connect to the Internet as well as provide their services as a resource on the web’ with the principal purpose of linking physical objects to the web. See Muhammad Rehan Faheem, Tayyaba Anees, & Muzammil Hussain, (2019) ‘The Web of Things: Findability Taxonomy and Challenges’ IEEE Access, 1.

[9] Alem Colakovic and Mesud Hadzialic, (2018) ‘Internet of Things (IoT). A Review of Enabling Technologies, Challenges and Open Research Issues’ (2018) Computer Networks, Vol. 114, pp17-39.

[10] Global Information Infrastructure (2021) Internet Protocol Aspects and Next Generation Networks. Next Generation Networks- Frameworks and Functional Architecture Models: Overview of Internet of Things ITU- Recommendation Y. 2060 server Y.

[11] Colakovic and Hadzialic (n 9) pp17-39.

[12] A. Al-Fuqaha, Mohsen Guizani, Mohammed Aledhari and Moussa Ayyash, (2015) ‘Internet of Things: A survey on Enabling Technologies Protocols and Applications’ IEEE Communication Surveys &Tutorials, Vol. 17 No.4, 2347.

[13] See Yen-Kuang Chen, (2012) ‘Challenges and Opportunities of Internet of Things’ 7th Asia and South Pacific Design Automation Conference, pp. 383-388, doi: 10.1109/ASPDAC.2012.6164978.

[14] Andrew Whitmore, (2015) ‘The Internet of Things - A Survey of Topics and Trends’ Information System Frontiers, Vol. 17, p261-274.

[15] CASAGRAS Partnership (2009) Final Report: RFID and the inclusive Model for Internet of Things: EU Project (216803) European Commission, London UK.

[16] Michael Weyrich and Christof Ebert, (2000) ‘Reference Architecture for the Internet of Things’ IEEE software, 1.

[17] Sasu Tarkoma and Artem Katasanov, ‘Internet of Things Strategic Research Agenda (IoT- SRA) Finish Strategic Centre for Science, 1.

[18] Theo Lynn et al, (2000) ‘The Internet of Things: Definition, Keep concepts and Reference Architectures’ in Theo Lynn, John G. Mooney, Brain Lee and Patricia Takako Endo, (eds) *The Cloud to thing Continuum, Opportunities and Challenges in Cloud, Fog and Edge Computing,* Palgrave Macmillan, 1.

[19] Luigi Atzori, (2010) ‘The Internet of Things: A Survey’ Computer Networks, Vol. 54, p2787-2805.

[20] Sachin Kumar et al, ‘Internet of Things is a Revolutionary Approach for Future Technology Enhancement: A Review, Journal of Big Data, Vol. 6, No. 111, p1.

[21] Somayya Madakam et al, (2015) ‘Internet of Things (IoT): A literature Review’ (2015) Journal of Computer and Communications, Vol. 3, 164-173.

[22] Owais Ahmed, (2019) ‘Internet of Things (IoT) A Review’ International Journal of Research in Engineering Application & Management, Vol. 4, No. 10, p2454.

[23] Mohd Muntjir et al, (2017) ‘An Analysis of Internet of Things (IoT): Novel Architectures, Modern Applications, Security Aspects and Future (2017) IJERT, Vol. 6, No.6, p422.

[24] Jaimon T. Kelly, (2020) ‘The Internet of Things: Impact and Implications for Health care J Med Internet Res. Vol. 22, No. 11, 1.

[25] Kenyur Patel and Sunil M. Patel, (2016) ‘Internet of Things – IoT: Definition, Characteristics, Architecture Enabling Technologies, Application and Future challenges’ IJESC, Vol. 6, No. 5, 6122.

[26] R. Nandhini, R. Aparna and P. Srilakshmi, (2018) ‘Study on Security Issues in Internet of Things’ International Conference on Social Impact of Internet of Things (IoT), p130.

[27] I.C.L. Ng and S.Y.L. Wakenshaw, (2017) ‘The Internet-of-Things: Review and Research Directions’ International Journal of Research in Marketing, Vol. 34, No.1, p3-21.

[28] Poornima Chanal and Mahabaleshwar S. Kakkasageri, (2021) ‘Preserving Data Confidentiality in the Internet of Things’ (2021) SN Computer Science, Vol. 2, p53.

[29] Charith Perera, ‘(2006] Privacy-by-Design Framework for Assessing Internet of Things Applications and Platforms’ < https://www.researchgate.net/publication/307967586_Privacy-by-Design_Framework_for_Assessing_Internet_of_Things_Applications_and_Platforms/link/5a42776eaca272d29458fe8e/download> accessed 2 August 2021.

[30] Aimad Karkouch et al, [2015] Data Quality Enhancement in Internet of Things Environment’ IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA).

[31] Vijay Laxmi Kalyani et al, [2015] ‘IoT: 'Machine to Machine' Application A Future Vision’ Journal of Management Engineering and Information Technology, Vol. 2, No. 4, p15.

[32] D. Vaishnavi, (2018) ‘Towards Cloud of Things from Internet of Things’ International Journal of Engineering & Technology, Vol 7, No.4, p112-116.

[33] Lo'ai Tawalbeh, (2020) 'IoT Privacy and Security: Challenges and Solutions' Applied Sciences, Vol 10, p1.

[34] Jan H. Ziegeldorf et al, (2014) 'Privacy in the Internet of Things: Threats and Challenges' Security and Communication Networks, Vol. 7, No.12, p2728.

[35] Baoan Li, et al, (2011) 'Research and application on the smart home based on component technologies and Internet of Things' Procedia Engineering, Vol. 15, p2087.

[36] In this context, privacy paradox refers to the 'documented fact that users have a tendency towards privacy-compromising behavior online which eventually results in a dichotomy between privacy attitudes and actual behavior' See Susanne Barth, (2017) 'The Privacy Paradox – Investigating Discrepancies Between Expressed Privacy Concerns and Actual Online Behaviour – Asystematic Literature Review' Telematics and Informatics, Vol. 34, p1038.

[37] Jyotsna Gaghane et al, (2011) 'Smart Homes System Internet of Things: Issues, Solutions and Recent Research Directors' International Research Journal of Engineering and Technology, Vol. 4, No.5, 1965.

[38] Baoan Li, (n. 35) p1.

[39] Frances K. Adrich, (2003) 'Smart Homes: Past Present and Future' in R. Harper (ed) *Inside Smart Homes*, Springer, London, pp17-39

[40] Nadine Guhr, et al, (2020) 'Privacy Concerns in the Smart Home Context' (2020) SN Applied Sciences, Vol. 2, No. 247, 1.

[41] Dawei Jiang, et al (2021) 'Research on Data Security and Privacy Protection of Wearable Equipment in Healthcare' Journal of Healthcare Engineering, Vol. 2021, 1.

[42] Ibid.

[43] Wolfgang Gruel et al, (2016) 'Assessing the Long-Term Effects of Autonomous Vehicles: A Speculative Approach' Transportation Research Procedia, Vol. 13, 18 - 29

[44] Rushit Dave et al, (2019) 'Efficient Data Privacy and Security in Autonomous Cars' Journal of Computer Science and Application, Vol. 7, No.11, p31-36; T.K. Chan, CS Chin, 'Review of Autonomous Intelligent Vehicle for Urban Driving and Parking' (2021) 10(9) Electronics, 1021, T. K. Chan, et al, 'A Comprehensive Review of Driver Behaviour Analysis Utilizing Smartphones' (2020) 21(10) IEEE Transations on Intelligent Transportation System, 4444-4475.

[45] In this article information security is used interchangeably with data security.

[46] Porras (n 3).

[47] Rossouw von Solms and Johan van Nickerk, (2013) 'From Information Security to Cybersecurity' (2013) Computers & Security, Vol. 38, p97-102.

[48] M.E. Whitman & H.J. Mattord, (2009) *Principles of Information Security,* 3rd ed. Thompson Course Tech., 8.

[49] GDPR, recital 39, 49, 75, 83, 85 article 5(1)(f).

[50] Poornima Chanal and Mahabaleshwar S. Kakkasageri, (2021) 'Preserving Data Confidentiality in the Internet of Things' SN Computer Science, Vol. 2, p53.

[51] See Tri Ngo Minh, 'Confidentiality and integrity for IoT/Mobile Networks' (2019) < https://www.intechopen.com/chapters/68117> accessed 15 September 2021.

[52] Fred H. Cate, Peter Cullen, and Victor Mayer-Schonberger, (2014) 'Data Protection Principles for the 21st Century, Revising the 1980 OECD Guidelines' < https://www.oii.ox.ac.uk/archive/downloads/publications/Data_Protection_Principles_for_the_21st_Century.pdf > accessed 6 May 2021.

[53] Aimad Karkouch et al, (2015) Data Quality Enhancement in Internet of Things Environment' IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA).

[54] GDPR, art. 15(1).

[55] Nina Olinder, et al, (2020) 'Personal Data Protection in the Internet of Things' Advances in Economics, Business and Management Research, Vol. 171, p227.

[56] Carsten Maple, (2017) 'Security and Privacy in the Internet of Things' Journal of Cyber Policy, Vol. 2, No. 2, p154.

[57] James Sterhenz et al, (2010) 'Resilience and Survivability in Communication Network, Strategies, Principles and Survey of Disciplines' Computer Networks, Vol. 54, No. 8, p1245.

[58] Maple (n 55).

[59] GDPR, art. 4(11).

[60] T. van der Geest et al, (2005) 'Informed Consent to Address Trust, Control and Privacy Concerns in User Profiling' Privacy Enhanced Personalization, 23-34.

[61] S. Sicari et al, (2015) 'Security, Privacy, and Trust in Internet of Things: The Road Ahead' Computer Networks, Vol. 76, p146-164.

[62] Ziegeldorf (n 34).

[63] Ibid.

[64] W. H. Hassan, (2019) 'Current Research on Internet of Things (IoT) Security: A Survey' Computer Networks, Vol. 148, p28

[65] Christoper Kuner et al, (2020) *General Data Protection Regulation (GDPR). A Commentary,* Oxford University Press, London, p2.

[66] GDPR, art. 2 and 3 provide for material and territorial scopes of the regulation.

[67] See GDPR articles 4(7), 26, 4(8), 4(4) and 4(10) respectfully.

[68] These consist of the device, IoT area network, gateway, access network is network, IoT platform and IoT application server. See International Telecommunication Union, 'Requirement of the network for the Internet of Things' (2016) < https://www.itu.int/rec/T-REC-Y.4113/en> accessed 15 September 2021.

[69] Jiahong chen et al, (2020) 'Who is responsible for data processing in smart houses? Reconsidering joint controllership and the household exemption' International Data Privacy Law, Vol. 10, No.4, p279.

[70] Suada Hadzovic et al, 'Identification of IoT Actors' (2021) 21 Sensors, 2093.

[71] Julia B, Earp, (2005) 'Examining Internet Privacy Policies within the Context of User Privacy Values' IEEE Transactions on Engineering Management, Vol. 52, No.2, p227.

[72] Lee Bygrave, (2017) 'Data Protection by Design and by Default: Deciphering the EU's Legislative Requirements' Oslo Law Review, Vol 4, No.2, p106, Woodrow Hartzog, (2018) *Privacy Blueprint: The Battle to Control the Design of New Technologies*, Harvard University Press.

[73] See Article 29 Data Protection Working Party. 'Working Document on Data Protection Issues delated to RFID Technology' (2005) < https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2005/wp105_en.pdf> accessed 15 September 2021.

[74] Aurelia Tamo – Larrieux, *Designing for Privacy and its Legal Framework, Data Protection by Design and Default for Internet of Things*, Springer Nature, Switzerland, 2018.

AUTHOR

Olumide Babalola, the author of Casebook on Data Protection (Nigeria's first and only law textbook on data protection) is a prolific and consummate digital rights, consumer rights, privacy and data protection lawyer. His rich and diverse digital rights litigation experience spans across all superior courts of records in Nigeria and regional courts in Africa including ECOWAS Community Court of Justice. He has specifically litigated on Privacy and Data Protection, Cybercrime, Hate Speech, Freedom of Information, Online Freedom of Expression, passage of laws protecting digital rights among others. Olumide is a seasoned Conference speaker at local and international fora. In 2019, he spoke at the RightsCon (The 8th Annual Summit on Human Rights in the Digital Age) held in Tunis and UN Internet Governance Forum in Berlin, 2019, among others.

Olumide has five published books to his credit: the first is a historical piece on the office of the attorney general of the federation and its occupants in Nigeria; the second being a casebook on Labour and employment law - which work was propelled by the volume of legal opinions (on Nigerian Labour regime especially the decisions of the courts on the peculiar issues) he had to write for his multi-national company on regular basis while the third is another casebook on corporate law and practice; Babalola's Law Dictionary, is reputed as Nigeria's first law dictionary (strictly so called) and his latest being a Casebook on Data Protection.

Olumide is the managing partner of Olumide Babalola, LP - his flagship full-service law office with particular bias for digital rights, consumer rights litigation, class actions, employment and corporate commercial litigation et al. The awardee of the "Nigerian Rising Star Award" is a member of the Nigerian Bar Association, Secretary of NBA Lagos Human Rights Committee, British Nigeria Law Forum, Internet Society, Internet Governance Forum Support Association (IGFSA), Chartered Institute of Arbitrators (UK), World Litigation Forum, International Bar Association, International Association of Privacy Professionals and International Network of Privacy Law Practitioners. (INPLP)

10-BIT, 1GS/S TIME-INTERLEAVED SAR ADC

Shravan Kumar Donthula[1] and Supravat Debnath[2]

[1]Department of Electrical Engineering,
Indian Institute of Technology, Hyderabad, India
[2]Integrated Sensor Systems, Centre for Interdisciplinary Programs,
Indian Institute of Technology, Hyderabad, India

ABSTRACT

This paper describes the implementation of a 4-channel, 10-bit, 1 GS/s time-interleaved analog to digital converter (TI-ADC) in 65nm CMOS technology. Each channel consists of interleaved T/H and ADC array operating at 250 MS/s, with each ADC array containing 14 time-interleaved sub-ADCs. This configuration provides high sampling rate even though each sub-ADC works at a moderate sampling rate. We have selected 10-bit successive approximation ADC (SAR ADC) as a sub-ADC, since this architecture is most suitable for low power and medium resolution. SAR ADC works on binary search algorithm, since it resolves 1-bit at a time. The target sampling rate was 20 MS/s in this design, however the sampling rate achieved is 15 MS/s. As a result, the 10-bit SAR ADC operates at 15 MS/s with power consumption of 560 μW at 1.2 V supply and achieves SNDR of 57 dB (i.e. ENOB 9.2 bits) near nyquist rate input. The resulting Figure of Merit (FoM) is 63.5 fJ/step. The achieved DNL and INL is +0.85\-0.9 LSB and +1\-1.1 LSB respectively. The 10-bit SAR ADC occupies an active area of 300 μm × 440 μm. The functionality of single channel TI-SAR ADC has been verified by simulation with input signal frequency of 33.2 MHz and clock frequency of 250 MHz. The desired SNDR of 59.3 dB has been achieved with power consumption of 11.6 mW. This results in a FoM value of 60 fJ/step.

KEYWORDS

ADC, SAR, TI-ADC, LSB, MSB, T/H, SCDAC, CDAC, SFDR, SINAD, SNR, TG, EOC, D-FF, MIM, MOM, DNL, INL.

1. INTRODUCTION

Time-interleaved analog to digital converters (TI-ADC) use a parallel combination of multiple sub-ADCs into a single ADC, that can improve the sampling rate compared to the sub-ADCs alone. The major advantage of time-interleaved ADC is that it can improve the sampling rate in a given technology. In wide band applications like wireless communication, serial data links require A/D converters with high sampling rate to meet the channel bandwidth and power requirements. Existing high-speed ADCs require high power to meet the required sampling rate. However, many applications demand for high sampling rates while keeping the power consumption relatively low. In these cases, low powered sub-ADCs operating at lower sampling rates can be used with interleaving to achieve the required sampling rate. In the proposed design we have selected SAR ADC architecture for sub-ADCs, since this architecture is most suitable for low power and moderate sampling rate. The major design challenge for time-interleaved ADCs is matching the performance of the sub-ADCs used with respect to bandwidth, timing, offset and gain. Mismatch in the performance of sub-ADCs degrades its major specifications like SFDR, SNDR etc. required for wireless communications. The mismatch in sub-ADCs can be

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 321-339, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112325

detected and corrected by using calibration methods either in analog or digital domain. This correction can remove the problems due to mismatch to a certain extent [6] [8]. The proposed design adds redundant sub-ADCs in each ADC array (each channel) to enable background calibration. This paper describes the design procedure for a 10-bit 1 GS/s TI-ADC and implementation of a 10-bit SAR ADC. The TI-ADCs consist of a total of 4 × 14 i.e. 56 time-interleaved SAR ADCs, with each SAR ADC working at 20 MS/s (250M/13 ~ 20 MS/s) sampling rate to meet the overall sampling rate of 1 GS/s. Each SAR ADC consists of comparator, DAC and SAR logic. The rest of the paper is organized as follows: Section-2 describes the 4-Channel TI-ADC Architecture. Section-3 describes the Implementation of 10-bit 20 MS/s SAR ADC. Section-4 provides layout design and post layout simulation results. Section-5 describes the implementation of single channel TI-SAR ADC. Finally, the conclusions and future work are given in Section-5.

2. 4-Channel TI-ADC Architecture

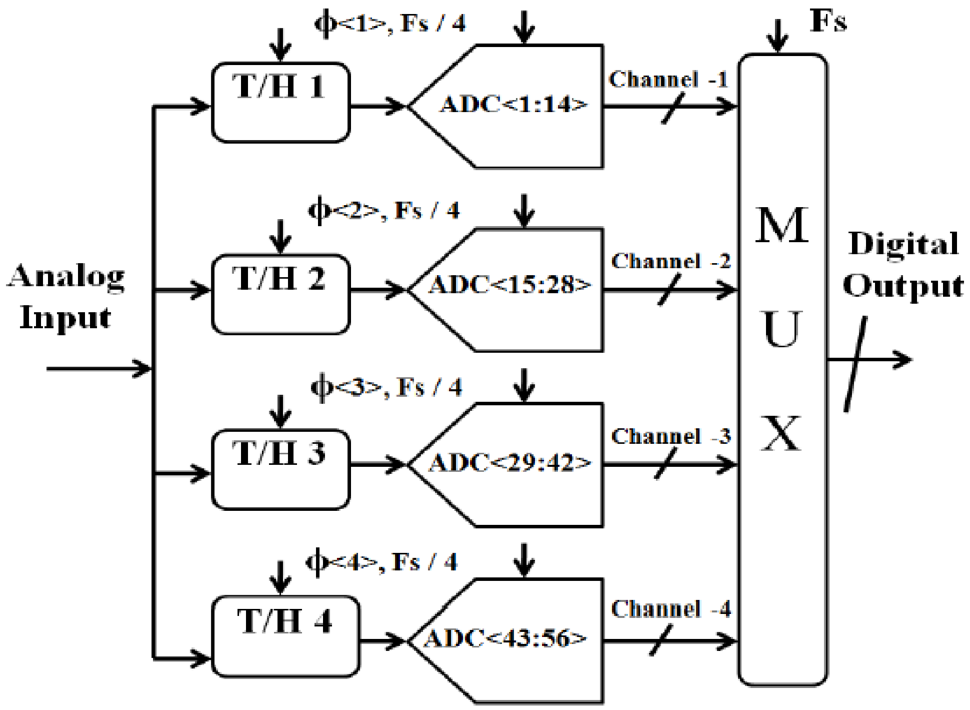


Figure 1. Architecture of 4-Channel TI-ADC.

Fig. 1 shows the architecture of the 4-channel TI-ADC, each channel consists of time-interleaved T/H and ADC array, each ADC array uses 14 time-interleaved 10-bit SAR ADCs. The 4-channels are driven by clock frequency of Fs/4, with equally spaced phases, ϕ< 1 > to ϕ< 4 >. In a TI-ADC with sampling rate (Fs) of 1 GS/s, each channel should operate at Fs/4 = 250 MS/s with equally spaced phases at ϕ < i > = 2π (i - 1)/4 where i = 1, 2, 3 etc. (0°, 90 °, 180 °, 270 °). Sampling rate of each SAR ADC should be 20 MS/s. The additional 10-bit SAR ADC used in each channel is for background calibration. Fig. 2 shows the timing diagram of 4-channel TI-ADC at each rising edge of master clock (Fs). The track and hold unit of each channel spends two clock cycles in track mode and two clock cycles in hold mode. This choice of timing presents a trade-off between input bandwidth and accurate sampling of the input. Since the operation of each channel is shifted with respect to the previous one by one clock cycle, at any given time, only two channels are in track mode. This implies that at any moment of time N=2 ('N' represents number of channels) sampling capacitors are connected to the input. While this limits the bandwidth [5], it gives enough settling time for sampling the data (two clock cycles) to the T/H unit. When the first T/H goes from track mode to hold mode it has acquired a sample of the analog input over two clock periods.

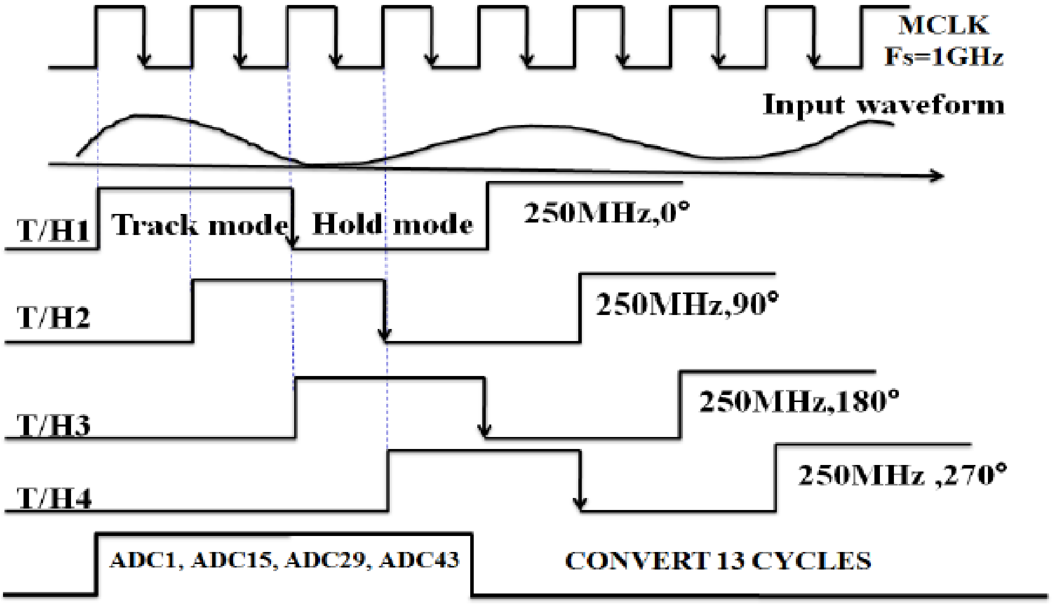


Figure 2. Timing diagram of 4-channel TI-ADC.

It is then converted to digital domain by the corresponding sub-ADCs in each channel. The master clock (Fs) works at 1 GHz while each track/hold and sub-ADC works at 250 MHz clock frequency with appropriate phase shift.

2.1. Limitations of Time-Interleaved ADC

The optimum number of channels are chosen based on the required sampling rate. To achieve high sampling rate, increasing the number of channels is not desirable because of offset, bandwidth, timing and gain error mismatches between the channels. Mismatch in the channels degrades the performance of time-interleaved ADC. Corrections for these mismatches can be implemented to a certain extent using calibration in either digital or analog domain. However, the calibration process in time-interleaved ADCs is constrained by the number of channels.

3. IMPLEMENTATION OF 20MS/S 10-BIT SAR ADC

A Successive approximation register (SAR) ADC works on binary search algorithm principle. Fig. 3 shows the block diagram of fully differential 10-bit SAR ADC. It consists of comparator, capacitive DAC (CDAC) & SAR logic. Binary search algorithm proceeds along the following steps to convert the analog input into its equivalent digital output [13]. The first step is to sample the differential inputs during sampling phase using CDAC. Next step is to set the 10-bit register in SAR logic to mid code (i.e., 1000...00, here MSB is '1'), this forces the 10-bit CDAC to V_{pref} /2 where V_{pref} is positive reference voltage to the CDAC. The comparator compares the differential DAC outputs (i.e., $V_p = - V_{inp} + V_{pref} /2$ & $V_n = -V_{inn}+V_{pref} /2$). If the $V_n > V_p$ (i.e., $V_{inp} > V_{inn}$) then MSB will remain at active high otherwise it will be forced to active low. After resolving MSB bit, the SAR logic moves to next bit, it forces that bit active high and does another comparison, this process is repeated for 10-bits till it reaches to least significant bit (LSB). Then 10-bit digital output code is available at end of conversion (EOC).

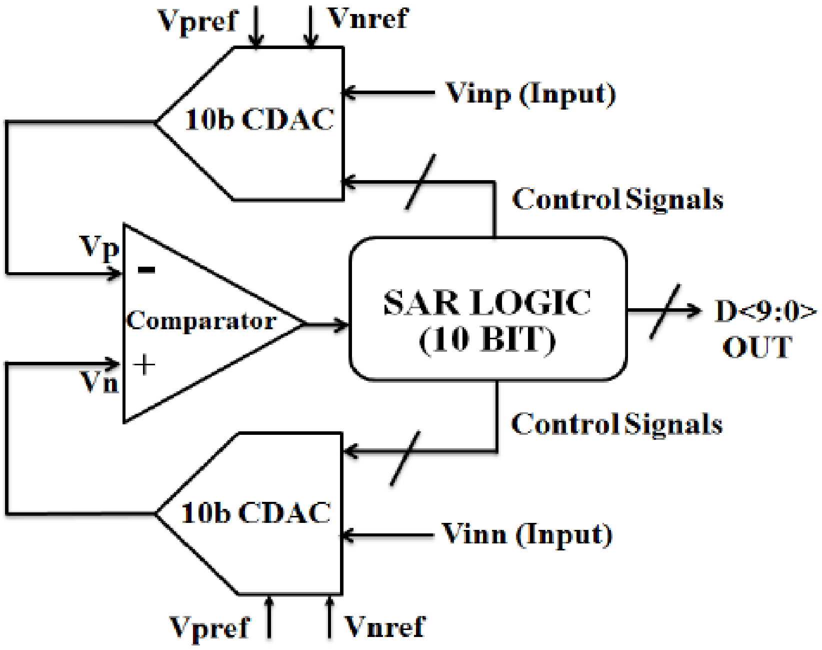


Figure 3. Block diagram of 10-bit SAR ADC

Since, SAR ADC resolve the input one bit at a time, the complexity and power consumption is lower at expense of reduced sampling rate. 10-bit SAR ADC requires the 10 + 3 clock cycles to get the digital output for given analog input inclusive of two cycles for sampling the input with comparator offset calibration before the conversion takes place and another cycle for data latch after the conversion. This cycle signal is called end of conversion (EOC).

3.1. 10-bit Fully Differential Capacitive DAC

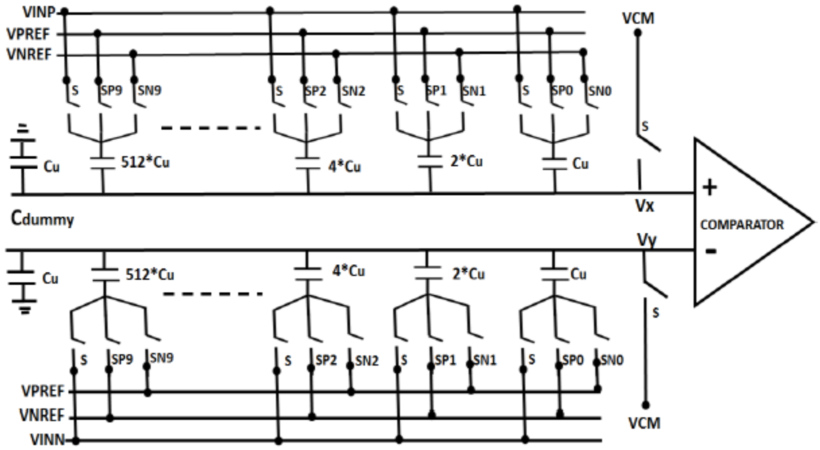


Figure 4. Architecture 10-bit fully differential CDAC

Fig. 4 shows architecture of 10-bit fully differential capacitive DAC (CDAC). It performs both sampling (S/H) operation as well as DAC operation. CDAC switching mechanism is controlled with SAR logic based on the comparator decision. In 10-bit CDAC, the capacitor network is in binary weighted fashion with dummy capacitor, Where C_u is a unit capacitor. For simplicity, the explanation given here is for a single sided CDAC. The same is applicable for a fully differential CDAC. During the sampling phase (S), the top plates of the capacitors in network are connected to common mode voltage (V_{cm} = 0.6 V) through a switch and bottom plates are charged to the input voltage (V_{inp}). The total charge on the capacitor during sampling phase is calculated by $Q_{tot} = 2^n * C_u (V_{cm} - V_{inp})$ for an n-bit CDAC. During the next cycle, the top plate switch is opened before the SAR logic sets the MSB bit to high ('1') thus the bottom plate of the MSB capacitor is now connected to V_{pref} (1.2 V). The remaining capacitors are connected to V_{nref} (0 V). Since the top plate is connected to a high input impedance of the comparator, the total charge on capacitors remains the same [14]. However, switching causes the charge to redistribute resulting in the top plate potential going to V_x. The total charge on top plate capacitor is given by

$$Q_{tot} = Q_{MSB} + ... + Q_{LSB} + Q_{dummy} \quad (1)$$

$$2^n C_u (V_{cm} - V_{inp}) = 2^{n-1} C_u (V_x - V_{pref}) + 2^{n-1} C_u (V_x) \quad (2)$$

$$V_x = - V_{inp} + V_{cm} + V_{pref} /2 \quad (3)$$

If $V_x < V_{cm}$, it means $V_{inp} > V_{pref} / 2$ and the most significant bit (MSB) value is set to high, otherwise it is reset to low ('0'). In next cycle the second MSB bit is set to high, which results in a comparison with either $3V_{pref} /4$ or $V_{pref} /4$, depending on the value of MSB bit, this process continues till the CDAC converges to V_{cm} (common mode voltage) and the LSB is determined. The major advantage of capacitive DAC is that it is less sensitive to parasitic effects across the capacitor array, which improves linearity (monotonic) at the DAC outputs.

3.2. Selection of Unit Capacitor (C_u) and Switch

The value and type of unit capacitor is chosen based on area and mismatch data given in UMC65nm CMOS technology. This technology has two different types of capacitors (i) MOM

Capacitor (ii) MIM Capacitor. MIM capacitor is more accurate compared to MOM capacitor, but it occupies more area. The value of unit capacitor is limited by thermal and quantization noise (i.e., the sum of rms thermal noise and rms quantization noise to be less than LSB/2). $KT/C_{total} + LSB^2/12 < (LSB/2)^2$, where LSB = $1.2/2^{10}$ = 1.17 mV, therefore C_{total} should be greater than 100 fF. Also, the total capacitance for 10-bit CDAC is $2^{10} * C_u = C_{total}$, where C_u unit capacitance. This gives a unit capacitance of $C_u > 100$ aF [15]. Choosing a lower value capacitance reduces the switching power and area but mismatch between capacitor arrays become very high which leads to non-linearity. In this design considering the area constraint, switching power and mismatch factors, the chosen unit capacitance (C_u) value is 10 fF which occupies an area of 4 µm * 4 µm. Transmission gates (TG) were used as switches to meet the settling error within LSB/2, given the settling time (800 pS) during the sampling phase. Considering the unit capacitance of 10 fF, the total capacitance during the sampling phase, $C_{hold}=2^{10} * 10$ fF =10.24 pF. The settling time of DAC output is 4 ns (5* $R_{on} * C_{hold}$ = 4 ns) with clock frequency of 250 MHz) so, the ON-resistance of switch should be less than 78 Ω (i.e., Ron <78 Ω). Which leads to TG transistor (Both NMOS and PMOS) sized with W = 10 µm and L = 0.1 µm.

3.3. Selection of Comparator Architecture

The comparator is a key component in an analog to digital converter (ADC). Every ADC contains at least one comparator (1-bit A/D converter). It must be able to resolve very small analog voltage and convert it into a rail-to-rail digital output. This small input voltage is known as the resolution of the comparator. The basic function of the comparator is to compare the differential input signal with the reference threshold and to give a digital decision accordingly. The architecture of the comparator is chosen based on resolution, speed, offset voltage and power dissipation. A dynamic latch comparator is most widely used for high speed and low power consumption. The main disadvantage of latch type comparators is their high input offset voltage due to mismatch in transistors and also the kick back noise effect at the inputs. To reduce the input offset voltage and the effect of kick back noise, a preamplifier is used in front of the dynamic latch. The requirements from the comparator are:

- Resolution (LSB) = 1.17 mV
- Input offset voltage and input integrated noise < LSB/2
- Settling time < 800ps, considering 250 MHz as clock frequency

3.4. Architecture of Preamplifier Based Dynamic Latch Comparator

Fig. 5 shows the schematic of preamplifier based dynamic latch comparator. It consists of three stages viz preamplifier, dynamic latch and SR latch. Preamplifier is a differential amplifier with resistive load (R0, R1 = 5 KΩ) with wide bandwidth and relatively small gain to achieve high speed as shown in Fig. 5a. The dynamic latch consists of two cross-coupled CMOS inverters used for regeneration as shown in Fig. 5b. It operates in either one of the following two modes:

- Rest phase (CLK = LOW)
- Evaluation phase (CLK = HIGH)

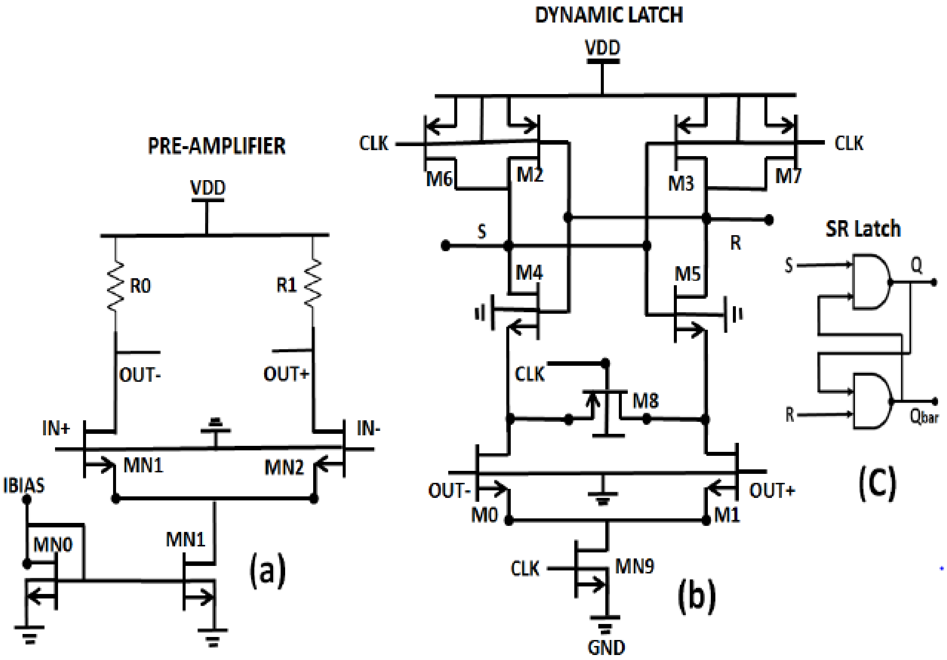


Figure 5. Schematic of preamplifier based dynamic latch

During the reset phase, the output nodes of cross coupled inverters (M2-M5) are reset to V_{DD} through the reset transistors M6 and M7. During the evaluation phase, the tail transistor MN9 is turned on. The input transistor pairs (M0, M1) start discharging at different time rates depending on the applied input voltage. This initiates positive feedback, enhancing the small differential voltage to a full swing differential output. The mismatch of transistor size in differential pair (M0, M1) and threshold voltage variation between M4, M5 transistor can lead to high input offset voltage. This offset voltage can be reduced by keeping a preamplifier in front of the dynamic latch which also prevents kick back noise to the differential inputs. The gain of the preamplifier is 19.5 dB with unity gain bandwidth (UGB) of 5 GHz, input offset voltage of ±872 µV and input integrated noise (100 Hz to 5 GHz) of 715 µV (rms).

3.5. 10-bit Successive Approximation Register Logic

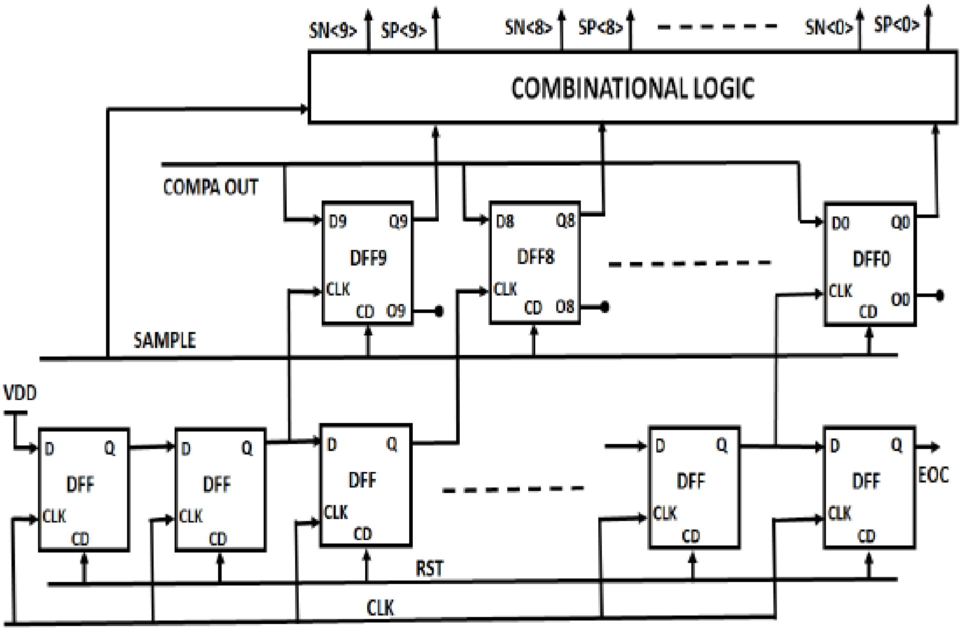


Figure 6. Block diagram of 10-bit SAR logic

There are two kinds of approaches to design SAR logic. One is a method proposed by Anderson [17] and the other is Rossi and Fucili method [18]. Fig. 6 shows 10-bit SAR logic proposed by Anderson. It is based on shift registers and combination logic. In this design, the binary search 10-bit SAR logic is implemented using this approach. The proposed 10-bit SAR logic requires at least 2*10 = 20 D-FFs. One chain of 10 D-FFs for storing the conversion results and another chain of 10 D-FFs for performing the shift operation and generation of necessary control signals (SN< 9:0 >, SP< 9:0 >) for DAC operation using combinational logic. SAR logic performs three main operations, (i) It shifts the initial mid code (i.e.,1000...00, here MSB is ‘1’) to the right by one bit (ii) It loads the result from the comparator (COMPA OUT) during the positive edge triggering of next nearest bit (iii) Finally it holds the converted bits. After 10 clocks, shift registers generate pulse called end of conversion (EOC), which means that the whole conversion is completed. The EOC also indicate the start of the next sampling.

4. LAYOUT DESIGN & POST LAYOUT SIMULATION RESULTS

4.1. Transient response of 10-bit CDAC

Fig. 7 shows the transient response of differential CDAC (V_x, V_y nodes shown in Fig. 4) with DC input voltages V_{inp} = 1.1 V and V_{inn} = 0.1 V at 10 MS/s sampling rate. The corner (typ, min, and max) and monte-carlo simulations have been carried out for differential CDAC and the

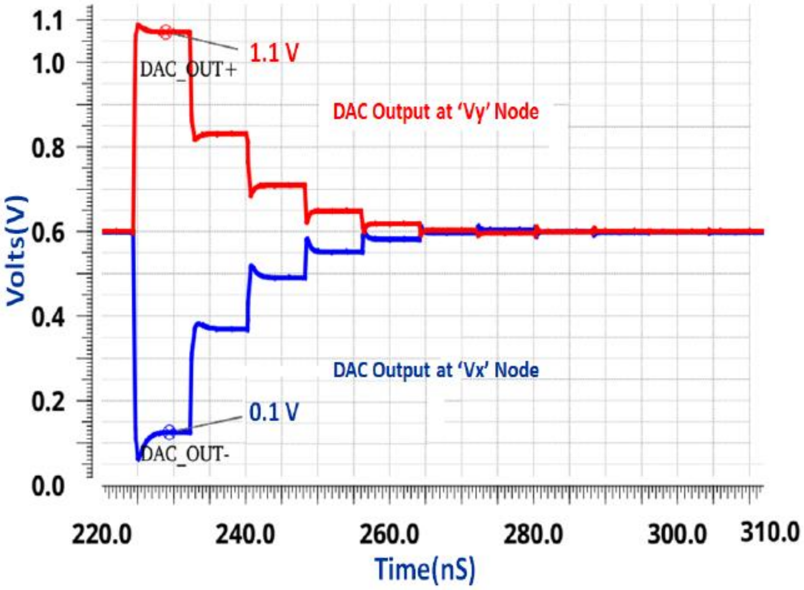


Figure 7. Transient response of 10-bit CDAC

output voltage variation of about ±900 µV at maximum input voltage (i.e., <LSB) is seen. DAC output voltages are settled within 0.5 LSB of the final value. The dynamic power (i.e., C_{total} * V_{DD}^2 * Fs) of 10-bit CDAC is 150µW with 1.2 V supply at 10 MS/s.

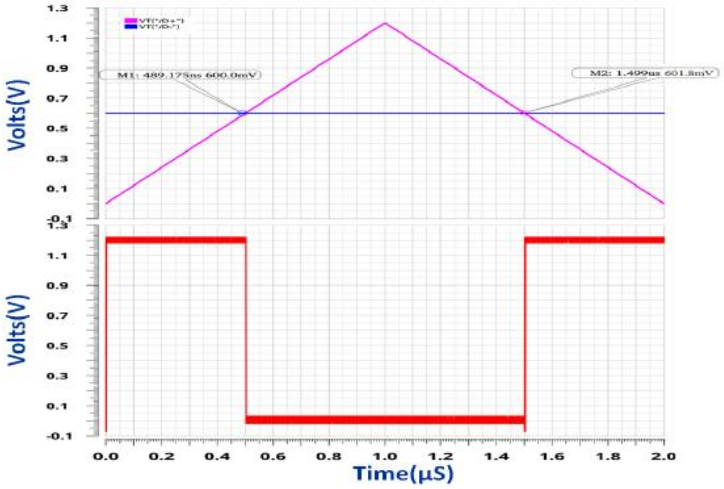


Figure 8. Transient response of dynamic comparator

Fig. 8 shows the transient response of comparator with triangular input signal and with a common mode voltage of 0.6 V. To estimate the input offset voltage of preamplifier-based latch comparator, Monte-Carlo simulations were carried out (number of runs = 50) and the comparator output transition were seen to be shifted towards positive and negative side in time which results an input offset voltage deviation of ±872 µV about the mean value respectively as shown in Fig. 9

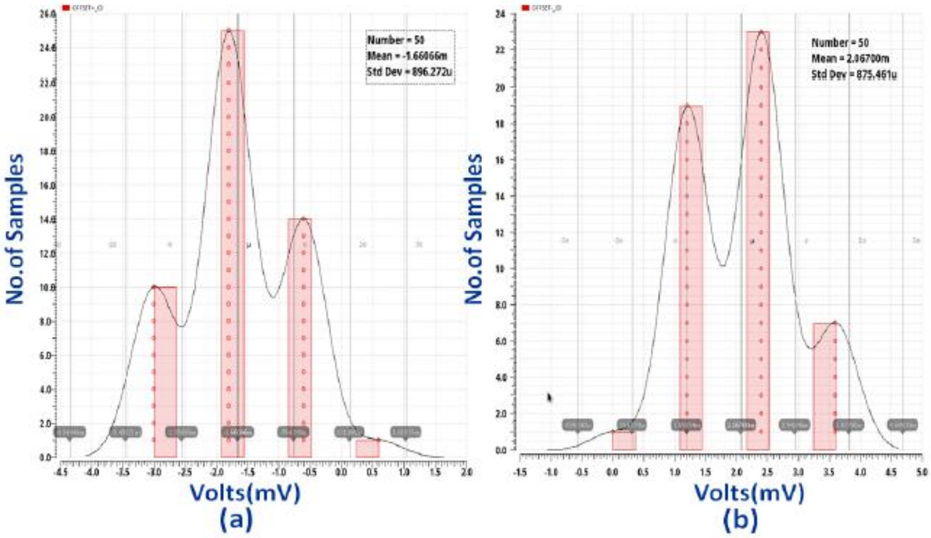


Figure 9. Preamplifier based dynamic latch comparator input offset voltage (a) at negative terminal (b) at positive terminal

4.2. Transient Response of 10-bit SAR Logic

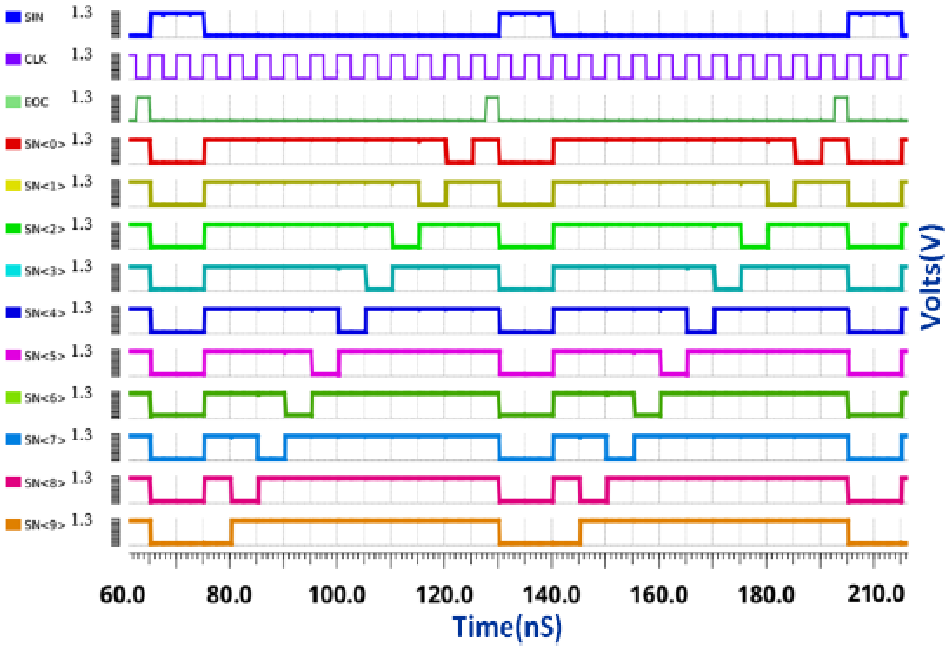


Figure 10. Transient response of 10-bit SAR logic

Fig. 10 shows the transient response of 10-bit SAR logic. The clock frequency of 250 MHz pulse input (period = 4 ns) is given to MOD-13 counter to generate the sample pulse (SIN). It repeats for every 13 cycles (i.e 13*4ns = 52 ns). 10-bit shift register used for generating the EOC pulse (T_{pulse} = 4 ns) just before the next sampling. The shift register and combination logic is used for generating control signals (SN< 0 >, SN<1 > .. SN< 9 > and SP< 0 >, SP< 1 > .. SP< 9 >) based on comparator decision (COMPA OUT).

4.2. Layout Design & Dynamic Performance of 10-bit SAR ADC

Fig. 11 shows the layout of 10-bit SAR ADC. In this design, the METAL stack with maximum metal width is used for routing critical nets like differential CDAC outputs, comparator output in a symmetric manner to avoid the output voltage mismatch, delay due to parasitic effects. All control signals which control the switching of CDAC are laid out symmetrically to meet the equal delay effect. The total area occupied by 10-bit SAR ADC is 300 µm * 440 µm.

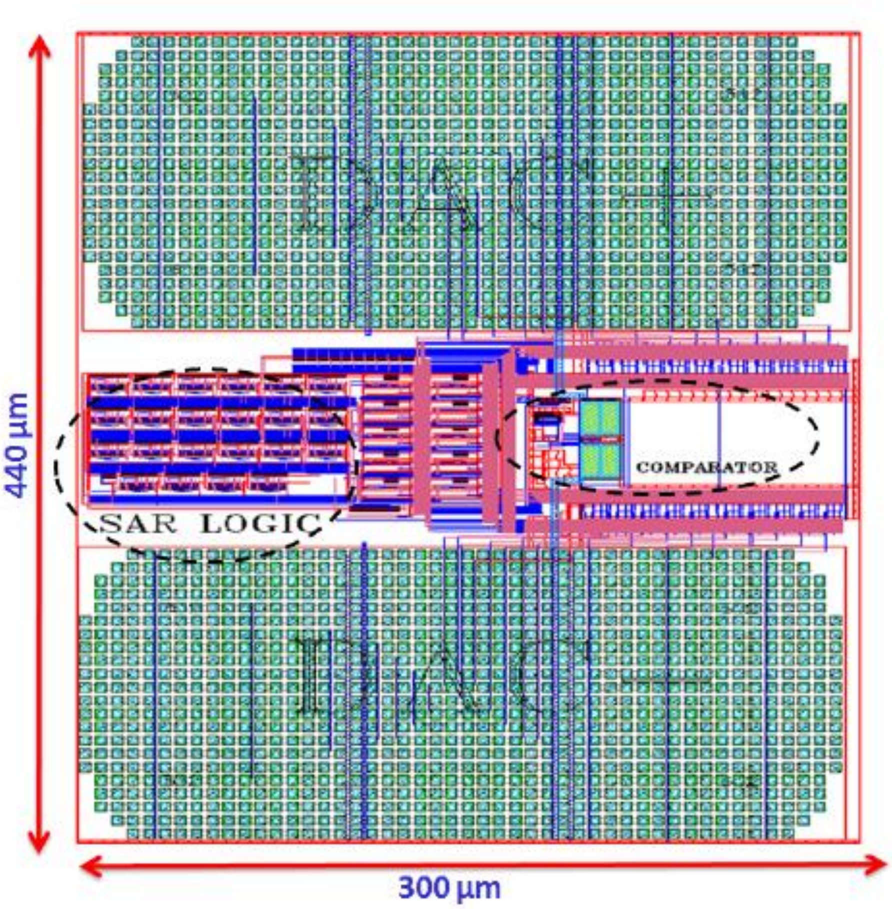
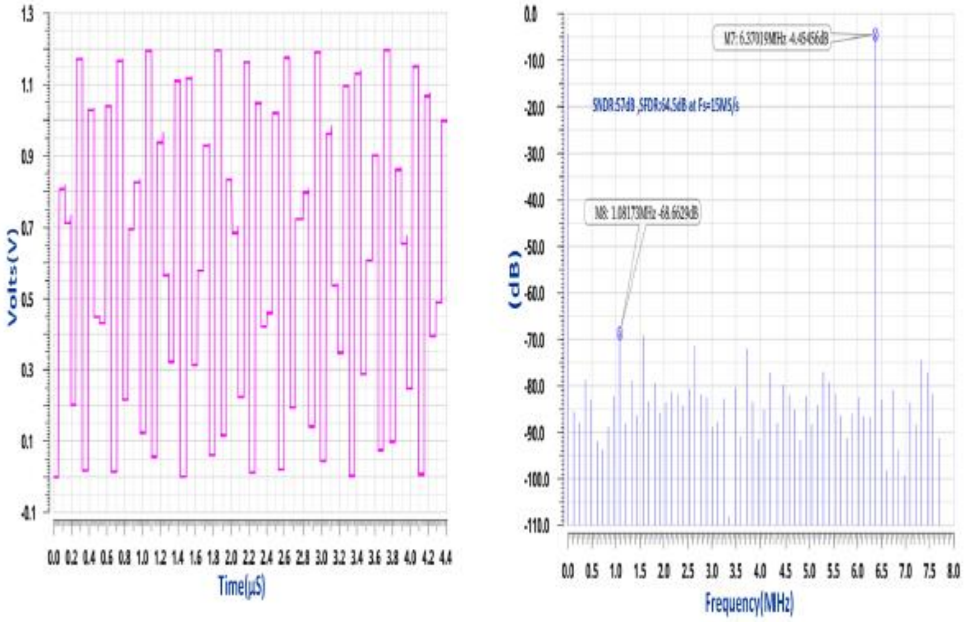


Figure 11. Layout of 10-bit SAR ADC

The coherent sampling method is used for evaluating dynamic performance of ADC as shown in Fig. 12b, the transient noise simulation is carried out to include quantization, thermal noise effects (F_{max} is chosen as maximum clock frequency given for ADC i.e., 250 MHz). The following inputs are given for evaluating the dynamic performance of 10-bit SAR ADC. V_{in} = 1 V_{p-p}, F_{in} = $F_s/2$ (i.e., nyquist rate input), F_s =15 MS/s and no. of points is 128. The achieved SFDR is 64.5 dB and SNDR of 57 dB (ENOB = 9.2 bits).



(a) (b)

Figure 12. 10-bit SAR ADC (a) Transient response (b) Output Spectrum

4.3. Static Performance of 10-bit SAR ADC

The evaluation of static performance (DNL and INL) of 10-bit SAR ADC is done by using endpoint method with ramp input. The following inputs are given for evaluating the static performance of 10-bit SAR ADC, ADC sample rate is F_s=15 MS/s or T_s = 66 ns, resolution of ADC is 1.17 m V, for LSB/4 measurement the resolution per code becomes 4 samples/code and ramp duration per code will be 264 ns (i.e., 4*66ns). So, ramp slope is 1.17 mV/0.264 µs. The achieved DNL of +0.85 \ -0.9 LSB is shown in Fig. 13a and INL of +1.061 \ -1.124 LSB is shown in Fig. 13b. The major INL and DNL jump occurs at first and second MSB transition, which are due to capacitor mismatches [12].

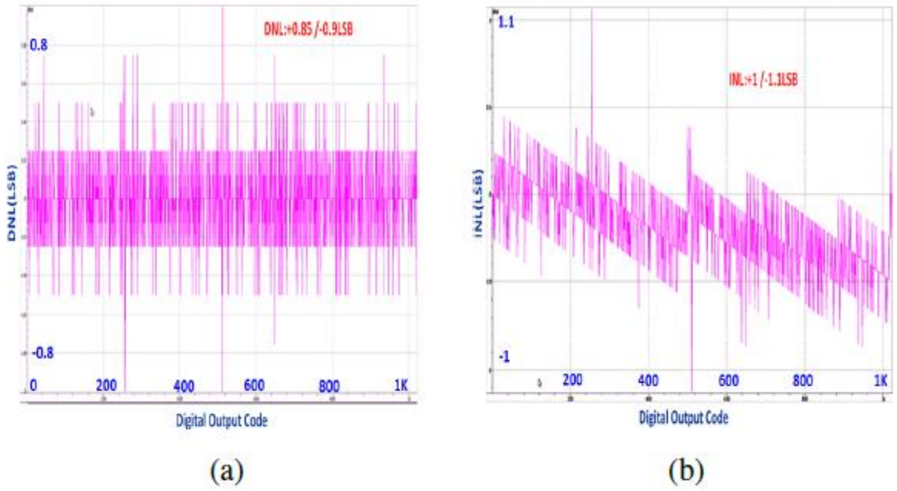


Figure 13: Static performance of 10-bit SAR ADC (a) DNL (b) INL

5. Architecture of Single Channel TI-SAR ADC

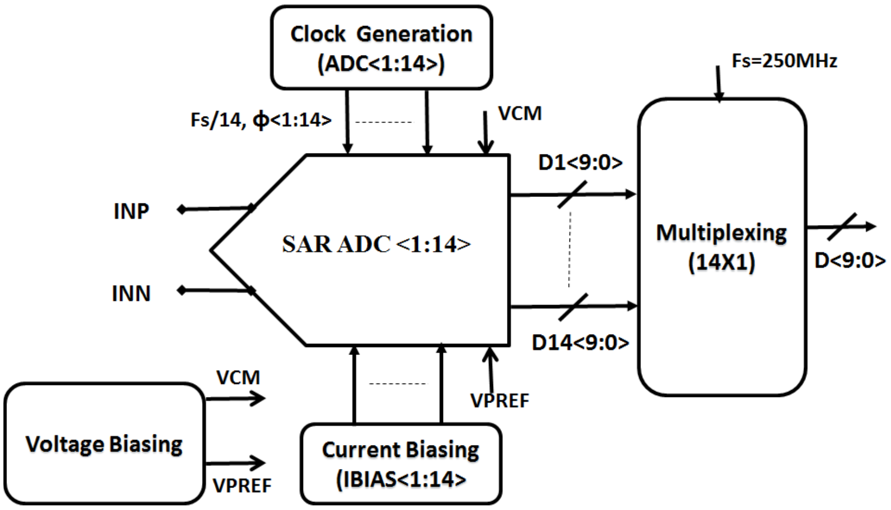


Figure 14. Block diagram of single channel 10 bit, TI-SAR

A Single Channel TI-SAR ADC architecture consist of the following sub-modules as shown in Fig. 14.

- Current Biasing
- Clock generation / Timing scheme
- Digital Multiplexing

5.1. Current Biasing

Fig. 15 shows the current basing circuit of single channel TI-SAR ADC. The two-stage differential amplifier (error amplifier) with gain of 65 dB is used to create the voltage controlled current source (voltage to current converter). In Fig. 15 shows the voltage to be converted is applied to noninverting terminal of the amplifier. The inverting terminal of the amplifier is connected in negative feedback viz resistor and transistor MN0. The output of amplifier drives the input gate of the transistor MN0. The error amplifier will force the required gate voltage such that the voltage across resistor R0 need to be V_{ref} is equal to 0.6 V, which results the reference current in R0 and MN0 will be 120 µV (i.e., V_{ref} / R0). Further this reference current is pass through PMOS current mirror (M0 to M14) and mirror current is scaled down to 50 µA by sizing the PMOS transistor, finally the bias currents are (IBIAS< 1 > to IBIAS< 14 >) fed to comparators of 14 TI-SAR ADC.

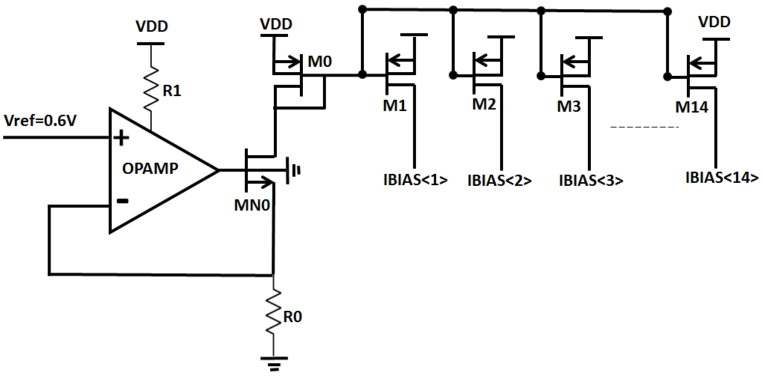


Figure 15. Schematic of voltage to current converter

5.2. Timing Scheme/Clock Generation

Fig. 16 shows the timing scheme of single channel time-interleaved SAR ADC. It is based on shift register. In this design, the timing signals (ADC< 1 > to ADC< 14 >) for operation of TI-SAR ADC are generated by cascading the DFFs, so that the output from the previous flip-flop becomes input to the next flip-flop. While performing shift operation the timing signals are shifted by one clock period, where the clock (CLK) operates at 250 MHz (i.e., period of 4 ns) and the reference sample pulse is generated by MOD-14 counter, which repeats for every 14 clock cycles.

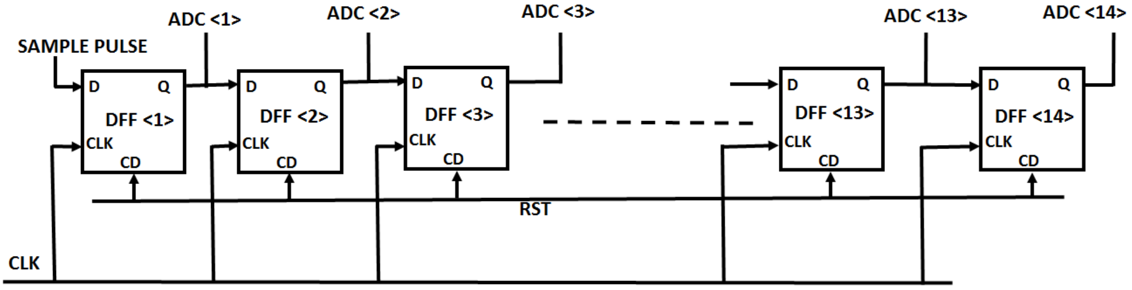


Figure 16. Block diagram of timing scheme

Fig. 17 shows the transient response of timing scheme for 14 time-interleaved SAR ADC, here the CLK operates at 250 MHz and reference sample pulse (SAMPLE PULSE) which repeats for every 14 clock cycles. The timing signals for operation of TI-SAR ADC (ADC< 1 > to ADC< 14 >) are shifted by one clock period without overlapping.

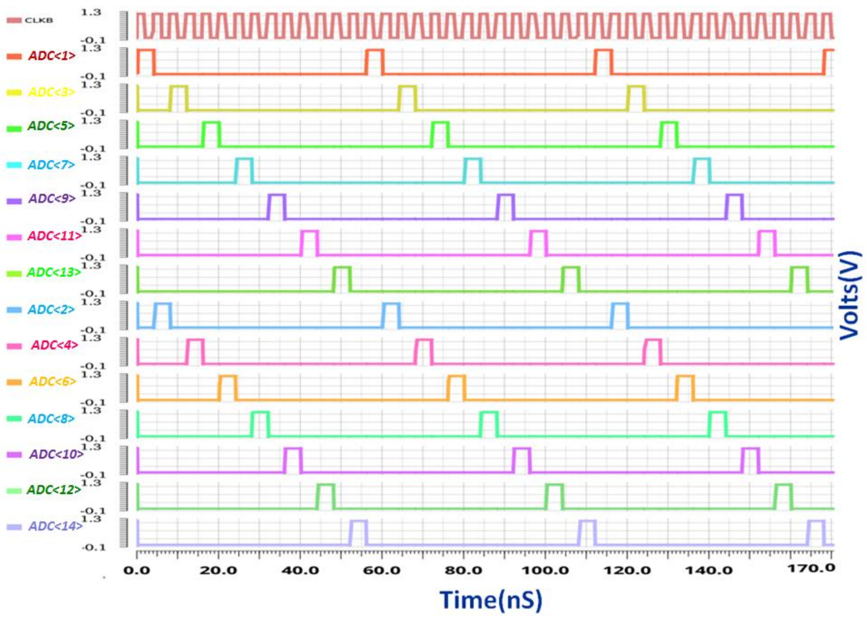


Figure 17. Transient response of timing scheme

5.3. Digital Multiplexing

Fig. 18 shows the block diagram of 14×1 digital multiplexing. In this design, the output data from each 10-bit SAR ADC (D < 1 : 10 >) is available at EOC. During rising edge of EOC the digital data from 14 TI-SAR ADC is latched into corresponding D-FFs (DFF1< 1 : 10 > to DFF14< 1 : 10 >). The latched data (D1< 1 : 10 > to D14< 1 : 10 >) is fed to corresponding 14 × 1 multiplexer and finally the multiplexed output data read in parallel combination, which is controlled by an MOD-14 counter (S < 1 : 4 >) operates at F_s = 250 MHz.

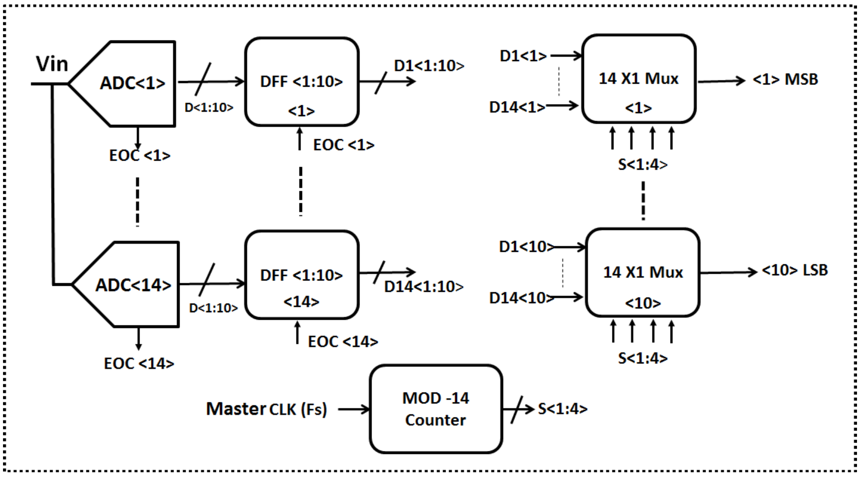


Figure 18. Block diagram of 14×1 digital multiplexing

6. SIMULATION RESULTS OF 14 TI-SAR ADC

6.1. Dynamic performance

The ideal 10-bit DAC is used to reconstruct sampled analog output voltages from the 10-bit digital outputs (D< 9 : 0 >) as shown in Fig. 19a, here D< 9 > represents the sign bit. The coherent sampling method is used for evaluating dynamic performance of TI-SAR ADC as shown in Fig. 19b. The transient noise simulation is carried out to include quantization, thermal noise effects (F_{max} is chosen as maximum clock frequency given for ADC i.e., 250 MHz). The following inputs are given for evaluating the dynamic performance of 10-bit TI-SAR ADC. V_{in} = 1 V_{p-p}, F_{in} = F_s/8 (33.6 MHz), F_s =250 MS/s and no. of sample points (N) is 128. The achieved SFDR is 66 dB and SNDR of 59.3 dB (ENOB = 9.6 bits). This results in a FoM value of 60 fJ/step.

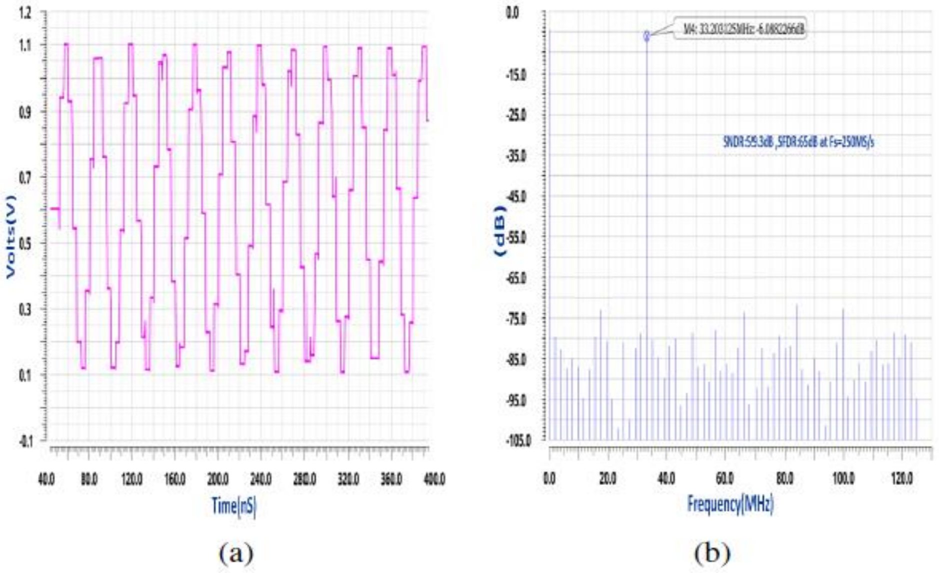


Figure 19. Single channel TI-SAR ADC (a) Transient response (b) Output Spectrum

7. PERFORMANCE SUMMARY AND COMPARISON

Table 1. Performance Summary of 10-bit SAR ADC

Parameter	**Value**
Process/Technology	UMC 65 nm CMOS Technology
Supply Voltage	1.2 V
Input Swing (differential)	2 V_{p-p}
Sampling Rate	15MS/s
Input Offset Voltage	±872 µV
SFDR	64.5 dB
SNDR	57 dB
ENOB	9.2 bits
Power consumption	560 µW
FoM	63.5 fJ/Step

Table 2. Comparison Table

	This Work	**ISSCC** [19]	**ISSCC** [20]	**VLSI** [21]	**JSSC** [22]
Architecture	SAR	SAR	SAR	SAR	SAR
Technology (nm)	65	90	65	90	65
Supply Voltage (V)	1.2	1	1	1	1.2
Sampling Rate (MS/s)	15	50	50	30	50
Resolution (bit)	10	9	10	10	10
Power (µW)	**560**	700	820	980	826
SFDR (dB)	64.5	**-**	**-**	68.16	61.8
ENOB (bits)	**9.2**	7.8	9.16	9.16	9.18
FoM (J/step)	63.5f	65f	30f	57f	27f

8. CONCLUSION AND FUTURE WORK

This paper describes about various architectures of time-interleaved ADCs, selection of number channels based on sampling rate and sub-ADC architecture. The mathematical model for 14 time-interleaved ADC has been developed to analyse the effect of gain, offset and timing mismatch among the channels. The design of 10-bit, 20 MS/s SAR ADC has been implemented in 65nm CMOS Technology. Layout design of 10-bit capacitive DAC, preamplifier based dynamic latch comparator is implemented by using common centroid method and their extracted views are included in simulations. The target sampling rate was 20 MS/s in this design, however the sampling rate achieved is 15 MS/s. As a result, the 10-bit SAR ADC operate at 15 MS/s with power consumption of 560 µW at 1.2 V supply and achieves SNDR of 57 dB (i.e. ENOB 9.2 bits) near Nyquist rate input. The design of 10-bit, 14 time-interleaved (single channel) SAR ADC has been implemented. The achieved SNDR of 59.3 dB (ENOB = 9.6 bits) and SFDR of 66 dB with power consumption of 11.6 mW. This results in a FoM value of 60 fJ/step.

8.1. Future Work

The future work following this are –

- Implementation of proposed 4-channel, 1 GS/s time-interleaved SAR ADC.
- Implementation of calibration techniques to overcome the non-ideal effects in time-interleaving ADC (offset, gain and timing mismatches among the channels).

ACKNOWLEDGEMENTS

I would like to express my appreciation and special thanks to my guide Prof. Dinesh K. Sharma, Department of Electrical Engineering, IITB, Mumbai for all of his guidance, support during my work and also to Prof. Ashudeb Dutta, at DARMIC Lab, IIT Hyderabad for giving valuable suggestion for this work.

REFERENCES

[1] Kenneth C. Dyer, John P. Keane, and Stephen H. Lewis Calibration and Dynamic Matching in Data Converters s Part 2: Time-interleaved analog-to-digital converters and background-calibration challenges, *IEEE Solid-state circuits Magazine*, Summer 2018.

[2] Wei Shu, Member, IEEE, and Joseph S. ChangA 1-GS/s 11-Bit SAR-Assisted Pipeline ADC With 59-dB SNDR in 65-nm CMOS, *IEEE transactions on circuits and systems—II: Express Briefs,* vol. 65, no. 09, Sept.2018.

[3] Jie Fang, Shankar Thirunakkarasu, *Member, IEEE,* Xuefeng Yu, Fabian Silva-Rivas, Chaoming Zhang, Frank Singor, and Jacob Abraham, *Fellow, "*A 5-GS/s 10-b 76-mW Time-Interleaved SAR ADC in 28 nm CMOS *IEEE J. Solid-State Circuits* –I: Regular Papers, vol.64, no.7, July2017

[4] Chithira Ravi, Vineeth Sarma, and Bibhudatta Sahoo, "At Speed Digital Gain Error Calibration of Pipelined ADCs" IEEE transactions on very large scale integration (VLSI) systems, vol. 25, no. 11, november 2017.

[5] Time-interleaved Analog-to-Digital Converters Authors: Louwsma, Simon, van Tuijl, E.D., Nauta, Bram

[6] A 480 mW 2.6 GS/s 10b Time-Interleaved ADC With 48.5 dB SNDR up to Nyquist in 65 nm CMOS Kostas Doris, Member, IEEE, Erwin Janssen, Member, IEEE, Claudio Nani, Athon Zanikopoulos, Member, IEEE,and Gerard van der Weide ,IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 46, NO. 12, DECEMBER 2011

[7] A Time-Interleaved Flash-SAR Architecture for High Speed A/D Conversion Ba Ro Saim Sung, Sang-Hyun Cho, Chang-Kyo Lee, Jong-In Kim, and Seung-Tak Ryu School of engineering, Information and Communications University

[8] Design of High-Speed Analog-to-Digital Converters using Low-Accuracy Components Timmy Sundstr¨om ,ISBN 978-91-7393-203-5 ,ISSN 0345-7524

[9] Kent, H. L. (2002). Analog to Digital converter testing.

[10] Gustavsson, M. (1998). CMOS A/D converters for telecommunications. LINKOPING STUDIES IN SCIENCE AND TECHNOLOGYDISSERTATIONS-.

[11] Qazi, S. (2010). Study of Time-Interleaved SAR ADC and Implementation of Comparator for High Definition Video ADC in 65nm CMOS Process.

[12] Mehta, N. (2013). Sampling Time Error Calibration for Time-Interleaved ADCs. MASTER OF SCIENCE THESIS, Delft University of Technology.

[13] Liu, C. C., Chang, S. J., Huang, G. Y., and Lin, Y. Z. (2010). A 10-bit 50-MS/s SAR ADC with a monotonic capacitor switching procedure. IEEE Journal of Solid-State Circuits, 45(4), 731-740.

[14] Ginsburg, B. P., and Chandrakasan, A. P. (2007). 500-MS/s 5-bit ADC in 65-nm CMOS with split capacitor array DAC. IEEE Journal of Solid-State Circuits, 42(4), 739-747.

[15] Yue, X. (2013). Determining the reliable minimum unit capacitance for the DAC capacitor array of SAR ADCs. Microelectronics Journal, 44(6), 473-478.

[16] Tsividis, Y. (2002). Mixed analog–digital VLSI devices and technology. World Scientific.

[17] Anderson, T. O. (1972). Optimum control logic for successive approximation analog-to-digital converters. Deep Space Network Progress Report, 13, 168-176.

[18] Rossi, A.,and Fucili, G. (1996). Nonredundant successive approximation register for A/D converters. Electronics letters, 32(12), 1055-1057.

[19] Craninckx, J., & Van der Plas, G. (2007, February). A 65fJ/conversionstep 0-to-50MS/s 0-to-0.7 mW 9b charge-sharing SAR ADC in 90nm digital CMOS. In Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International (pp. 246-600). IEEE.

[20] M. Yoshioka, K. Ishikawa, T. Takayama, and S. Tsukamoto, "A 10b 50 MS/s 820 mW SAR ADC with on-chip digital calibration," in Proc.IEEE ISSCC Dig. Tech. Papers, Feb. 2010, pp. 384–385.

[21] Huang, G. Y., Chang, S. J., Liu, C. C., & Lin, Y. Z. (2013). 10-bit 30-MS/s SAR ADC using a switchback switching method. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 21(3), 584-588.

[22] Liu, C. C., Chang, S. J., Huang, G. Y., & Lin, Y. Z. (2010). A 10-bit 50-MS/s SAR ADC with a monotonic capacitor switching procedure. IEEE Journal of Solid-State Circuits, 45(4), 731-740.

[23] Gupta, S., Choi, M., Inerfield, M., & Wang, J. A 1 GS/s 11 b time-interleaved ADC in 0.13muhboxm CMOS. In IEEE ISSCC Dig. Tech. Papers (Vol. 49, pp. 576-577).

[24] Black, W. C., & Hodges, D. A. (1980). Time interleaved converter arrays. IEEE Journal of Solid-state circuits, 15(6), 1022-1029.

[25] Jamal, S. M., Fu, D., Chang, N. J., Hurst, P. J., & Lewis, S. H. (2002). A 10-b 120-Msample/s time-interleaved analog-to-digital converter with digital background calibration. IEEE Journal of Solid-State Circuits, 37(12), 1618-1627.

[26] Ginsburg, B. P., & Chandrakasan, A. P. (2005, May). An energy-efficient charge recycling approach for a SAR converter with capacitive DAC. In Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on (pp. 184-187). IEEE.

[27] Ginsburg, B. P., & Chandrakasan, A. P. (2007). Dual time-interleaved successive approximation register ADCs for an ultra-wideband receiver. IEEE Journal of Solid-State Circuits, 42(2), 247-257.

[28] Pelgrom, M. J., Tuinhout, H. P., & Vertregt, M. (1998, December). Transistor matching in analog CMOS applications. In Electron Devices Meeting, 1998. IEDM'98. Technical Digest., International (pp. 915-918). IEEE.

[29] Mulder, J., Ward, C. M., Lin, C. H., Kruse, D., & Westra, J. R. A 21 mW 8 b 125 MS/s ADC occupying 0.09hboxmm2 in 0.13muhboxm CMOS. In IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (pp. 260-261).

[30] Hegong Wei, *Member, IEEE*, Peng Zhang, Bibhu Datta Sahoo, and Behzad Razavi, *Fellow, IEEE* "An 8-Bit 4GS/s 120mW CMOS ADC", *IEEE J. Solid-State Circuits*, vol.49, no.8, Aug.2014.

[31] M. Brandolini et al., "26.6 A 5GS/S 150mW 10b SHA-less pipelined/SAR hybrid ADC in 28nm CMOS," in ISSCC Dig. Tech. Papers, San Francisco, CA, USA, 2015, pp. 1–3.

AUTHORS

Shravan Kumar Donthula Currently pursuing PhD in Micro-Electronics and VLSI Design, IIT Hyderabad with focus on developing Analog and Mixed Signal IC design for low power applications. Prior to this he completed MTech in Micro-electronics and VLSI Design from IITB, Mumbai.

Supravat Debnath Currently pursuing MTech in Integrated Sensor Systems as a Research Assistant under Dr. Ashudeb Dutta, at DARMIC Lab, IIT Hyderabad with focus on developing Analog/RF front-ends for low power, long range communication systems. Previously he completed BTech in Electronics and Comm. engineering from Kalyani Govt. Engg. College.

MULTICHANNEL ADC IP CORE ON XILINX SOC FPGA

A.Suresh, S.Shyama, Sangeeta Srivastava and Nihar Ranjan

Embedded Systems, Product Development and Innovation
Center (PDIC), BEL, India

ABSTRACT

Sensing of analogue signals such as voltage, temperature, pressure, current etc. is required to acquire the real time analog signals in the form digital streams. Most of the static analog signals are converted into voltage using sensors, transducers etc. and then measured using ADCs. The digitized samples from ADC are collected either through serial or parallel interface and processed by the programmable chips such as processors, controllers, FPGAs, SOCs etc. In some cases, Multichannel supported ADCs are used to save the layout area when the functionalities are to be realized in a small form factor. In such scenarios, parallel interface for each channel is not a preferred interface considering the more number of interfaces / traces between the components. Hence, Custom, Sink synchronized, Configurable multichannel ADC soft IP core has been developed using VHDL coding to interwork with multichannel supported, time division multiplexed ADCs with serial interface. The developed IP core can be used either as it is with the SPI interface as specified in this paper or with necessary modifications / configurations. The configurations can be the number of channels, sample size, sampling frequency, data transfer clock, type of synchronization – source / sink, control signals and the sequence of the operations performed to configure ADC. The efficiency of implementation is validated using the measurements of throughput, and accuracy for the required range of input with acceptable tolerances. ZYNQ FPGA and LTC2358 ADC are used to evaluate the developed IP core. Integrated Logic Analyser (ILA) which is an integrated verification tool of Vivado is used for Verification.

KEYWORDS

Configurable ADC soft IP Core, Sensor, Multichannel ADC, Sink Synchronization, FPGA, VHDL [3].

1. INTRODUCTION

The latest digital technology revolution and the rapid growth of semiconductor technology in the fields of FPGA, SOC [2][5][6], RFSOC [5], etc. and programming flexibility lead to choosing either SOC or RFSOC based applications. ASIC based solution is expensive for very high throughput [3]. In general ASIC solution is not cost effective unless there is no mass manufacturing. Programmable chip based platform demands the engineer to develop VHDL, Verilog, System C based IP cores when there are no readily available IP cores from the vendor for the particular interface or for the application. Since the FPGA has further grown into SOC and RFSOC with the integration of inbuilt multicore processor, controller such as ARM cores, Application Processing Unit (APU), real time Processing Unit (RPU) etc., these platform demand further to have the software development skill based on C, C++, Embedded C either with Operating system such as LINUX, VxWORKS, or customized OS such as PetaLINUX or without any OS – in bare metal mode. Programmable chip has to incorporate with indigenously

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 341-353, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112326

developed IP core or vendor specific, technology dependent, readily available free IP core or purchasable third party IP cores.

The developed architecture and the platform uses such SOC FPGA with inbuilt dual ARM core, logic resources such as LEs, memory bits, etc. The developed reconfigurable soft IP core and the approach can be used to develop such custom IP cores for any FPGA and for any type of ADC except the ADC which is hard IP core of RFSOC. The developed IP core also is configurable to change the clock mode, sampling frequency, no of channels etc. The soft ADC IP core and inbuilt ADC hard IP core which is available in RFSOC FPGAs have the limitation in supporting input voltage range for higher voltages, generally beyond 1V. In such scenarios, ADC IC has to be used.

As per the requirement, the input voltage range from -10.24 volts to +10.24 volts is to be measured. In addition to this requirement, considering the number of channels, sample size, accuracy, number of pins available at FPGA, sampling frequency, acceptable propagation and processing delay, complexity of the configuration, obsolescence of the ADC IC etc; ADC LTC2358 [1] was chosen. These aforementioned parameters are the motivation to choose a specific IC and serial interface with minimal interconnections for high speed at Mbps, advantage of sink sync scheme over source sync scheme are the motivation to the associated algorithm. But the reusability of the configurable multichannel ADC soft IP core across the different technology based FPGAs has been considered and behavioural VHDL has been written for the core ADC IP core. More details are provided in Section II.

In practical scenarios, the input voltage level range may vary and expected accuracy also may vary. Hence, it would be a desired feature / requirement to configure the sensing input voltage range and other parameters either through GUI (Graphical User Interface) / CUI (Command User Interface) or to set from the possible options / settings as a part of the factory settings of the firmware. Provisions are provided to select the parameters through CUI after the POST is successful. Even in this implementation, the IP core is set with default configuration and further can be changed through CUI as and when the requirement changes. LTC2358 has an option to set the voltage ranges from +/-10.24V, 0V to 10.24V, +/-5.12V, or 0V to 5.12V. Correspondingly, the achievable accuracy in terms of error is 0.12 mV theoretically and practically 11.37 mV.

Sink synchronization is preferred in few scenarios where the whole processes have to be operated with reference to a single master clock for better synchronization, to reduce the number of interconnections, and to avoid the multiple clock domain, to avoid glitches, to avoid or to reduce the phase error and to mitigate the effect of the accumulated jitter. Otherwise, additional techniques such as buffering or one clock delay in the case of source synchronous mode are required. Desired Serial interface and sink synchronization are exploited as additional advantages with the configurable and reconfigurable custom multichannel ADC soft IP core. At the same time, JEDEC JESD204 is recommended for high speed ADC interfaces at Gbps [9]. The developed IP core's main functionality is verified by sensing the static analog signals, and the processing of dynamic ADC samples is not the scope of the work. That's why the performance measurement such as SINAD measurement, SFDR, etc are not required and not done. Integrated Logic Analyser (ILA) [7] which is an integrated verification tool of Vivado is used for Verification. No third party tool is required, whereas [2] uses Synopsis Discovery AMS platform. The main contributions are the realization of Sink synchronized, Configurable multichannel ADC soft IP core in PL section, the driver, API, and the embedded software developed for the inbuilt ARM core and the developed CUI for the configuration of ADC IP core and for the verification of results. CUI is operated through debug port based on UART interface. The implemented ADC IP core has a configurable FIFO size which can be configured only in the firmware. This FIFO size can be varied based on the application requirement. Depth of FIFO or storage space is configurable from 8 samples to 1K samples. The same space is required in ARM processor also. The achieved processing speed of the algorithm supports 200 Ksps / Channel with the acquisition time of 570ns.

In the present paper, concept and flow are explained through Section II to Section VI. Section II covers the relevant portion of the hardware platform architecture, data flow between the components, and brief summary about the relevant components used in the hardware. The software development approach, associated tools, and software mapping into the hardware are explained in Section III. Section IV depicts the data transfer protocol, flow chart, pseudo random code / portion of the code, and brief explanation. Section V demonstrates the simulation and test results of the implementation to verify the functional requirements and the measurement of the main parameters such as throughput, and accuracy. Section VI concludes with the novel claims, developments and chances for further proliferation.

2. HARDWARE PLATFORM ARCHITECTURE, AND DATAFLOW

IP core for Max104 ADC is developed on Virtex2Pro in [3]. But, here, the hardware consists of ZYNQ [4] XC7Z045FFG900-0I as the core. The hardware platform architecture (HPA) with the functional off chip peripherals is shown in Figure 1. The analogue signals to the hardware is coming through a 3U VPX board edge connector which is filtered and given to the ADC LTC2358. The digitized output from ADC is then received into the PL section of FPGA and received packetized / interleaved / multiplexed 6 channels data is unwrapped into 6 channels at PL, and then serial data is sent to the PS section using AXI interface where it is further processed and converted into digital voltage and displayed on the terminal. If the data to be received into the ARM is not required to be sent to any other chip, Board or system, channel identifier is not to be sent from PL to PS and they can be discarded at PL. If it is to be sent to other chip, board, or system, then channel identifiers also are to be received into ARM processor. GPIO address identifier is used for all the peripherals realised in PL section and connected to ARM core through AXI – On chip peripheral bus to transfer the data through AXI bus. But, those address identifiers are not the better choice to identify the channel, since the size of the address identifier is greater than or equal to 32 bits, because, generally an address range is reserved for each AXI peripheral. The complete dataflow is shown in figure 2.

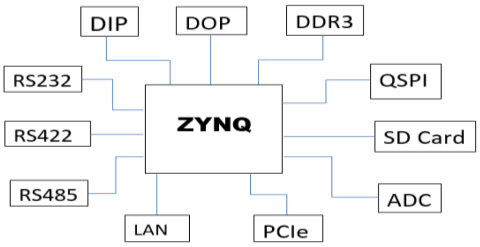


Figure 1. Hardware platform architecture

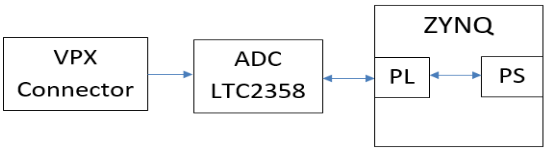


Figure 2. Dataflow

LTC2358 from Analogue Devices is the ADC chosen. It is a 16 bit SAR ADC for which throughput of 200ksps can be achieved per channel. It has eight buffered differential input channels, of which six have been used but can be upgraded for all 8 channels. The board space is considerably decreased by accommodating 8 channels into one ADC. The input ranges supported are +/-10.24V, 0V to 10.24V, +/-5.12V, 0V to 5.12V, +/-6.25V, and 0V to 6.25V. It operates from a 5V low voltage supply. LTC2358 supports both LVDS and CMOS modes of operation. CMOS mode is chosen since it gives the output of 6 channels in 6 different pins whereas in LVDS mode all the 6 channel data is multiplexed and given in the same set of differential pins and hence the speed is considerably reduced. Also the power dissipation of CMOS mode is 259mW which is less compared to 287mW in LVDS mode. Hence CMOS mode of operation allows the user to optimize bus width, throughput and power.

ZYNQ 7000 SOC [2],[4],[5],[6] is chosen to achieve the complex algorithms in the design. It integrates a feature-rich dual-core ARM® Cortex™-A9 based processing system (PS) and 28 nm Xilinx programmable logic (PL) in a single device.

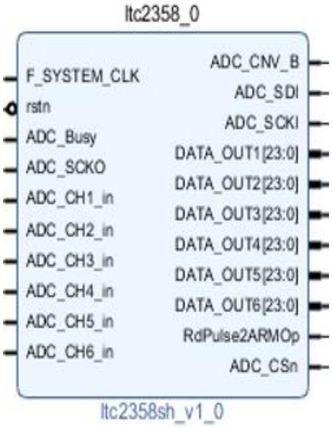
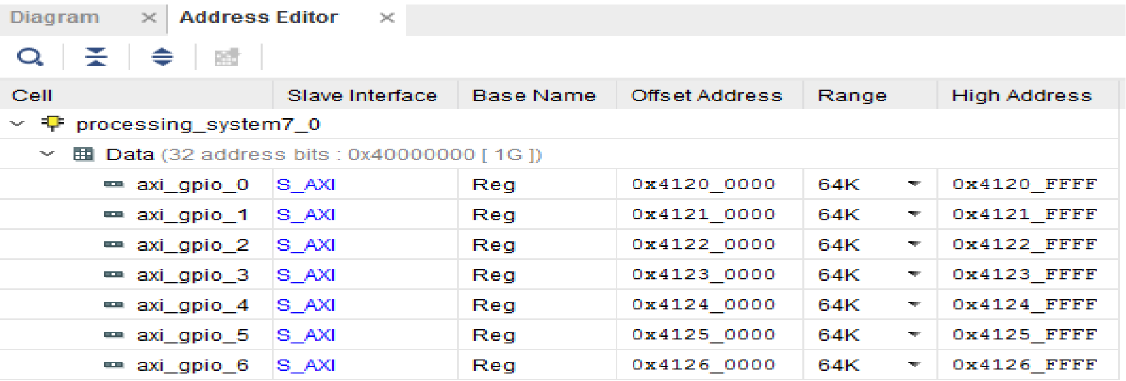


Figure 3. Custom ADC IP core

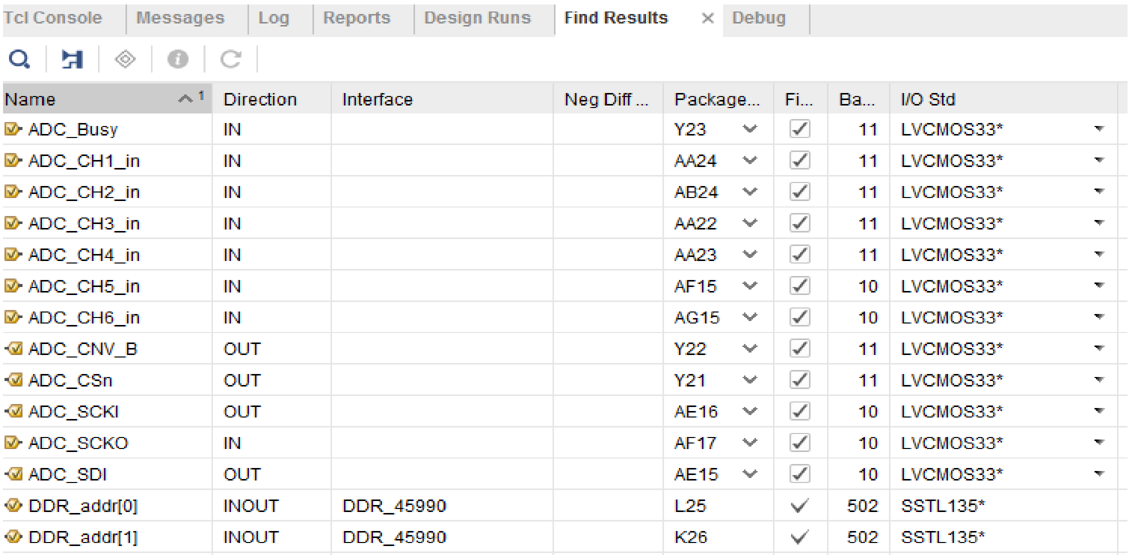
3. Software Development Approach, Associated Tools, And Software Mapping into the Hardware

The software design to receive the digitized output from ADC and to packetize it to transfer to PS section was done in VHDL logic using VIVADO 2019.2 tool from XILINX. After receiving the data into the PS section it is processed and verified in SDK [6] 2019.2 provided by XILINX. Petalinux OS version 2019.2 is used for the development of embedded software and the CUI. In the VIVADO [6] tool, VHDL is written in the VHDL editor code to receive input from the ADCs. Later this VHDL logic was converted to a custom IP core as shown in figure 3 and invoked to the top level block diagram. The digitized packetized data output from the ADC custom IP core is given to the AXI GPIO IP core [2]. For each channel of ADC, one AXI GPIO IP core is used through which an address identifier is mapped to each AXI peripheral.



Cell	Slave Interface	Base Name	Offset Address	Range	High Address
processing_system7_0					
Data (32 address bits : 0x40000000 [1G])					
axi_gpio_0	S_AXI	Reg	0x4120_0000	64K	0x4120_FFFF
axi_gpio_1	S_AXI	Reg	0x4121_0000	64K	0x4121_FFFF
axi_gpio_2	S_AXI	Reg	0x4122_0000	64K	0x4122_FFFF
axi_gpio_3	S_AXI	Reg	0x4123_0000	64K	0x4123_FFFF
axi_gpio_4	S_AXI	Reg	0x4124_0000	64K	0x4124_FFFF
axi_gpio_5	S_AXI	Reg	0x4125_0000	64K	0x4125_FFFF
axi_gpio_6	S_AXI	Reg	0x4126_0000	64K	0x4126_FFFF

Figure 4. Address mapping from PL to PS



Name	Direction	Interface	Neg Diff ...	Package...	Fi...	Ba...	I/O Std
ADC_Busy	IN			Y23	☑	11	LVCMOS33*
ADC_CH1_in	IN			AA24	☑	11	LVCMOS33*
ADC_CH2_in	IN			AB24	☑	11	LVCMOS33*
ADC_CH3_in	IN			AA22	☑	11	LVCMOS33*
ADC_CH4_in	IN			AA23	☑	11	LVCMOS33*
ADC_CH5_in	IN			AF15	☑	10	LVCMOS33*
ADC_CH6_in	IN			AG15	☑	10	LVCMOS33*
ADC_CNV_B	OUT			Y22	☑	11	LVCMOS33*
ADC_CSn	OUT			Y21	☑	11	LVCMOS33*
ADC_SCKI	OUT			AE16	☑	10	LVCMOS33*
ADC_SCKO	IN			AF17	☑	10	LVCMOS33*
ADC_SDI	OUT			AE15	☑	10	LVCMOS33*
DDR_addr[0]	INOUT	DDR_45990		L25	✓	502	SSTL135*
DDR_addr[1]	INOUT	DDR_45990		K26	✓	502	SSTL135*

Figure 5. I/0 ports pin assignment

Each AXI GPIO IP core is mapped to the PS using a unique base address as shown in figure 4. Then the ADC data is passed on to the PS through AXI interconnect block from AXI GPIO IP core. After completing the block diagram in VIVADO, HDL wrapper is generated for the design. Then the design is synthesized and pin assignments are done in I/O port assignments as shown in figure 5. Then the design is implemented and the bit stream is generated. The bit stream is then exported to the SDK tool where the BSP to test the interface is developed. Then the bit stream is flashed into the PL through JTAG emulator and the BSP is run on PS using the debug port and viewed in SDK terminal.

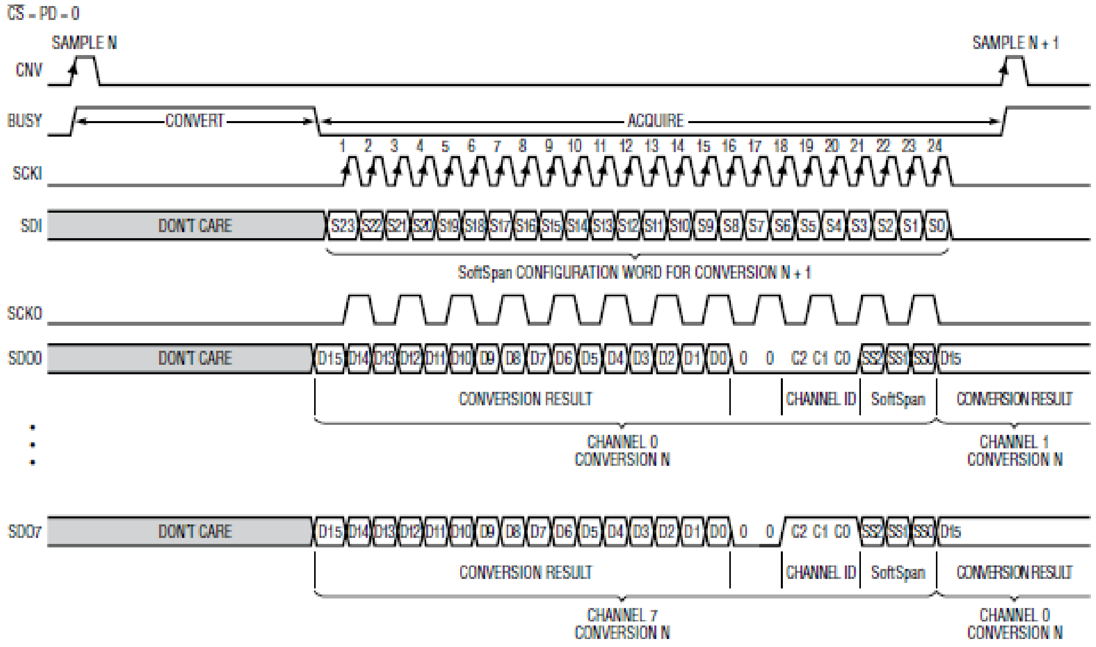


Figure 6: Protocol / Timing diagram [1]

4. DATA TRANSFER PROTOCOL, FLOW CHART, PSEUDO RANDOM CODE AND BRIEF EXPLANATION

The ADC custom IP core is designed based on standard SPI CMOS interface in CMOS I/O mode. SPI interface generally consists of slave select (SS), Master Output / Slave Input (MO/SI), Master Input / Slave Output (MI/SO), and Serial Clock (SCK). Respectively, the following signals / Pins are utilised as follows : (i) ADC chip select signal -"ADC_CSn" as $\overline{CS}$ / (SS), (ii) SoC's output /ADC's input serial data to configure ADC -"ADC_SDI" as SDI (MO/SI), (iii) SoC's input /ADC's output serial data for 8 channels – "DATA_OUT1" / "ADC_CH1_in" to "DATA_OUT6" / "ADC_CH6_in" as SDO0 to SDO7 (MI/SO), and (iv) Serial clock from Master (SoC) – "ADC_SCKI" as SCKI (SCK).

First the CHIP SELECT pin - "ADC_CSn" is made low (Active Low Reset) from the FPGA to enable ADC. This active low reset signal is generated only once and then maintained or kept low throughout the ADC operation. Protocol does not demand any resynchronization between ADC chip and FPGA. The developed IP core also is working consistently.

Then ADC convertor pulse "ADC_CNV_B" is made high for 20ns. The same pulse "ADC_CNV_B" is repeated once for every 24 bits, since all the 8 channels send the digitized data and the channel related data sequentially through its own data pins (SDO0 to SDO7) to FPGA. During the high of "ADC_CNV_B" pulse, ADC samples the analog input for the enabled channels. The channels are configured with the range of input voltage as mentioned in Table 1.

Next during high value of "BUSY" signal, digitized 16 bit data is packetized with 3 bits of channel id, 3 bits of soft span code as given in table 1, and with 2 bits of zeros as per the protocol shown in figure 6. Binary soft span code [2:0] generated by the SDO lines are configured by the FPGA for the ADC configuration and it can be updated on the fly if instructed through CUI for various voltage levels. This information comes along with the sampled data on the ADC_CHx_IN (SDO) lines.

For example, if the digitized 16 bit sample data for a particular analog input is “0000 0101 0111 1100”, then the packetized 24 bit word as per protocol is “0000010101111100 00 000 111” for the enabled channel “000” and for the voltage to be measured “+/- 10.24V”.

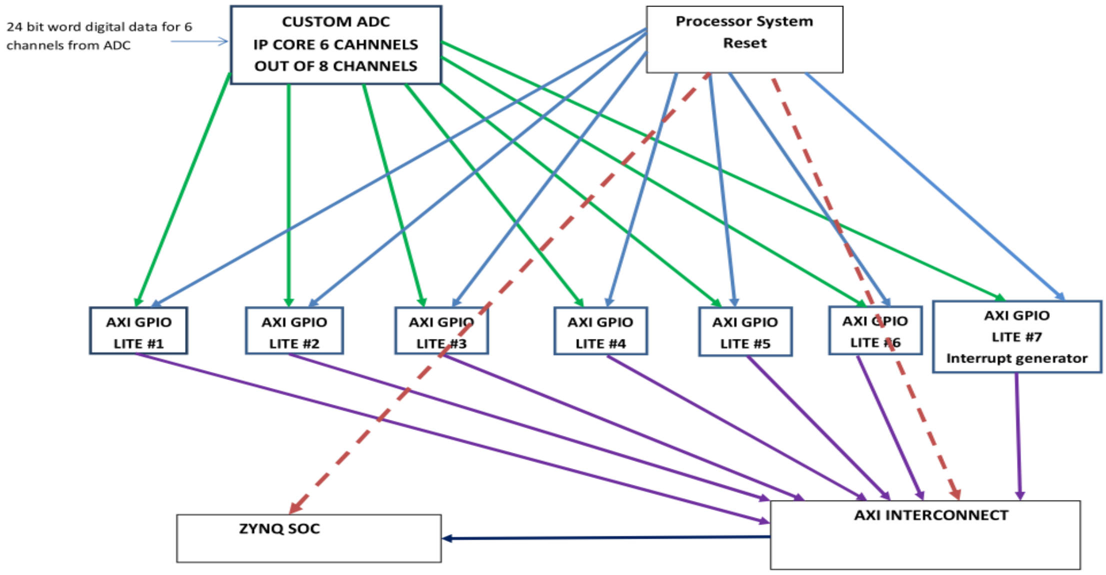


Figure 7. PL section Design with ADC IP core

New data transaction starts at the end of each conversion on the falling edge of BUSY. As per the protocol as shown in Figure 6, 24 bit data frame is sent from ADC to ADC IP core, named as “ltc2358sh_V1_0” whose symbol or IO block diagram is shown in Figure 3.

The mapping between the serial Soft Span configuration word, the internal Soft Span configuration register, and each channel’s 3-bit Soft Span code are illustrated in Table 1 [1]. The input voltage range for each channel and enabling of channels can be enabled from the ARM core in the PS section of Zynq FPGA, based on the requirement by setting the corresponding soft span code. The flow chart is shown in figure 7. PL section Design with ADC IP core is shown in Figure 8. A portion of the code or Pseudo random code is provided. Next the inner block diagram of ADC IP core and the realization is explained below.

Table 1. Soft Span Configuration

Sl no	**Binary Soft Span CODE SS[2:0]**	**Analog Input Range**
1	111	+/- 10.24V
2	110	+/- 10V
3	101	0 to 10.24V
4	100	0 to 10V
5	011	+/- 5.12V
6	010	+/- 5V
7	001	0 to 5.12V
8	000	CHANNEL DISABLE

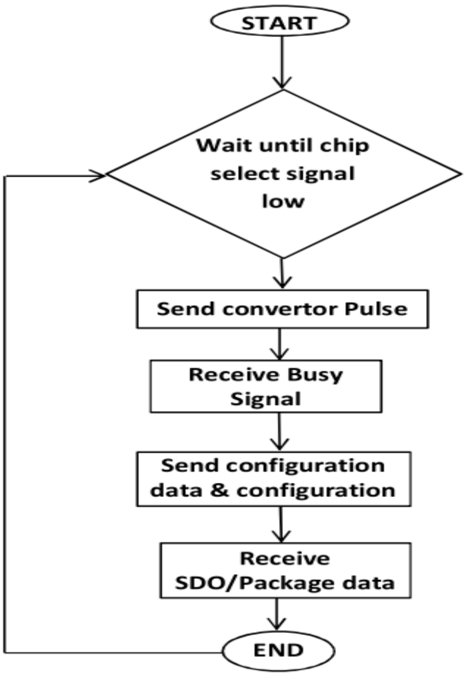
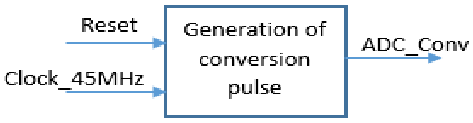

START
Wait until chip select signal low
Send convertor Pulse
Receive Busy Signal
Send configuration data & configuration
Receive SDO/Package data
END

Figure 8. Flow chart

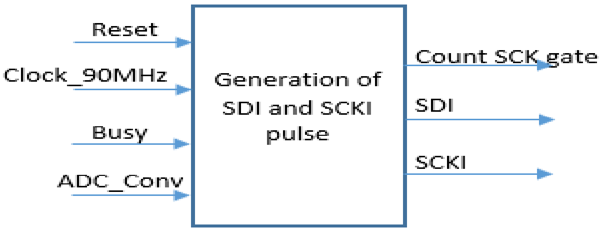
4.1. Generation of Conversion Pulse

According to the figure shown if the reset signal is zero, the output as well as internal counter is set / initialised to zero. And if the reset is high, the process for the generation of ADC conversion pulse begins. An ADC convertor pulse of 44.44ns is generated by running an internal counter. The counter is made high for 2 consecutive clock pulses of 45 MHz.



4.2. Generation of SDI and SCKI pulses

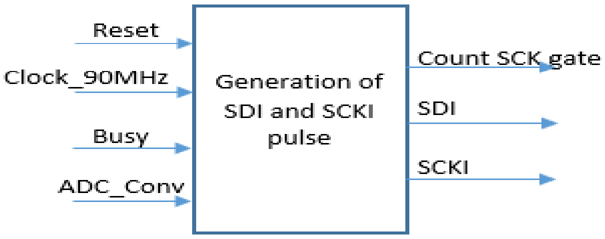
According to the figure shown if the reset signal is zero, the output internal signals, internal counter and shift register are set to zero. And if the reset is high, the process for the generation of Serial data input (SDI) and serial clock input (SCKI) begin with respect to 90 MHz clock.



If busy pulse equals to “0” and ADC convertor pulse equals to “0”, it is checked if sense busy is “1”, then the counter is incremented for 48 pulses. SCKI of 45MHz is generated by toggling the 90 MHz SCKI pulse and SDI is transferred with the configuration data “03FFFF” from a 24 bit shift register which was already initialized to “03FFFF”. In all the other cases of busy pulse and ADC convertor pulse, internal signals and counters are reset to “0”. After this all the internal counter and shift register is reset to zero.

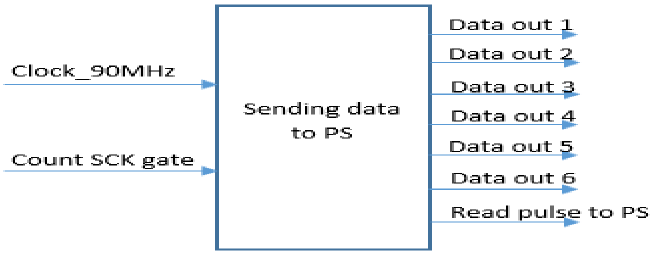
4.3. ADC Channel Data Reception

For the data reception process, during each falling edge of SCKI, 24 bit shift registers (ADC_CH1 to ADC_CH6) are filled from the data coming from 6 ADC channels respectively.



4.4. Sending ADC data to PS (ARM core)

As shown in the figure below for the rising edge of 90MHz clock, when the *Count_SCK_Gate = “49”*, a frame of complete 24 bit data is transferred to ZYNQ. An interrupt pin called read pulse is made high which indicates the PS that now the data is ready to be read from PL. After this process, the data in 6 channels are kept constant until the next data frame is ready and the read pulse is made “0”.



A portion of the ADC IP core to generate convertor pulse in PL section is,

```
if( Count_conv < 447 )    then Count_conv <=Count_conv + 1;
     elsif (Count_conv >= 447) then Count_conv <= 0;
```

The pseudo random code to generate SDI and SCKI in PL section is,

```
if (sense_busy = '1') then
                    if(Count_SCK_Gate < 447) then
                         if(Count_SCK_Gate < 48) then
          SCK_Gate <= '1';              SCKI <= not SCKI;
          if(SCKI = '1') then SDI <= SDIreg(23);  SDIreg <= SDIreg(22 downto 0) & '0';
```

```
    elsif(SCKI = '0') then null;
    end if;
   elsif(Count_SCK_Gate >= 48) then
         SCKI <= '1';      SDI <= '0';      SCK_Gate <= '0';
  end if;
       Count_SCK_Gate <= Count_SCK_Gate+1;   sense_busy <= '1';
  elsif (Count_SCK_Gate >= 447) then
          Count_SCK_Gate <= 0;   sense_busy<='0';
             SCKI <= '1';     SDI <= '0';     SCK_Gate <= '0';
 end if;
```

The pseudo random code to capture ADC data is,

```
if(SCK_Gate = '1') then
DATA_OUT1<=ADC [1]_CH1(23 DOWNTO 0); RdPulse2ARM<='1'; datacount<= 0;
```

Pesudo Random code / Section of the code written in PS section is provided below:

```
/* Initialize the ctrl driver */
Status = XGpio_Initialize(&ctrl, GPIO_ctrl_DEVICE_ID);
if (Status != XST_SUCCESS) {
        xil_printf("Read control from PL Failed\r\n"); return XST_FAILURE;
                }
                /* Set ctrl as inputs*/
XGpio_SetDataDirection(&ctrl, control, RdPulse2ARMOp);
                xil_printf("Capturing ADC data of all 6 channels\r\n");
                while(Rdcnt < 10)
                readctrl_adc = XGpio_DiscreteRead(&ctrl, control);
                if (readctrl_adc == 0x00)      OnePadcrd  = 0x01;

        else if ((readctrl_adc == 0x01) && (OnePadcrd  == 0x01))
                OnePadcrd  = 0x00;             Rdcnt += 1;
                xil_printf("-----%dth Read ---\r\n",Rdcnt);
//ADC_CH1    // Initialize the ADC_CH1 driver
Status = XGpio_Initialize (&ADC_CH1,  GPIO_ADC_CH1_DEVICE_ID);
        if (Status != XST_SUCCESS) {
        xil_printf("ADC1 Initialization Failed\r\n");
                return XST_FAILURE;
        // Set ADC_CH1 as inputs
 XGpio_SetDataDirection(&ADC_CH1, ADC1_CHANNEL, adcin);
                // Reading ADC input
CH1_IN = XGpio_DiscreteRead(&ADC_CH1,ADC1_CHANNEL);
                xil_printf("CH1_IN = %x\n",CH1_IN);
```

5. SIMULATION RESULTS AND TEST RESULTS

The simulation results were viewed on ILA in VIVADO tool, as shown in figure 9. As a part of booting, ADC IP core and its associated functional modules are implemented in PL section to transfer the data from ADC IP core to Zynq PS section through AXI interconnect and to transfer the configuration words for all the 6 channels through AXI interconnect from PS section. After successful booting of the board in PetaLinux, the ADC program was executed in PS section as per the application imposed sequence. The output will be shown in the running console to verify the ADC value and Voltage of all 6 channels as shown in figure 9. The figure 9 shows the test results when an input voltage of zero is fed to all the 6 channels.

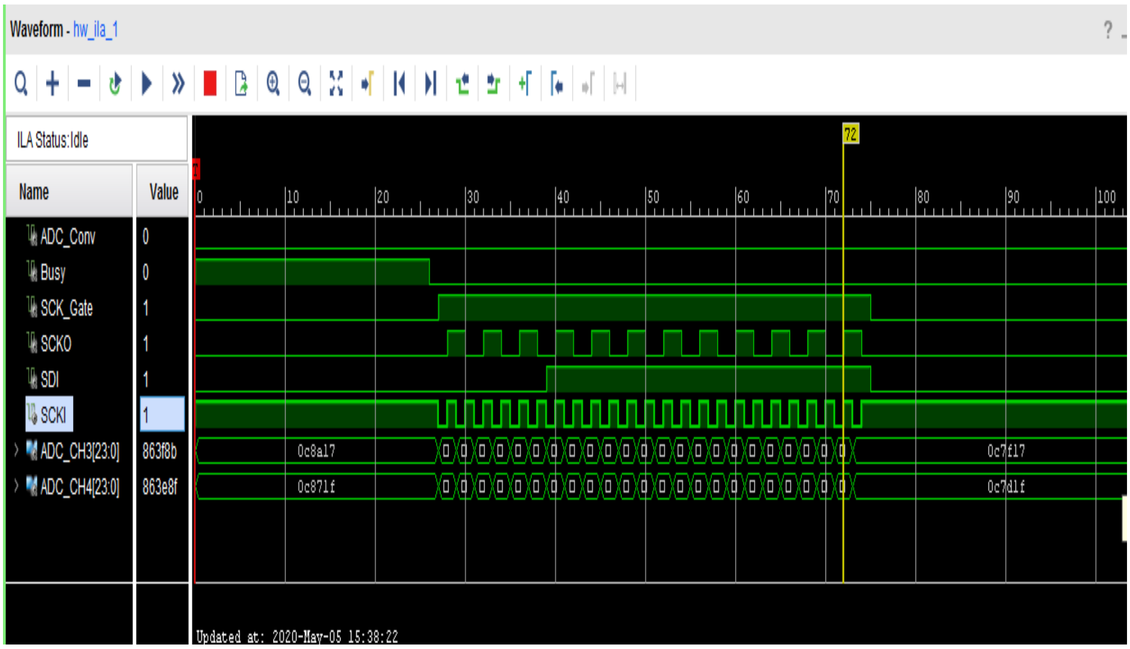
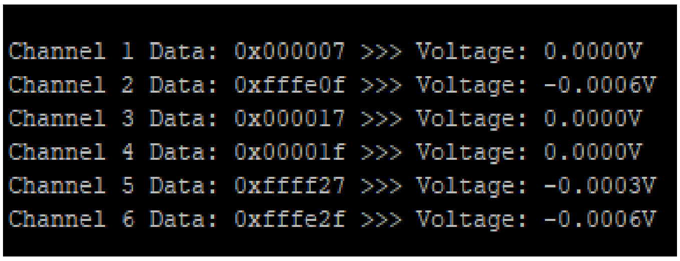


Figure 9: Simulation result



```
Channel 1 Data: 0x000007 >>> Voltage: 0.0000V
Channel 2 Data: 0xfffe0f >>> Voltage: -0.0006V
Channel 3 Data: 0x000017 >>> Voltage: 0.0000V
Channel 4 Data: 0x00001f >>> Voltage: 0.0000V
Channel 5 Data: 0xffff27 >>> Voltage: -0.0003V
Channel 6 Data: 0xfffe2f >>> Voltage: -0.0006V
```

Figure 10. Test result

6. Conclusions

Provision is made to configure the input analogue voltage range of each channel of ADC based on the application through soft span configuration, So that the developed IP core can be reused for many applications. Most of the Custom IP cores for medium logic can be developed by following the development flow explained in this paper, unless the requirement needs to any high speed data transfer. Even for such application requirement, DMA based data transfer is implemented to store the data at the end in the DDR attached with PS. Since PS section has the on chip peripheral support for DDR, DDR was connected to PS. Integrated Logic Analyser (ILA) [7] which is an integrated verification tool of Vivado is used for Verification. No Third party tool is required. Accuracy, and the consistency have been verified successfully in standalone mode testing and even in the integrated mode when this code is integrated with others in the equipment and then at the end with the system at the actual field. The ADC IP core is developed in a modular approach [8] to scale up to accommodate the balance 2 channels for the variant / different user requirement. Because of Modular design, the same design can be used to enhance to N numbers such ADC channels.

The realised ADC IP core and the approach followed will be used as reference to develop other AXI peripherals such as NVSRAM, RTC interface, UART, MRAM, etc., since the front core interface only differs based on the off chip peripheral interface for future work. Once the data is received into PL section as per the custom interface, then those data are to be reframed, and packed into FIFO or BRAM to send them to ARM processor through AXI. Memory interface with custom front core interface is recommended to allot and distribute the available processor's execution time to multiple AXI peripherals and other off chip peripherals directly connected to on chip ARM processors.

Since the SPI interface based data transfer protocol can support up to Mbps only, JESD204A, B, C are used for Gbps range data transfer. JESD204C is the latest version of JESD204 which is the only interface recommended for Gigabit ADCs and DACs. The same methodology will be exploited to develop the custom JESD204C IP core with MPSoC [10].

ACKNOWLEDGEMENTS

The authors would like to thank all the people who have come across in their life from whom the authors have learnt either directly or indirectly.

REFERENCES

[1] LTC2358 datasheet Buffered Octal, 16-Bit, 200 ksps / Channel Differential ±10.24V ADC with 30VP-P Common Mode Range. D16870-0-5/18(A) . www.analog.com ANALOG DEVICES, INC. 2016-2018.

[2] Yueli Hu, Wenyi Jing, Ying Liu, "Integrating ADC IP in SoC and Its Verification", IEEE International Symposium on High Density Packaging and Microsystem Integration, Shanghai, China, 26-28 June 2007.

[3] Cristian Sisterna, Marcelo Segura, Martin Guzzo, Gustavo Ensinck, Carlos Gil, "FPGA Implementation of Ultra-High Speed ADC interface." IEEE VII Southern Conference on Programmable Logic [SPL], Cordoba, Argentina, 13-15 April 2011.

[4] ZYNQ -7000 SoC Technical Reference Manual UG585 (v1.12.2) July 1, 2018 Xilinx

[5] ZYNQ UltraScale+ RFSOC Data Sheet: Overview DS889 (v1.10) September 14, 2020 Xilinx

[6] Xilinx VIVADO/SDK Tutorial (Laboratory Session 1, EDAN15) Flavius.Gruian@cs.lth.se March 21, 2017

[7] Integrated Logic Analyser V6.2 – LogicCORE IP Product Guide, Vivado Design Suite, PG172 October 5, 2016, XILINX

[8] Yunpeng Bai, Dominic Gaisbauer, Stefan Huber, Igor Konorov, Dmytro Levit, Dominik Steffen, Stephan Paul, "Intelligent FPGA Data Acquisition Framework", IEEE-NPSS Real Time Conference, Padua, Italy, 6-10 June 2016.

[9] Jiiadong Yuan, Min Xie, Siyuan Liu, Dengyue Zhai, "Design of JESD204B Multichannel Data Acquisition and playback system based on SoPC", 11th international congress on Imange and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2018

[10] https://www.zynq-mpsoc-book.com

AUTHORS

A. Suresh was born in Virudhunagar District in TN. He received his B.E. degree in ECE from Government College of Engineering, Tirunelveli in 1998. He worked as a Lecturer in AKCE, Krishnankovil, TN from 1998 to 2000. He received his M.E in Communication Systems from REC, Trichy in 2002. Since 2002, he is working in Bharat Electronics Limited, Bengaluru. Now, he is DGM in Embedded Systems department, PDIC, Bengaluru, India.

AI4TRUTH: AN IN-DEPTH ANALYSIS ON MISINFORMATION USING MACHINE LEARNING AND DATA SCIENCE

Kevin Qu and Yu Sun

California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

A number of social issues have been grown due to the increasing amount of "fake news". With the inevitable exposure to this misinformation, it has become a real challenge for the public to process the correct truth and knowledge with accuracy. In this paper, we have applied machine learning to investigate the correlations between the information and the way people treat it. With enough data, we are able to safely and accurately predict which groups are most vulnerable to misinformation. In addition, we realized that the structure of the survey itself could help with future studies, and the method by which the news articles are presented, and the news articles itself also contributes to the result.

KEYWORDS

Machine Learning, Cross Validation, Training and Prediction, Misinformation

1. INTRODUCTION

With the advent of the information age, the internet has given us access to previously unimaginable wealth of information [7]. With tools such as google, we can access all the collective knowledge of humanity at the press of a button [8]. Yet, with all this power and knowledge, misinformation is somehow more prevalent than ever before [9]. Social media platforms such as Facebook allow misleading headlines and sometimes outright lies to spread to millions of users before anyone can do anything about it. There is a quote - commonly attributed to Mark Twain - that states that "a lie can travel halfway around the world while the truth is still putting on its shoes" [10]. This is made all the more ironic by the fact that Mark Twain most likely never said those words. That does not, however, take away from the truth in the statement, especially in this day and age.

According to the Washington Post, 59% of people comment on fake news headlines before they read the actual article. This can be especially devastating as headlines are often specifically crafted to grab a reader's attention. They often leave out information or straight up lie for views. This means that the majority of people will not get the full story.

According to statists, there are around 4.2 billion internet users across the globe. That is over half of the almost 7.87 billion people in the world, according to world meters. This means that it has become trivially easy to post practically anything and have it be seen. This means that it has become trivially easy to post practically anything and have it be seen. While this does mean that it is easier to spread information, it is also easier to spread falsehoods and rumors.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 355-362, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112327

This study focuses on how many people actually read an article after they have seen the headline [11]. Instead of a survey, they simply divided the amount of people who actually clicked on the URL by the people that saw the post. This approach is different from ours mainly because while our study relies on direct user interaction, this one uses a more indirect method [12]. One potential shortfall of this method is that people who have clicked on the links may not necessarily have fully read the article. This study functions in an incredibly similar fashion to this one. They were presented with misinformation and asked whether or not they believed it.

This method, while incredibly similar, is not the same. Instead of directly presenting the participants with misinformation, this survey asks them to identify which one they think is misinformation. This may not sound like a large difference but it is. Their method can introduce unconscious biases that may affect the results. They may also be hesitant to directly admit they believe in misinformation. These two factors could lead to skewed and biased results that cannot be accounted for. This survey, with the randomized questions and intentionally ridiculous headlines, attempts to address this issue by making it so that the real news story cannot be distinguished with ease.

This study uses a very similar approach as this one [14]. It also uses a survey of sorts and focuses on WhatsApp. The information of the participants (Age and Occupation) are taken and their responses filed under those two categories. The questions themselves ask the participant to identify which messages contain real information and which ones don't. One of the messages will also have a link of some kind to source materials while the other won't. A "score" is then calculated using what the participant thought were true or false.

This study focuses on whether or not a user will believe the information at first glance. It does not take into account the link that was provided or the website it leads to. It also only includes two factors (age and occupation) while this study has seven. This study also focuses on multiple areas of misinformation, not just health misinformation.

This study aims to send out a survey for the general populous to take. This survey would ask them to identify whether or not a news headline is real or fake based on a screenshot of the website. There are ten such questions and they address multiple areas of interest, including healthcare. The results, along with the attributes of the participants are then fed into a python script where multiple methods of classification are tested. The purpose of the algorithm is to predict the answers of the participants using their attributes. It does not necessarily predict whether or not they will believe in misinformation but rather what factors influence their decision. Unfortunately, we were unable to send this study out so we have opted to use dummy data for the purposes of refining the algorithm [15].

The factors being looked at are: gender, education, age, main source of news, social media use, income and political standing. All of those things are collected in the survey itself, with each one split into multiple categories. Age, for example, is split into the 0-13, 14-17, 18-21, 22-27, 28-35, 35-50, 50-60, and 60+ groups while political standing is split into the far left, left leaning, moderate, right leaning and far right groups. This is to make data collection easier as, even though it somewhat limits the algorithm's range, it does not allow for answers not easily parsed by the algorithm.

The way we will be determining how accurate the results are will be by comparing the algorithms output to the original data. 30% of the original data will be set aside for the algorithm to test on, with the remaining 70% to go towards testing. Unfortunately, as we have mentioned before, we do not have actual data. We only have dummy data for the sole purpose of testing the algorithm so we unfortunately cannot compare our results to that of other studies.

The rest of the paper is organized as follows: Section 2 gives the details on the challenges that we met during the experiment and designing the sample; Section 3 focuses on the details of our solutions corresponding to the challenges that we mentioned in Section 2; Section 4 presents the relevant details about the experiment we did, following by presenting the related work in Section 5. Finally, Section 6 gives the conclusion remarks, as well as pointing out the future work of this project.

2. CHALLENGES

In order to build the tracking system, a few challenges have been identified as follows.

2.1. Designing the Algorithm

One of the challenges we faced was how we wanted to design the algorithm itself [13]. We needed a way for the algorithm to determine which attribute was a contributor to a person's decisions. There were many ways we could have approached this; an example was to feed the algorithm a simple percentage of how many survey questions were correct and have it predict the person’s attributes but this would not have been ideal. First off, there were simply too many attributes for an algorithm such as this to have reliably pulled off. Secondly this would not tell us much about which specific attribute contributed the most; it would only tell us which combination would make a person get a certain percentage. The system we ended up going with allowed us to accurately see exactly which attributes would lead to which decisions on all the questions. This allowed for much
more information to be collected from the same amount of data.

2.2. Picking Out the News Articles

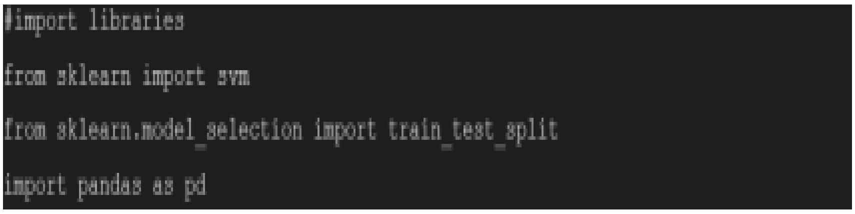
Once the general structure of the algorithms had been decided, the next challenge was to actually pick out the news articles. This was surprisingly difficult as - to provide the most unbiased and accurate set of data - the real news articles have to sound as ridiculous as possible and the fake ones have to sound as real as possible. An example of this would be in the sports section where the real news article was titled “Olympic athlete stuck in quarantine calls lack of fresh air ‘inhumane’” (from CNN) while the fake one was titled “Olympics under fire for human rights violations after forcing athletes to exert themselves” (from the Onion). Both of these sound rather far fetched and while the one from the Onions sounds a little more so, both seem to be within the realm of reality. This means that the test comes down to the participants' knowledge of satirical sites (of which the Onion is one) and other factors. It removes the potential for people to easily discern which is which based simply on the ridiculousness of the headline.

2.3. Getting Necessary Data

Another major challenge in conducting this study has been actually getting the necessary data from the survey we constructed. While we did manage to get a handful of responses, it was not nearly enough to both train the algorithm and test it. Beyond that, any survey responses we did manage to get would be heavily skewed and biased seeing as our own friend groups would most likely share the same or at least similar views with us. This means, short of sending out the survey en masse, that any data collected would be more or less useless. As a result, we decided to use dummy data to train the algorithm and make sure it works just so that the experiment can go on.

3. SOLUTION

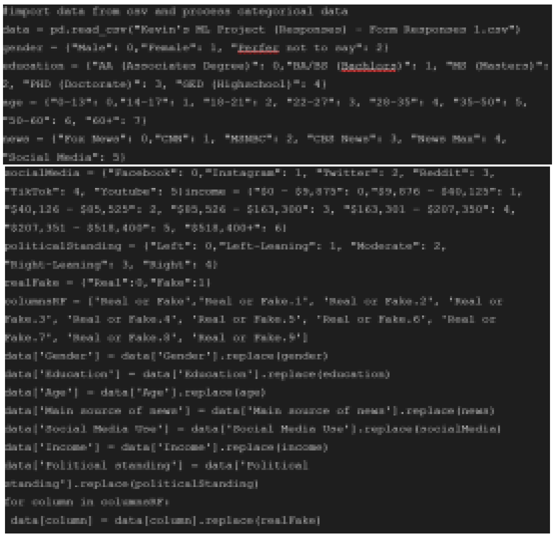
The purpose of the code is to predict, given ample training data, which attributes contribute the most to believing in misinformation. The machine first takes in roughly 70% of the survey responses to train the model. It then uses the remaining 30% to test the accuracy. If the model is able to accurately determine the attributes given the person's responses to the survey, then we know that this would have been a determining factor in whether or not they may believe in misinformation. This test can be repeated for each attribute to determine which one is most likely a determining factor. The first step of this process is to collect data. After the raw data is collected using the survey, it is imported using the pandas library and all the words are swapped with numbers for the machine learning library to understand. The Scikit Learn library is then used for the actual machine learning aspect of the code. Finally, the scores for each of the attributes is printed out at the end to determine whether or not a person with that attribute is likely to believe in misinformation. The first segment of code is the importing of all the libraries.



```
#import libraries

from sklearn import svm

from sklearn.model_selection import train_test_split

import pandas as pd
```

Figure 1. Code of importing libraries

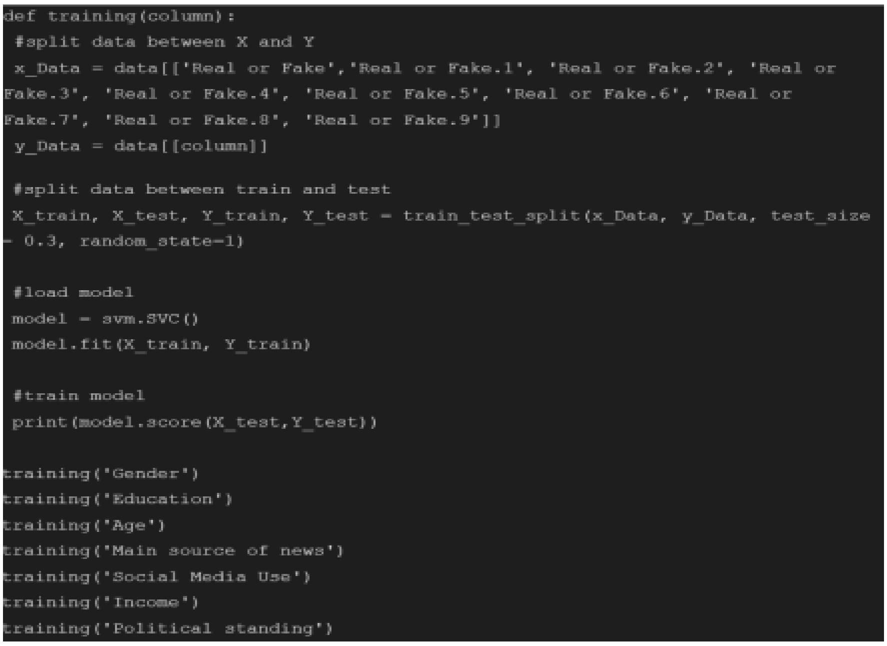
Then comes the data preparation. This includes importing the data with the pandas library and swapping all the words with numbers so that the machine learning library can understand it. An exanple of this process would be that “Male” is replaced with 0 and “Female” is replaced by 1 in the Gender column.



```
#import data from csv and process categorical data
data = pd.read_csv("Kevin's ML Project (Responses) - Form Responses 1.csv")
gender = {"Male": 0,"Female": 1, "Perfer not to say": 2}
education = {"AA (Associates Degree)": 0,"BA/BS (Bachlors)": 1, "MS (Masters)":
2, "PHD (Doctorate)": 3, "GED (Highschool)": 4}
age = {"0-13": 0,"14-17": 1, "18-21": 2, "22-27": 3, "28-35": 4, "35-50": 5,
"50-60": 6, "60+": 7}
news = {"Fox News": 0,"CNN": 1, "MSNBC": 2, "CBS News": 3, "News Max": 4,
"Social Media": 5}
socialMedia = {"Facebook": 0,"Instagram": 1, "Twitter": 2, "Reddit": 3,
"TikTok": 4, "Youtube": 5}income = {"$0 - $9,875": 0,"$9,876 - $40,125": 1,
"$40,126 - $85,525": 2, "$85,526 - $163,300": 3, "$163,301 - $207,350": 4,
"$207,351 - $518,400": 5, "$518,400+": 6}
politicalStanding = {"Left": 0,"Left-Leaning": 1, "Moderate": 2,
"Right-Leaning": 3, "Right": 4}
realFake = {"Real":0,"Fake":1}
columnsRF = ['Real or Fake','Real or Fake.1', 'Real or Fake.2', 'Real or
Fake.3', 'Real or Fake.4', 'Real or Fake.5', 'Real or Fake.6', 'Real or
Fake.7', 'Real or Fake.8', 'Real or Fake.9']
data['Gender'] = data['Gender'].replace(gender)
data['Education'] = data['Education'].replace(education)
data['Age'] = data['Age'].replace(age)
data['Main source of news'] = data['Main source of news'].replace(news)
data['Social Media Use'] = data['Social Media Use'].replace(socialMedia)
data['Income'] = data['Income'].replace(income)
data['Political standing'] = data['Political
standing'].replace(politicalStanding)
for column in columnsRF:
 data[column] = data[column].replace(realFake)
```

Figure 2. Code of data preparation

After the data is all processed, the machine learning part of the code can finally start. Since there are multiple attributes, it is simpler to use a function. In the function, the data is split randomly so that 70% is used to train and 30% is used to check the answers, though only one attribute is used at a time. Once the data is split, the training data is fed into a linear classification algorithm. The remaining 30% of the data is then fed into the algorithm, the answers checked, and the resulting scores printed. This is repeated for each and every attribute.



```
def training(column):
 #split data between X and Y
 x_Data = data[['Real or Fake','Real or Fake.1', 'Real or Fake.2', 'Real or
Fake.3', 'Real or Fake.4', 'Real or Fake.5', 'Real or Fake.6', 'Real or
Fake.7', 'Real or Fake.8', 'Real or Fake.9']]
 y_Data = data[[column]]

 #split data between train and test
 X_train, X_test, Y_train, Y_test = train_test_split(x_Data, y_Data, test_size
= 0.3, random_state=1)

 #load model
 model = svm.SVC()
 model.fit(X_train, Y_train)

 #train model
 print(model.score(X_test,Y_test))

training('Gender')
training('Education')
training('Age')
training('Main source of news')
training('Social Media Use')
training('Income')
training('Political standing')
```

Figure 3. Code of machine learning

4. EXPERIMENT

The purpose of our study is to determine the factors that contribute to a person's belief in misinformation. In order to determine this, we need to first determine the relevant factors of the participant and whether or not they would believe in misinformation. The easiest way to accomplish the first was to simply ask them for it. This way is the most reliable and it is also easy to get plenty of information from it. The basic information section has ten questions, asking the user their gender, education, age, main source of news, social media use, income and political standing. The second one is slightly more challenging. We decided, instead of simply asking, to ask the participant to determine which ones of the news articles are fake and which ones are real. This way, we get a clearer picture of their decision making process than if we had asked them outright, allowing biases to skew the results. There are ten questions divided into 5 categories: Science, Health, Trivia, Politics and Entertainment/Sports. Each of these categories will have 2 screenshots of a news article including the headline, an image and perhaps a small fragment of the first paragraph or two. Each of these will have 2 possible answers: true or false. This way, it is possible to tell numerous things from the survey. It would be able to tell which area the participant is most interested in, which area is most vulnerable to misinformation and even which political party the participant may be aligned with in the politics section. This study does not take much advantage of this but a future, more in depth study could. The screenshots and news headlines are chosen to be intentionally ridiculous as well to make it harder to distinguish between the real and the fake news articles. Unfortunately, we were not able to get mass responses so we generated our own dummy data instead.

Below is a chart of our experiment results. After generating the dummy data, we needed to develop the best algorithm to process it. As a result, we decided to process the data using different methods to determine the best one. We used SVC, Random Forest Classifier and a Linear Regression Classifier. From the below chart, it would seem that all three methods fared incredibly closely. Since, again, this is dummy data and in no way reflects the real world, the results don't matter much but it is still very clear that all three models agree to an extent. It would seem that, from the dummy data, ender, main source of news and social media use all play a fairly large role in determining a person's decisions to either believe or discount misinformation.

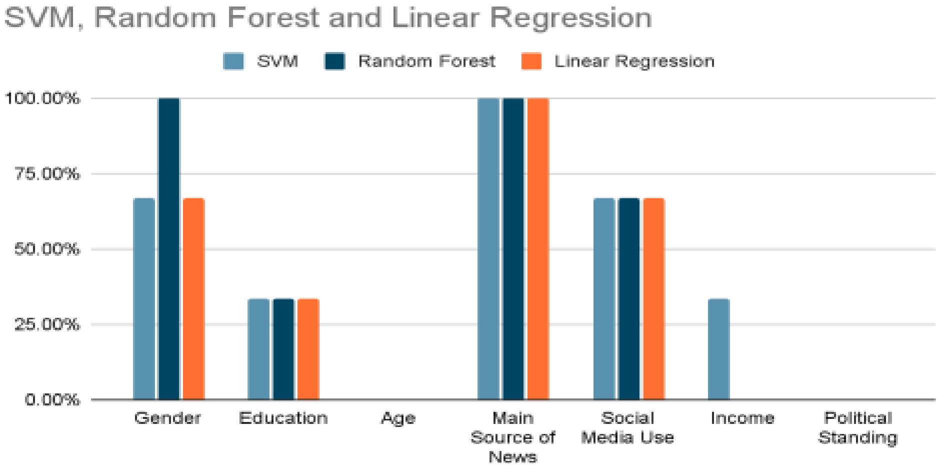


Figure 4. Result of experiment

In the above experiment, we solved the problem of getting participants, getting the basic information of the participants and getting their behavioral patterns. We got around the first by generating dummy data to start. We got around the second by using the simple method of asking them for it and we got around the final one by asking them to classify real and fake news. The experiment was constructed around the goal of training the algorithm to identify which attributes contribute to a participants decision making so the dummy data was generated with clear trends in mind. The main attributes that were focused on were education, main source of news and social media use. These were the ones that were not randomized and instead heavily targeted. They had their answers modified to produce a clear result to prove the algorithm was working. Do keep in mind that this is still dummy data generated for the purpose of developing this algorithm. From the above chart, it is also clear that it worked. The algorithm was able to predict a clear trend in those areas with a few minor deviations.

5. Related Work

This study focuses on how many people actually read an article after they have seen the headline [4]. Instead of a survey, they simply divided the amount of people who actually clicked on the url by the people that saw the post. Through this method, it is much easier to collect a large sample of data and it will have little to no influence from biases. Our method is much more detailed but is otherwise subject to the personal biases of the participant. An example of this would be on the political standings question. The participant may feel like they belong in one group but may instead normally be classified in another.

This study functions in an incredibly similar fashion to this one [5]. They were asked whether or not they would believe health information if they received it from a source such as WeChat. This is very direct and very simple so it is hard to mess up on. The problem with this is also one that we faced: how do we know they are being honest? Many can claim to not believe in information unless provided with a credible source but putting it into practice is another thing altogether. This survey does cut down on that slightly by using more indirect methods to probe the participant for information rather than asking outright but certain questions, such as the basic information, still requires honesty.

This study uses a very similar approach as this one [6]. It also uses a survey of sorts and focuses on WhatsApp. The information of the participants (Age and Occupation) are taken and their responses filed under those two categories. The questions themselves ask the participant to identify which messages contain real information and which ones don't. One of the messages will also have a link of some kind to source materials while the other won't. A "score" is then calculated using what the participant thought were true or false.

6. CONCLUSIONS

These algorithms have numerous applications in the real world. With enough data, we will be able to safely and accurately predict which groups are most vulnerable to misinformation. The structure of the survey itself could help with future studies [1]. The method by which the news articles are presented and the news articles itself. The articles are intentionally misleading and ridiculous so that it cannot be immediately determined which one is real and which is not.

One large challenge that we faced in this study was actually getting responses to the survey. We did not have the resources to send this out at a large-scale and collect that many replies [2]. As a result, we decided to use dummy data as a proof of concept and to develop the algorithm. Another limitation is the reliance of the participant itself to provide information. While the rest of the survey attempts to get around this by using indirect methods, the basic information still relies on the honesty of the participants. This also means that any information they give us will be subject to bias as well.

The issue of getting the survey out is not too big of an issue to solve [3]. There are plenty of ways to get many people from taking a survey ranging from paid survey companies to sending out a post on social media. The issue of honesty is harder to solve without violating privacy concerns. Another simple solution would be to ask their friends or family to describe them or ask them to describe why they wrote down what they did.

REFERENCES

[1] Fioranelli, M., et al. "5G Technology and induction of coronavirus in skin cells." (2020).

[2] Lal, P., et al. "Edible vaccines: current status and future." Indian journal of medical microbiology 25.2 (2007): 93-102.

[3] Stribling, Jeremy, Max Krohn, and Dan Aguayo. "Scigen-an automatic cs paper generator." (2005).

[4] Gabielkov, Maksym, et al. "Social clicks: What and who gets read on Twitter?." Proceedings of the 2016 ACM SIGMETRICS international conference on measurement and modeling of computer science. 2016.

[5] Pan, Wenjing, Diyi Liu, and Jie Fang. "An Examination of Factors Contributing to the Acceptance of Online Health Misinformation." Frontiers in Psychology 12 (2021): 524.

[6] Bapaye, Jay Amol, and Harsh Amol Bapaye. "Demographic Factors Influencing the Impact of Coronavirus Related Misinformation on WhatsApp: Cross-sectional Questionnaire Study." JMIR public health and surveillance 7.1 (2021): e19858.

[7] Carnegie, Andrew. "Wealth." The North American Review 148.391 (1889): 653-664.

[8] Schmidt, Eric, and Jonathan Rosenberg. How google works. Grand Central Publishing, 2014.

[9] Godfrey-Smith, Peter. "Misinformation." Canadian Journal of Philosophy 19.4 (1989): 533-550.

[10] O'Hara, Maureen. "What is a quote?." The Journal of Trading 5.2 (2010): 10-16.

[11] Iarovici, Edith, and Rodica Amel. "The strategy of the headline." (1989): 441-460.

[12] Cox, David R. "Interaction." International Statistical Review/Revue Internationale de Statistique (1984): 1-24.

[13] Moschovakis, Yiannis N. "What is an algorithm?." Mathematics unlimited—2001 and beyond. Springer, Berlin, Heidelberg, 2001. 919-936.

[14] Norvig, P. Russel, and S. Artificial Intelligence. A modern approach. Upper Saddle River, NJ, USA:: Prentice Hall, 2002.

[15] Coombs, Clyde H. "A theory of data." (1964)

EMOTION CLASSIFICATION USING 1D-CNN AND RNN BASED ON DEAP DATASET

Farhad Zamani and Retno Wulansari

Telkom Corporate University Center, Telkom Indonesia, Bandung, Indonesia

ABSTRACT

Recently, emotion recognition began to be implemented in the industry and human resource field. In the time we can perceive the emotional state of the employee, the employer could gain benefits from it as they could improve the quality of decision makings regarding their employee. Hence, this subject would become an embryo for emotion recognition tasks in the human resource field. In a fact, emotion recognition has become an important topic of research, especially one based on physiological signals, such as EEG. One of the reasons is due to the availability of EEG datasets that can be widely used by researchers. Moreover, the development of many machine learning methods has been significantly contributed to this research topic over time. Here, we investigated the classification method for emotion and propose two models to address this task, which are a hybrid of two deep learning architectures: One-Dimensional Convolutional Neural Network (CNN-1D) and Recurrent Neural Network (RNN). We implement Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) in the RNN architecture, that specifically designed to address the vanishing gradient problem which usually becomes an issue in the time-series dataset. We use this model to classify four emotional regions from the valence-arousal plane: High Valence High Arousal (HVHA), High Valence Low Arousal (HVLA), Low Valence High Arousal (LVHA), and Low Valence Low Arousal (LVLA). This experiment was implemented on the well-known DEAP dataset. Experimental results show that proposed methods achieve a training accuracy of 96.3% and 97.8% in the 1DCNN-GRU model and 1DCNN-LSTM model, respectively. Therefore, both models are quite robust to perform this emotion classification task.

KEYWORDS

Emotion Recognition, 1D Convolutional Neural Network, LSTM, GRU, DEAP.

1. INTRODUCTION

Many industries and human resource fields began to implement emotional recognition of the employee in their organization. When they can assess the emotional state of the employee, the human resource could gain advantages from it as they could improve the quality of decision makings regarding their employee. This subject could become a base for emotion recognition tasks in human resource cases. For this reason, this task will become important widely used shortly.

Emotions play a crucial role in human beings as these are associated with neuro physiological aspects that also correspond to a coordinated set of responses, which may include verbal, behavioral, physiological, and neural mechanisms. In another perspective, the emotion can be mapped into the Valence, Arousal, and Dominance (VAD) dimensions. Valence represents a dimension that corresponds to the level of pleasantness that goes from very positive (pleasure) to very negative (displeasure). On the other hand, arousal is the intensity level of emotion that an

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 363-378, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112328

event creates, ranging from excited (positive) to calm (negative). Lastly, dominance is the degree of control exerted by a stimulus[1]. The most common model used is the Circumplex Model of Affect, which only describes emotion in the valence and arousal dimension [2].

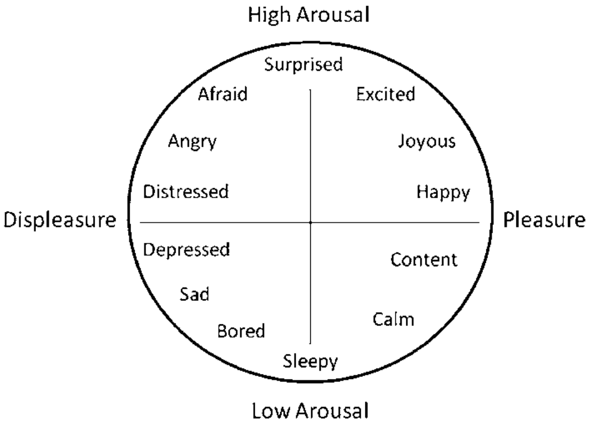


Figure 1. Circumplex Model of Affect Graphical Representation.

In recent studies, human emotion can be recognized based on their peripheral physiological signal, such as electroencephalogram. The electroencephalogram (EEG) is defined as the electrical activity of an alternating type recorded from the scalp surface after being picked up by metal electrodes and conductive media [3]. A typical amplitude of EEG ranged from 0.5 to 100uV. In general, this EEG signal is divided into a specific range of frequencies, namely delta, theta, alpha, beta, and gamma. Each range is associated with a particular condition or state of the brain.

Recently, numerous researchers have been using EEG signals to aid them to classify emotional states. Moreover, various EEG datasets that can be used in emotion recognition studies have been published by many researchers. DEAP dataset is commonly used for this study. Several signal processing methods, as well as machine learning algorithms, are applied in many studies.

In this study, we suggest simpler signal processing methods to prepare the DEAP dataset before we feed it into our models. We also propose a combination of two neural network architectures as our model, which is CNN and RNN. One-Dimensional CNN is chosen as our CNN architecture, and for the RNN, we will evaluate the performance of GRU and LSTM which we will put together with 1D-CNN. Subsequently, the accuracy and loss value in the training, validation, and testing process of both models, 1D-CNN + GRU and 1D-CNN + LSTM, will be compared.

The layout of the paper is as follows. In section 2, we overview several approaches related to our research topic. The materials and methods of the experiment are described in detail in section 3. The results of the experiment and discussions are covered in Section 4. The conclusion, limitation and future attempt of this work follows in Section 5.

2. RELATED WORKS

Xiang Li et al. (2016) proposed a hybrid deep learning model, C-RNN, which integrates CNN and RNN, for emotion recognition [4]. The data is pre-processed using continuous wavelet transform and frame construction before being trained in the model. This experiment performance is 74.12% and 72.06% for arousal and valence dimensions, respectively. Alhagry et al. (2017) suggested LSTM as its deep learning method for emotion recognition tasks based on the DEAP dataset [5]. The average accuracy given from this method is 85.65%, 85.45%, and 87.99% with arousal, valence, and liking classes, respectively.

Lin et al. (2017) presented an approach to perform emotional states classification by end-to-end learning of deep CNN based on the DEAP dataset [6]. The datasets were transformed into six gray images which contain time and frequency domain information, then the feature extracted before trained using the AlexNet model. The model accuracy of this study is 87.30% and 85.50% for arousal and valence, respectively. Li et al. (2017) applied a combined CNN and LSTM RNN (CLRNN) in their emotion recognition task and carried it out with the DEAP dataset [7]. The dataset is transformed into a Multidimensional Feature Image sequence beforehand. The average emotion classification accuracy of each subject with the hybrid neural networks proposed in the study is 75.21%.

Acharya et al. (2020) compared the performance of LSTM and CNN models to carry out emotion classification tasks [8]. The dataset used in this study is the DEAP dataset and the feature extraction method implemented is Fast Fourier Transform. The best result obtained from the experiment was from the LSTM model with 88.6% accuracy on the liking emotion and 87.2% accuracy when using the CNN model. Zhang et al. (2020) investigated the application of several deep learning models to the research field of EEG-based emotion recognition: DNN, CNN, LSTM, and a hybrid model of CNN and LSTM [9]. This research also used the DEAP dataset, and several features were extracted from it, such as mean, standard deviation, max value, min value, skewness, and kurtosis. The CNN and CNN-LSTM models displayed the best performance to perform this task with 90.12% and 94.17% accuracy, respectively.

Anubhav et al. (2020) studied the EEG signals in scope for developing a headgear model for real-time monitoring of emotions [10]. Band power and frequency domain features were extracted from the DEAP dataset and compared the classification accuracies for valence and arousal dimensions. In this study, the LSTM model achieves the best classification accuracy of 94.69% and 93,13% for valence and arousal, respectively. Dar et al. (2020) proposed 2D CNN architecture to process EEG signals for emotion recognition tasks [11]. The datasets used in this experiment are DREAMER and AMIGOS, in which both data are converted into a 2D feature matrix (PNG format) before the data is fed into CNN. Aside from EEG, other peripheral physiological signals such as ECG and GSR are also utilized for the multi-modal emotion recognition process. Accuracy acquired using EEG modality only is 76.65%, and the overall highest accuracy of 99.0% for the AMIGOS dataset and 90.8% for the DREAMER dataset is achieved with multi-modal fusion.

3. MATERIALS AND METHODS

Firstly, this section will elaborate on the details of datasets used in this study as well as how the data will be classified. Then, the proposed methodology for the experiment will be discussed in detail.

3.1. Diagram Block

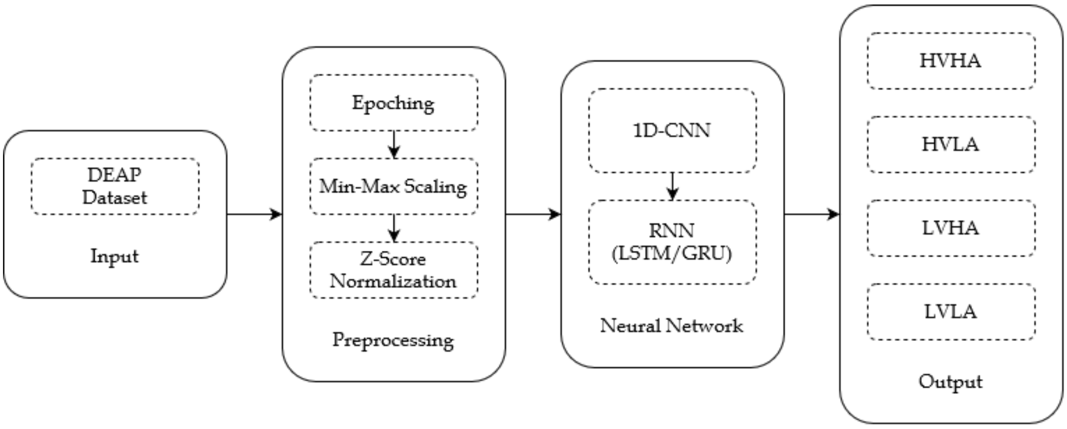


Figure 2. Diagram Block of Our Proposed Method

As shown in Figure 1, the framework for this study is made up of two main processes, the data pre-processing part, and the neural network part. The details of each process will be explained in the following section.

3.2. Datasets and Label Classification

The dataset used in this study is DEAP which was made publicly available for researchers to test their affective state estimation method [12]. This dataset contains EEG and other peripheral physiological signals, such as EOG and EMG. The data was acquired from 32 participants as each watched 40 one-minute-long music videos. Participants rated each video in terms of the levels of arousal, valence, like/dislike, dominance, and familiarity.

The DEAP dataset contains 32 groups of EEG data in total, corresponding to the experimental data of 32 subjects (s01–s32). For each subject, EEG signals are acquired from 32 channels and 8 additional channels are from other peripheral signals, which makes 40 channels in total. The data of each subject contains two arrays: data and labels. The structure for each file in this dataset is shown in Table 1.

Table 1. Dataset Structure.

Array Name	Array Shape	Array Contents
data	40 x 40 x 8064	video/trial x channel x data
labels	40 x 4	video/trial x label (valence, arousal, dominance, liking)

In this study, we used a pre-processed version of the DEAP dataset. This version of the signal has been down sampled to 128 Hz and filtered by a band-pass filter with a frequency bandwidth of 4-45 Hz. EOG artifacts were also removed, and the data were averaged to the common reference.

Each video/trial was rated on a scale from 1 to 9. The lower the rating, the weaker the emotion, and the higher the rating, the stronger the emotion. The dataset is divided into 4 classes based on the threshold of valence and arousal value, then they will be classified as the following: High-Valence High-Arousal (HVHA), High-Valence Low-Arousal (HVLA), Low-Valence High-

Arousal (LVHA), and Low-Valence Low-Arousal (LVLA). The threshold used in this classification is 5. If the value is greater than 5, it will be classified as high, otherwise is low.

Table 2. Label Classification.

Label	HVHA	HVLA	LVHA	LVLA
Valence	>5	>5	<=5	<=5
Arousal	>5	<=5	>5	<=5

To reduce the computational cost for this study, not all channels are selected. Only 14 channels which we further process for emotion classification task [13]. This channel selection is based on the significance of the brain region which represent emotional states. Those channels are listed below in Table 3. This task was done after we classify the label into four categories: HVHA, HVLA, LVHA, LVLA. As a result, it will increase the training accuracy as well as decrease the validation loss.

Table 3. Channel Selection.

Channel	Index
AF3	1
F3	2
F7	3
FC5	4
T7	7
P7	11
O1	13

Channel	Index
AF4	17
F4	19
F8	20
FC6	21
T8	25
P8	29
O2	31

3.3. Data Pre-Processing

To prepare the DEAP dataset before we feed it into our proposed model, it must be processed first to get a better training result. In the following sections, our data pre-processing methods will be described in detail.

3.3.1. Epoching

Epoching is a procedure in which specific time windows are extracted from the continuous EEG signal. To begin with, this task is performed by extracting 2 seconds wavelengths of each signal with 2 second time step. Therefore, it will give us 31 signals for each channel, with 256 data points for each epoch that represents 2 seconds of the signal.

Figure 3 is a sample of the signal after the epoching process.

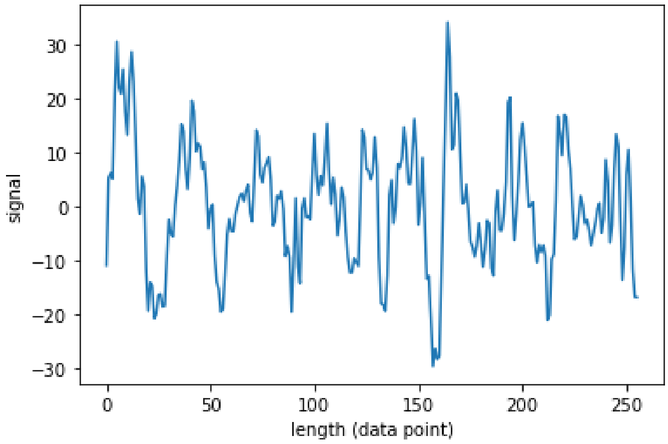


Figure 3. Sample Signal after Epoching

3.3.2. Standardization

Every signal in the dataset that is used in this study will be standardized first through two methods: z-score normalization and followed by min-max scaling. These two processes are carried out to prevent overfitting as well as increase the accuracy of the model.

3.3.2.1. Min-Max Scaling

Min-Max Scaling (often also simply called normalization) is a standardization method to scale data to a fixed range: 0 to 1. The min-max scaling is calculated using the following equation.

$$Xnorm = \frac{X - Xmin}{Xmax - Xmin} \tag{1}$$

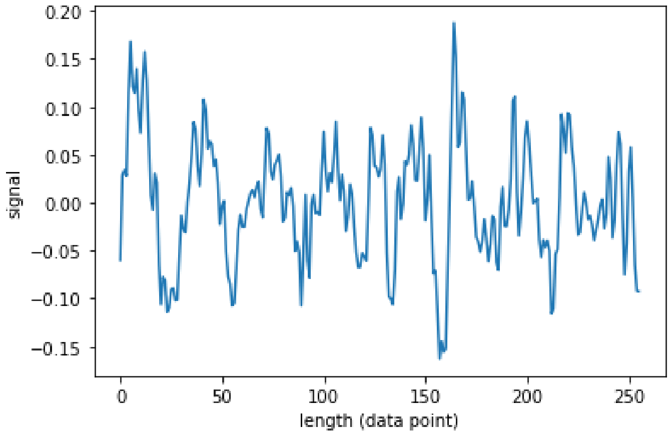


Figure 4. Signal Sample after Min-Max Scaling Normalization

3.3.2.2. Z-Score Normalization

The result of standardization (also called z-score normalization) is that the feature will be rescaled so that they will have the properties of a standard normal distribution with:

$$\mu = 0 \; and \; \sigma = 1 \tag{2}$$

where μ is the mean (average) and σ is the standard deviation from the mean; standard scores (also called z scores) of the samples are calculated as follows:

$$z = \frac{x - \mu}{\sigma} \tag{3}$$

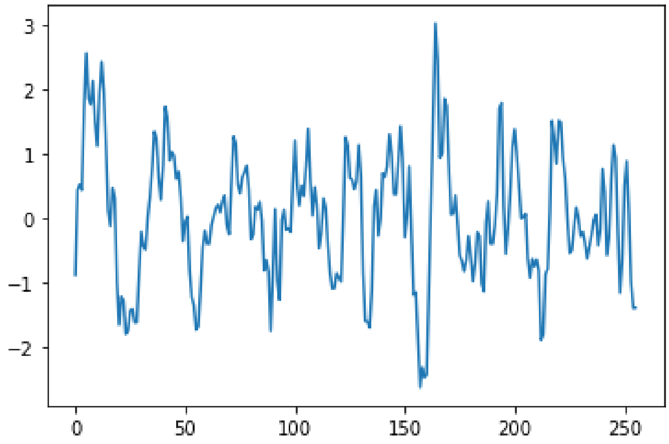


Figure 5. Sample Signal after Z-Score Normalization

3.4. Neural Network Architecture

The neural network architecture that will be used in the proposed method is One-Dimensional Convolutional Neural Network and Recurrent Neural Network. The overview of each architecture will be portrayed in brief in the following section.

3.4.1. 1D-CNN

In this study, all signals used are one-dimensional. One-Dimensional Convolutional Neural Network (1D-CNN) has become a very powerful tool to be used upon time-series data like EEG signals. This model can extract and distinguish several features of EEG data which represent significant features for emotion classification tasks.

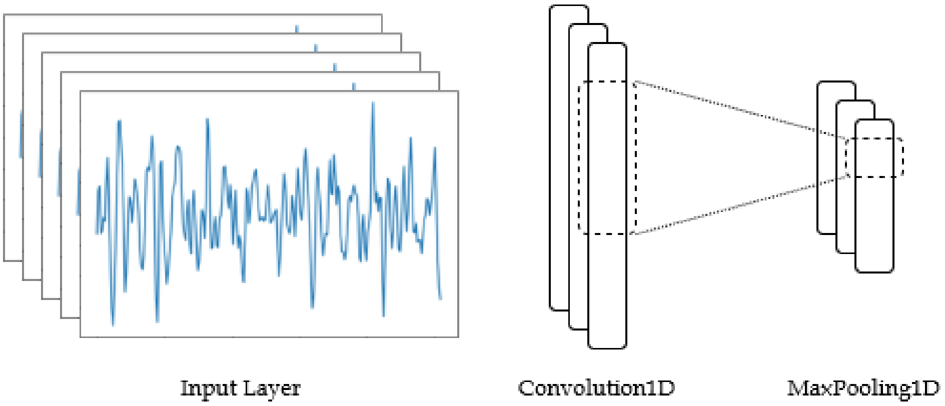


Figure 6. 1D-CNN Structure.

3.4.2. LSTM

Simple Recurrent Neural Network suffers from short-term memory and is often called vanishing gradient problem. LSTM is a specialized version of RNN that was designed to overcome this shortcoming. LSTM was first introduced by Hochreiter (1997)[14], and this solution works well on a large variety of problems. The core concept of LSTM's are the cell state and its various gates: forget gate, input gate, and output gate.

Cell state act as a memory of the network which carries relevant information throughout the processing of the sequence from earlier steps to later steps. Information gets added via gates composed out of the sigmoid layer and point wise multiplication operation. LSTM model can learn which information is relevant to keep or forget during the training.

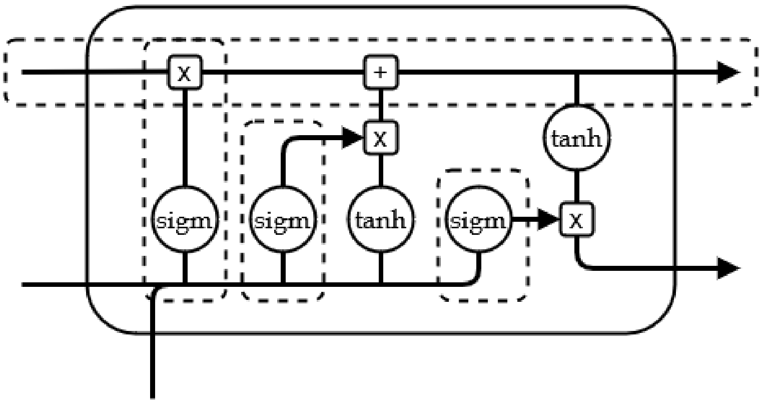


Figure 7. LSTM Cell Structure.

3.4.3. GRU

GRU is the newer generation of Recurrent Neural Network and is similar to an LSTM[15]. GRU's got rid of the cell state ad used the hidden state to transfer information within the network. It only has two gates. The first is the reset gate, which is used to decide how much past information to forget. The second is, instead of use forget gate and input gate, GRU uses a single update gate. It also merges the cell state and hidden state. As a result, this model is much simpler than LSTM model.

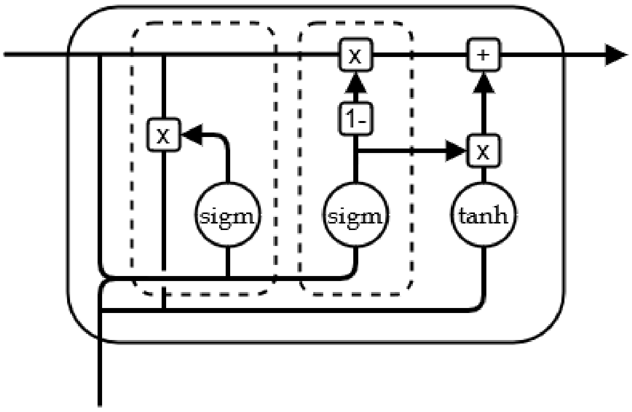


Figure 8. GRU Cell Structure.

3.5. Proposed Model

3.5.1. Data Shape

For this study, we use all of the preprocessed DEAP dataset from 32 subjects. The preprocessed dataset went through further preprocessing procedures and label classification. Then, we randomly split the dataset into train, validation, and test data with 60:20:20 ratio. The data shape for our experiment is shown in Table 4.

Table 4. Data Shape.

Set	Data	Target	Percentage
Train Set	(333312, 256, 1)	(333312, 4)	60%
Validation Set	(111104, 256, 1)	(111104, 4)	20%
Test Set	(111104, 256, 1)	(111104, 4)	20%

3.5.2. Hyperparameter

To get a good result, hyperparameter must be tuned adequately. The batch size finalized for our architectures is 256 and the optimizer used is Adam. The loss function implemented for updating the weights during backpropagation is categorical cross-entropy since we use a multi-labeled dataset. Furthermore, we use the callbacks method with a patience value of 10 which monitors validation loss value. As a result, the epoch for the two models will be varied depends on the validation loss value.

Table 5. Hyperparameter.

Parameters	Value
Batch Size	256
Optimizer	Adam
Loss Function	Categorical cross entropy

3.5.3. 1D-CNN – GRU

This model is a composite of two kinds of deep learning structures: 1D-CNN and GRU. The structure of the model is presented in Table 6. The input size of the networks is 256 x 1, that correspond with two seconds of the epoched signal which contains 256 data point. We set the number of convolutional filters as 128 with ReLU as its activation layer. Following the first convolutional layer is a max-pooling layer with a pool size of 2. We set a dropout of 0.2 after the max-pooling layer to reduce the probability of overfitting. The second convolutional layer is the same as the first.

The convolutional layers are followed by the GRU layer with 256 units and 32 units, subsequently. After every GRU layer, we also set a dropout layer of 0.2. Before connecting to the dense unit, a flattening operation is implemented to transform features into a one-dimensional feature vector. The dense layer units are set to 32 with the ReLU activation function. The last layer is 4 units dense layer that represents four labels of classification. The activation function of this dense layer is softmax. The number of trainable parameters of this model is 378.820. The overview of this model configuration is illustrated in Table 6.

Table 6. Configuration of 1D-CNN – GRU.

Layer	**Parameter**
InputLayer	input_shape = (256,1)
Conv1D + ReLU	units = 128, kernel = 3
MaxPooling1D	pool_size = 2
Dropout	rate = 0.2
Conv1D + ReLU	units = 128, kernel = 3
MaxPooling1D	pool_size = 2
Dropout	rate = 0.2
GRU	units = 256, return_sequences = True
Dropout	rate = 0.2
GRU	units = 32
Dropout	rate = 0.2
Flatten	none
Dense + ReLU	units = 128
Dense + Softmax	units = 4

3.5.4. 1D-CNN – LSTM

The configuration of this model is identical to 1D-CNN – GRU. The only difference is, we substitute the GRU layer with the LSTM layer. The number of trainable parameters of this model is 485.764, which is larger than that of 1DCNN-GRU model. The overview of this model configuration is summarized as shown in Table 7.

Table 7. Configuration of 1D-CNN – LSTM.

Layer	**Parameter**
InputLayer	input_shape = (256,1)
Conv1D + ReLU	units = 128, kernel = 3
MaxPooling1D	pool_size = 2
Dropout	rate = 0.2
Conv1D + ReLU	units = 128, kernel = 3
MaxPooling1D	pool_size = 2
Dropout	rate = 0.2
LSTM	units = 256, return_sequences = True
Dropout	rate = 0.2
LSTM	units = 32
Dropout	rate = 0.2
Flatten	none
Dense + ReLU	units = 128
Dense + Softmax	units = 4

4. Experimental Results

In this section, we present the results of our experiments. Firstly, we will elaborate on the performance of both models used in this study based on the accuracy and loss value of the model. Following this, we present the comparison of the results between GRU and LSTM that is used in this experiment. Later, the overall results and analysis of the experiment will be explained in detail. Lastly, we discussed limitations of the proposed method.

All the experiments are implemented by using the Tensorflow framework version 2.5.0 and Python version 3.7.10. The workstation used is MSI GF65 Thin 10SDR, which included Intel i7-10750H (12 CPUs @ 2.6 GHz), NVIDIA GeForce GTX 1660 Ti for GPU acceleration, 500 GB SSD, and 8 GB RAM.

The first experiment was designed to evaluate the performance of the 1D-CNN – GRU model for the emotion classification task. The result is shown in Table 8. We obtained training accuracy of 96.3%, along with validation and test accuracy above 99%. Additionally, we also acquire training, validation, and test loss of 10.7%, 0.7%, and 0.6%, respectively. Compared with previous relevant works, this model shows a very good performance.

As illustrated in Figure 9, train and validation graphs show the desired value of accuracy and loss change per epoch. There is no overfitting occurred and the training process stopped at 165 epochs. The average time consumed for training per epoch is 50 seconds. Furthermore, we also made a confusion matrix to analyze the difference between the predicted and the actual value (Figure 10). Moreover, this technique is good to be applied to measure the multilabel classification performance. We got F1-Score and recall value of 1 which infer that the quality of this model is good.

Table 8. 1D-CNN – GRU experimental results.

GRU	**Training**	**Validation**	**Test**
Accuracy	96.3%	99.9%	99.9%
Loss	10.7%	0.7%	0.6%

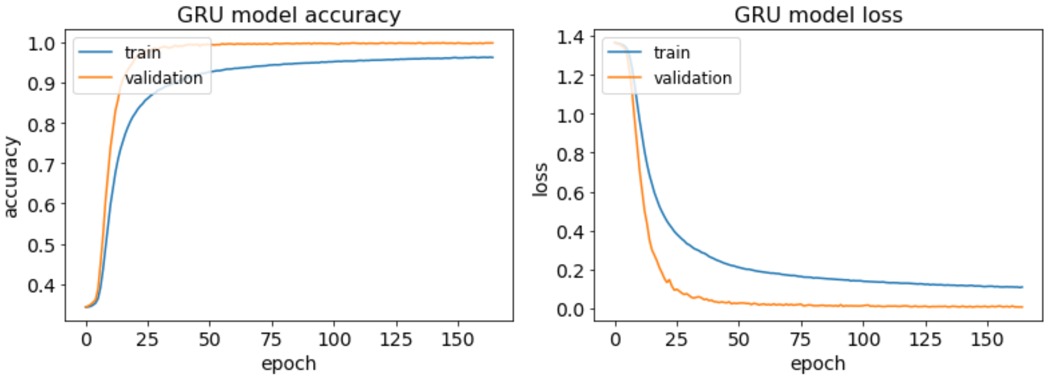


Figure 9. Train and Validation Graph of 1D-CNN – GRU model.

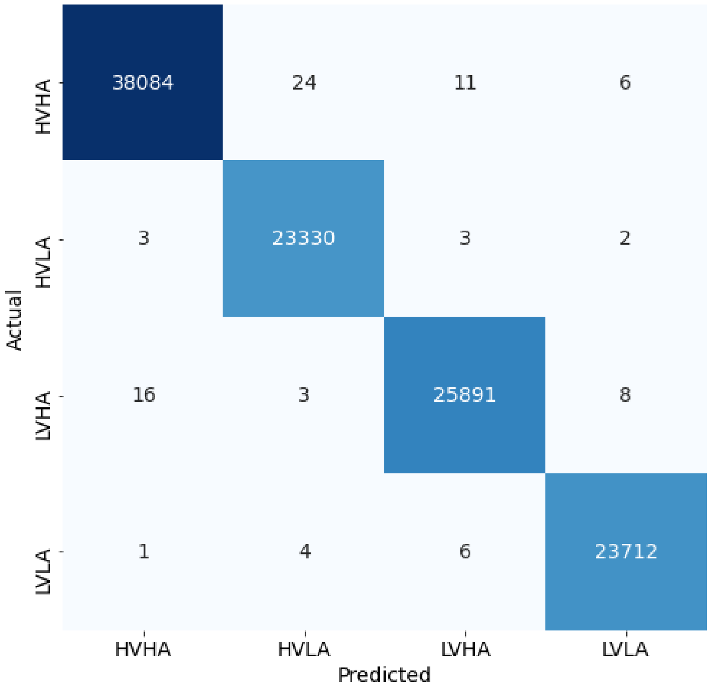


Figure 10. Confusion Matrix for 1D-CNN – GRU model

The second experiment was designed to evaluate the performance of the 1D-CNN – LSTM model to carry out the emotion classification task. Table 9 illustrates the result of this experiment. We obtained a training accuracy of 97.8%. The validation and test accuracy values of this model are 99.9% and 99.9% respectively which is above the first model. Aside from that, we acquire a training loss of 6.7%, with validation and test loss of 0.1% and 0.1%.

Table 9. 1D-CNN – LSTM experimental results.

LSTM	**Training**	**Validation**	**Test**
Accuracy	97.8%	99.9%	99.9%
Loss	6.7%	0.1%	0.1%

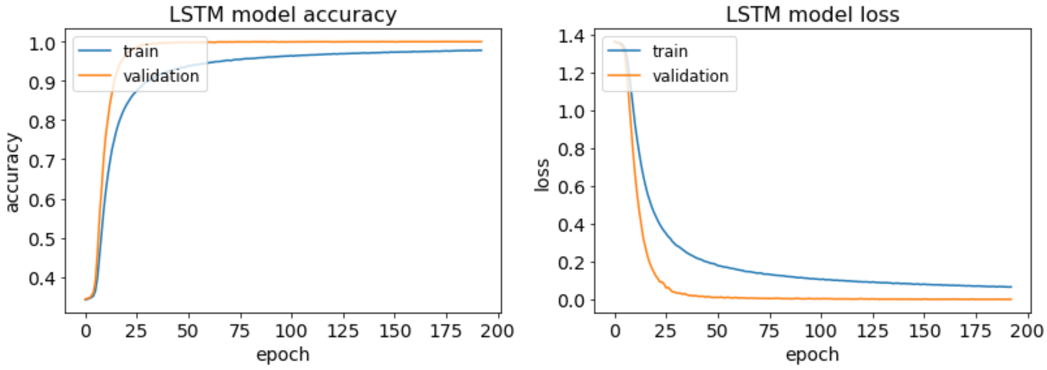


Figure 11. Train and Validation Graph of 1D-CNN – LSTM model.

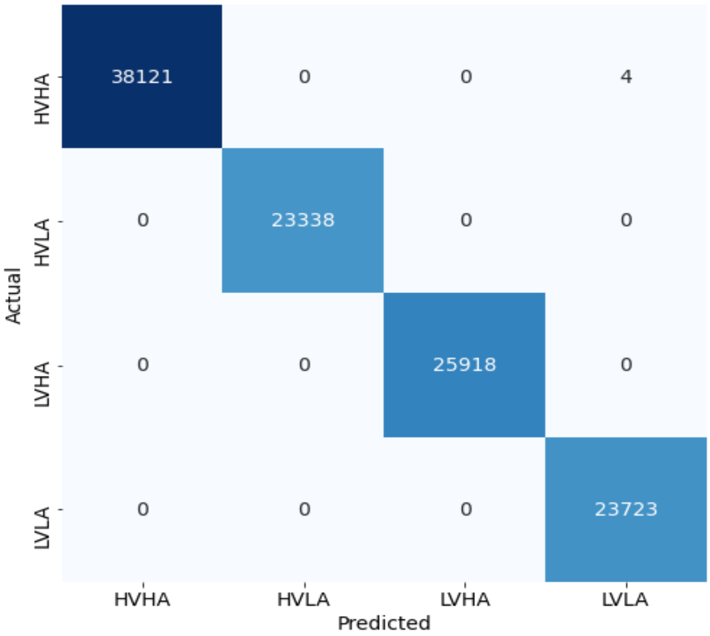


Figure 12. Confusion Matrix for 1D-CNN – LSTM model.

In Figure 11, we can see that train and validation graphs also show the desired value change per epoch. In this model, we can avoid overfitting as well. The training process stopped at 193 epochs with an average training time per epoch is 69 seconds, more than the 1D-CNN – GRU model. We also analyze the difference between the predicted value and actual value using a confusion matrix (Figure 12). We got an F1-Score and a recall value of 1. It concludes that the overall quality of this model is better than the previous model, albeit the training time is longer.

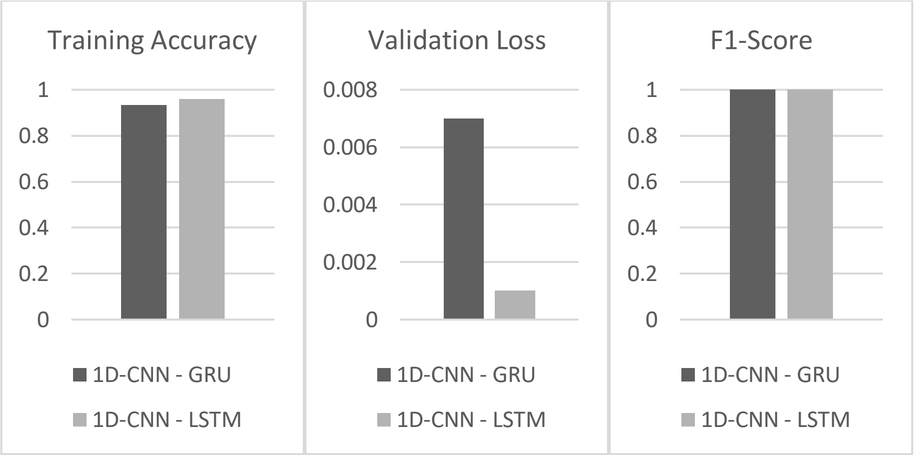


Figure 13. Performance chart comparison based on training accuracy, validation loss, and f1-score value.

If we compare the performance of two models, 1D-CNN – GRU and 1D-CNN – LSTM, it is clear the latter is better than the former. More specifically, if we take account of the training accuracy and validation loss, the 1D-CNN – LSTM model performance surpasses the other one. The F1-score of the 1D-CNN – LSTM model is the same with 1D-CNN – GRU, with a score of 1. The comparison of both models is shown in Figure 13.

Through this experiments, it was proven that One-Dimensional Convolutional Neural Network can be regarded as a feature extractor that can recognize a hidden patterns or features of the EEG signal. Even though we do not perform Fast Fourier Transform (FFT), Continuous Wavelet Transform (CWT), or other feature extraction methods to acquire a particular features of the signal, we can conclude that this methods still could execute this task very well. Furthermore, the proposed methods, from the preprocessing to the classification task, do not require large memory to be executed since the methods is relatively simple. Besides that, since we only select 14 channels from DEAP dataset, the overall training speed was quite fast, and the model can achieve convergence before 100 epochs.

The overall performance of our proposed method outperformed the previous approaches that have been explained before in related works section. Each of these methods implemented various preprocessing and deep learning technique to perform the EEG feature extraction and emotion classification tasks. All the methods we compared use the DEAP dataset, and the classification accuracy comparison for each method is illustrated in Table 10.

Table 10. Methods and classification accuracy comparison with previous related works which used DEAP dataset.

Study	**Methodology**	**Classification Accuracy**
[5]	LSTM	85.65% (A), 85.45% (V)
[6]	CWT, CNN	87.30% (A), 85.30% (V)
[7]	CNN-LSTM	75.21%
[8]	FFT, CNN	84.7% (A), 85.01% (V)
[8]	FFT, LSTM	81.91% (A, 84.39% (V)
[9]	CNN-LSTM	94.17%
[10]	Welch, LSTM	93.13% (A), 94.69% (V)
Proposed study	1DCNN-GRU	96.3%
Proposed study	1DCNN-LSTM	97.8%

5. Conclusion, Limitation, and Future Works

In this paper, we present simpler signal processing methods as well as 1D-CNN – RNN hybrid model to perform emotion classification tasks on the DEAP dataset. The results of both models show that proposed models can achieve very high accuracy in classifying EEG signals based on four emotion states: HVHA, HVLA, LVHA, and LVLA. The 1DCNN-LSTM architecture with 97.8% classification accuracy is slightly better than the 1DCNN-GRU architecture that has 96.3% classification accuracy. The 1D-CNN architecture has shown its quality to extract EEG signal features and the RNN architectures, GRU and LSTM, are able to perform well in learning and processing the sequence data of time-series signals.

Although the proposed model has achieved significant results in emotion classification task, we still need to evaluate it further in other datasets and improve the robustness of our model. One of the advantages of our proposed model is its simplicity. Because of that, this model do not require data with numerous signal preprocessing before fed into the model. However, there are still limitation that need to be addressed in the next work. More specifically, this model only trained using DEAP dataset and still not tested in other subjects. Therefore, in our future work, we intend to re-train and test this model using of another EEG datasets that used in emotion recognition study, such as DREAMER and AMIGOS. Moreover, we plan to implement a k-fold cross validation in the next work in order to give more accurate estimation of performance of our proposed method. Lastly, we aim to use this model to assess emotional states of a subject and integrate it in a real-time application. The results of the experiment suggest that it is a good start to begin implementing this work in the human resources field, especially to gain insights from employee states of emotion.

ACKNOWLEDGMENT

This research work is performed under the Bio signal and AI Research Team of Telkom Corporate University Center, Telkom Indonesia, Bandung, Indonesia.

REFERENCES

[1] S. M. Alarcão and M. J. Fonseca, "Emotions Recognition Using EEG Signals: A Survey," in IEEE Transactions on Affective Computing, vol. 10, no. 3, pp. 374-393, 1 July-Sept. 2019, doi: 10.1109/TAFFC.2017.2714671.

[2] Tseng, Angela & Bansal, Ravi & Liu, Jun & Gerber, Andrew & Goh, Suzanne & Posner, Jonathan & Colibazzi, Tiziano & Algermissen, Molly & Chiang, I-Chin & Russell, James & Peterson, Bradley. (2013). Using the Circumplex Model of Affect to Study Valence and Arousal Ratings of Emotional Faces by Children and Adults with Autism Spectrum Disorders. Journal of autism and developmental disorders. 44. 10.1007/s10803-013-1993-6.

[3] M. Teplan, “Fundamentals of EEG Measurement,” IEEE Measurement Science Review, Vol. 2, 2002, pp. 1-11.

[4] X. Li, D. Song, P. Zhang, G. Yu, Y. Hou and B. Hu, "Emotion recognition from multi-channel EEG data through Convolutional Recurrent Neural Network," 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2016, pp. 352-359, doi: 10.1109/BIBM.2016.7822545.

[5] Salma Alhagry, Aly Aly Fahmy and Reda A. El-Khoribi, “Emotion Recognition based on EEG using LSTM Recurrent Neural Network” International Journal of Advanced Computer Science and Applications(ijacsa), 8(10), 2017.

[6] Lin W., Li C., Sun S. (2017) Deep Convolutional Neural Network for Emotion Recognition Using EEG and Peripheral Physiological Signal. In: Zhao Y., Kong X., Taubman D. (eds) Image and Graphics. ICIG 2017. Lecture Notes in Computer Science, vol 10667. Springer, Cham.

[7] Li Y, Huang J, Zhou H, Zhong N. Human Emotion Recognition with Electroencephalographic Multidimensional Features by Hybrid Deep Neural Networks. Applied Sciences. 2017; 7(10):1060.

[8] Acharya D. et al. (2021) Multi-class Emotion Classification Using EEG Signals. In: Garg D., Wong K., Sarangapani J., Gupta S.K. (eds) Advanced Computing. IACC 2020. Communications in Computer and Information Science, vol 1367. Springer, Singapore.

[9] Zhang Y, Chen J, Tan JH, Chen Y, Chen Y, Li D, Yang L, Su J, Huang X and Che W (2020) An Investigation of Deep Learning Models for EEG-Based Emotion Recognition. *Front. Neurosci.* 14:622759. doi: 10.3389/fnins.2020.622759

[10] Anubhav, D. Nath, M. Singh, D. Sethia, D. Kalra and S. Indu, "An Efficient Approach to EEG-Based Emotion Recognition using LSTM Network," 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), 2020, pp. 88-92, doi: 10.1109/CSPA48992.2020.9068691.

[11] Dar MN, Akram MU, Khawaja SG, Pujari AN. CNN and LSTM-Based Emotion Charting Using Physiological Signals. Sensors. 2020; 20(16):4551.

[12] S. Koelstra et al., "DEAP: A Database for Emotion Analysis ;Using Physiological Signals," in IEEE Transactions on Affective Computing, vol. 3, no. 1, pp. 18-31, Jan.-March 2012, doi: 10.1109/T-AFFC.2011.15.

[13] N. K. Al-Qazzaz, M. K. Sabir, S. Ali, S. A. Ahmad and K. Grammer, "Effective EEG Channels for Emotion Identification over the Brain Regions using Differential Evolution Algorithm," 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2019, pp. 4703-4706, doi: 10.1109/EMBC.2019.8856854.

[14] Hochreiter, S. & Schmidhuber, J"urgen, 1997. Long short-term memory. *Neural computation*, 9(8), pp.1735–1780.

[15] Chung, J., Gülçehre, Ç., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. ArXiv, abs/1412.3

AUTHORS

Farhad Zamani is a biomedical engineering graduate from Bandung Institute of Technology. Currently he works as a researcher in Biosignal and Artificial Intelligent Research Team at Telkom Indonesia, Bandung. His research work focuses specifically on the biosignal and machine learning. He also has interests on medical informatics and bioinformatics. In his free time, Farhad ride his bicycle or read a novel.

Retno Wulansari is a researcher for Indonesia Telecommunication and Digital Research Institute at Telkom Corporate University Center, Bandung. Before that, she worked as researcher for advanced technology focused on Artificial Intelligent, Natural Language Processing and Bio Signal Research Team at Telkom Digital Service Division. Her favorit things to do in her free time is gardening and cooking.

A fair allocation algorithm for predictive police patrolling

Isabella Rodas Arango[1], Mateo Dulce Rubio[1], and Álvaro J. Riascos Villegas[1,2]

[1]Quantil
[2]Universidad de los Andes

Abstract

We address the tradeoff of developing good predictive models for police allocation vs. optimally deploying police officers over a city in a way that does not imply an unfair allocation of resources. We modify the fair allocation algorithm of [1] to tackle a real world problem: crime in the city of Bogotá, Colombia. Our approach allows for more sophisticated prediction models and we show that the whole methodology outperforms the current police allocating mechanism in the city. Results show that even with a simple model such as a Kernel Density Estimation of crime, one can have much better prediction than the current police model and, at the same time, mitigate fairness concerns. Although we can not provide general performance guarantees, our results apply to a real life problem and should be seriously considered by policy makers.

Keywords

Predictive Policing, Algorithmic Fairness, Optimal Allocation of Police, Censored Data.

1 Introduction

Police agencies around the world have widely started to include data mining techniques in their decision-making processes. In particular, machine learning algorithms have been designed and implemented in recent decades to predict crime events in cities, which then serve as input for the allocation of scarce law enforcement resources [2]. These predictive patrol models have become a standard in the way police operate in major cities around the world [3].

To take advantage of these developments, Colombia's government has recently introduced changes in the way patrolling is carried out in Bogotá, its capital city. First, they announced the development of machine learning models to predict high-impact crime events tailored to the city's social and criminal dynamics [4]. On the other hand, they are modifying the patrolling strategy from predetermined static geographic units to new dynamic units [5]. The present article is framed in the second effort and investigates how to reassign police patrols to the newly defined patrol areas using machine learning models as input in a fair resource allocation algorithm with partial observation of crime (i.e., censored data).

The use of predictive models to guide the patrolling of a city has caused strong criticism from various organizations, as they have been shown to generate biased predictions and disparate impacts

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 379-388, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112329

on disadvantaged populations. [6] empirically demonstrated that using predictive models trained with biased police data to assign patrols, and then retraining the models including the observed data, generates a feedback loop bias that does not allow the models to recover the underlying crime distribution. This is often the case: police data reflects historical policing patterns and captures complex dynamics between criminality, law enforcement agents and the society.

Some approaches have been proposed to mitigate the undesired effects of these models. For the problem of feedback loop due to censored data [7] stands out. In this paper the authors show that filtering the observed crimes used to retrain the models avoids feedback loop (under the assumption of observing crimes at the true rate of occurrence) and allows the model to recover the true criminal distribution. The proposed filter consists of including a crime in the re-training dataset with probability inverse to that of going to the region in which the crime was observed.

Other proposals add regularization components to the objective function of statistical models to penalize for deviations from some particular definition of fairness. For instance, [8] modifies a widely used model based on self-exciting point processes and penalizes the likelihood function so that the number of police patrols exposed to each racial group is proportional to the share of the race in the total population. In [9] the authors modify a GANs model and include in the learning function a term that penalizes for model miscalibration, arguing that a poorly calibrated model generates over- and under-policed areas. Despite the proposed approaches, none is currently widely accepted by the academic community.

In this paper we build on another approach to this problem proposed by [1]. In their article the authors propose an algorithm for the fair allocation of limited resources, in particular police patrols. The authors consider an allocation as fair if the probability of a crime being discovered in every pair of regions does not differ by more than a threshold α. This notion is similar to the well studied notion of equal opportunity in the machine learning literature (see [10]). With this in mind, the proposed algorithm iteratively constructs allocations to maximize the expected number of crimes discovered by the police by ruling out unfair allocations, i.e., where the difference in the probability of observing crimes between some pair of neighborhoods exceeds α.

Following [1] we allow for censored data in the sense that not all crimes are discovered, but only in proportion to the number of police and crimes per unit area. If there is more police in a particular region than crimes, then all of these are discovered. Otherwise, crimes are randomly discovered in a region depending on the number of police patrols and crimes. This setting allows for a more realistic framework than traditional approaches that ignore altogether this obvious partial observability phenomena in crime data.

We extend the algorithm proposed by [1] in two main ways. First, the authors assume that crime occurrence follows a Poisson distribution, which does not hold for our data for Bogotá. Consequently, we use a Kernel Density Estimation (KDE) model to flexibly learn the distribution of crime occurrence in each region and use it in the allocation algorithm of [1]. Second, we modify the greedy assignment rule such that the algorithm assigns an additional patrol to the region with the highest probability of exhibiting more crimes than its current allocation, rather than the region with the highest marginal probability of having an additional crime.

We run the modified algorithm for different fairness thresholds α and we compare our results with an approximation to the current allocation strategy used in Bogotá, a naive allocation in which patrols are assigned according to the proportion of crime in each region, and with a greedy allocation without taking into account any fairness requirement. In particular, we found that the current allocation method used by Colombia's government taking into the account administrative small regions (a total of 1051 in Bogotá) called *cuadrantes* is particularly unfair and inefficient in comparison to all the methodologies we propose in this paper.

The rest of the article is organized as follows. In the following section we detail the problem setup and the data sources. Section 3 presents a summary of the proposed algorithm in [1], illustrates the use of KDE methodology to flexibly estimate the crime distribution, and discusses the proposed modifications to the algorithm. Section 4 describes the implementation of the algorithm, and presents the results of police allocation in Bogotá against baseline allocation scenarios. The last section concludes and discusses some limitations of the present work and possible future research directions.

2 Problem setup

Bogotá is a relatively wealthy and developed city with 8 million inhabitants. In 2010, the National Plan for Community Policing by Quadrants was created (PNVCC, by its Spanish abbreviation) in which the city was divided into 1051 quadrants (*cuadrantes*) as basic units for police patrolling. Each quadrant is assigned a police patrol with two officers for each police shift [11]. Bogotá thus has 1051 patrols that it allocates uniformly among its 1051 quadrants.

Recently, after an increase in the number of criminal events in the city, the Mayor's office of Bogotá announced modifications to this patrolling system including the introduction of dynamic quadrants to tackle insecurity in the city. This change implies a reduction in the number of quadrants which raises the central question of this paper: *How should the 1051 patrols available in Bogotá be reallocated?*

To answer this question we use the fact that Bogotá is administratively divided into 19 localities (*localidades*),[1] each of them with a Police Station that is in turn in charge of a certain number of quadrants. The spatial distribution of localities and quadrants in Bogotá can be seen in Figure 1. It should be noted that each PNVCC police quadrant belongs to one, and only one, locality.

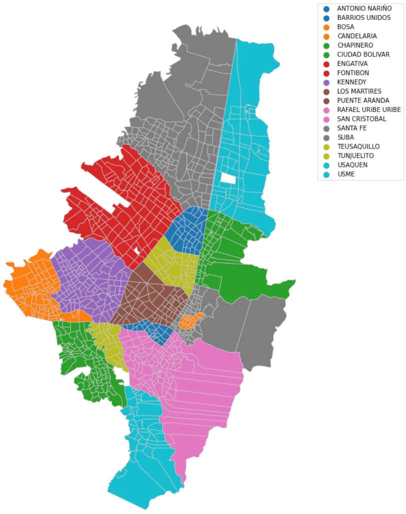


Figure 1: Spatial distribution of localities and quadrants in Bogotá.

With this in mind, we refine our central question to: *How should the 1051 patrols available in Bogotá be reallocated among Bogotá's 19 localities?*

[1] Bogotá is divided into 20 localities but we ignored Sumapaz locality as it is entirely rural and has no quadrants assigned according to the PNVCC.

2.1 Data

We used records of robberies in the city officially registered by the Metropolitan Police of Bogotá. The complete dataset consists of robbery occurrences registered anywhere in Bogotá during 2018 and 2019 which consists of 1,366,712 robbery records.

We aggregate this data weekly for each locality to emulate the weekly allocation of the different patrols in Bogotá. Every locality has a different weekly crime distribution that can be detailed in Figure 2 that shows histograms for the weekly number of crimes recorded for the 19 localities. The empirical crime distributions do not follow a Poisson distribution, therefore the weekly robbery count for each locality is used to train a kernel density estimation model to learn the crime distribution for each urban area.

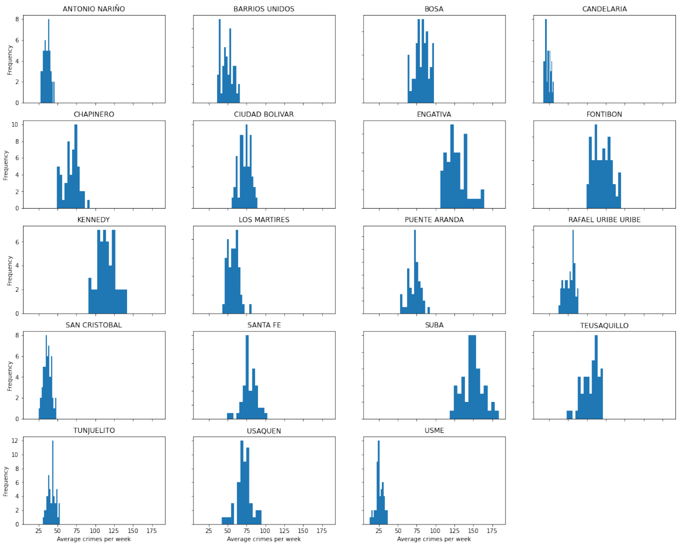


Figure 2: Number of crimes recorded per week for the 19 localities in Bogotá.

3 Methodology

This section summarizes the fair algorithm for learning in allocation problems proposed in [1]. The algorithm maximizes the expected number of crimes discovered in the city by taking into account exclusively fair police allocations. In detail, the authors define an allocation as fair if the probability of discovering a crime for any two regions does not differ by more than a parameter α (equation (1)). In other words, an allocation is fair if a crime is observed in each region with (almost) equal probability. This notion is similar to the well studied notion of equal opportunity in the machine learning literature [10]. To formalize this fairness notion, let

$$f_i(v_i) = \mathbb{E}_{c_i \sim C_i}\left[\frac{\min(v_i, c_i)}{c_i}\right]$$

be the probability of discovering a crime in region i when v_i police officers are allocated to this region, where C_i is the true distribution of crimes in the region. Implicitly, we are assuming that $\min(v_i, c_i)$ crimes are actually observed by the police. Hence, the discovery model assumes that each police officer is able to observe one, and only one, crime, and allows for censored data in the sense that not all crimes that occur are actually discovered by the police. With this notation, the fairness requirement is expressed as:

$$\left|f_i(v_i) - f_j(v_j)\right| \leq \alpha, \quad \forall\, i, j \in \mathcal{G}, \tag{1}$$

To compute this expected value, the authors assume that crime occurrences follow a Poisson distribution C_i for each geographical region i.

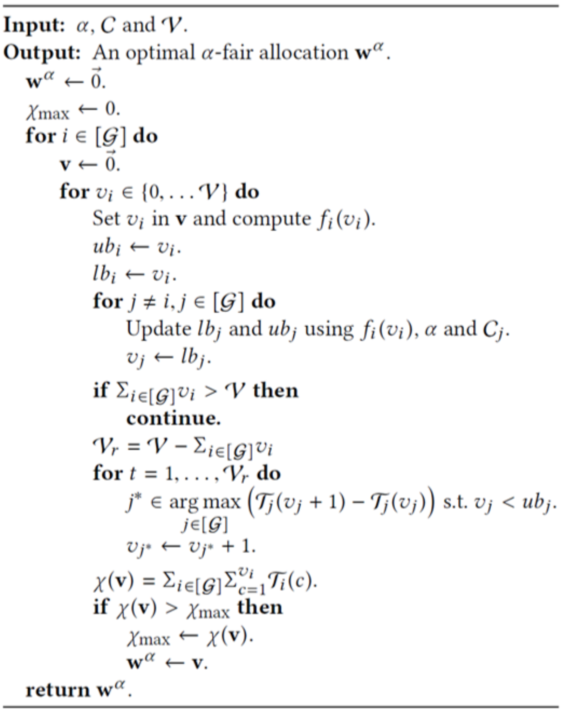
The proposed algorithm uses the fact that once a region i is assigned v_i units, this assignment uniquely defines bounds on the admissible allocations of the other regions: $f_j \in [f_i - \alpha/2, f_i + \alpha/2]$, $\forall\ j$. It then assigns to each region the minimum number of police officers such that $f_j \geq f_i - \alpha/2$ and check that this allocation does not exceed the total number of units available. If so, the remaining units are assigned according to a greedy rule that assigns an additional patrol to the region in $\mathcal{G}$ with the highest marginal probability of having an additional crime:

$$\operatorname*{argmax}_{i \in [\mathcal{G}]} \left(\mathcal{T}_i(v_i^t + 1) - \mathcal{T}_i(v_i^t)\right),$$

where $\mathcal{T}_i(v_i^t) = \mathbb{P}_{c_i \sim C_i}[c_i > v_i^t]$ is the right tail of the Poisson distribution for region i. These tails are also used to compute the expected number of crimes discovered for an allocation $v = (v_1, \ldots, v_g)$ among the regions $i \in \mathcal{G}$:

$$\chi(v) = \sum_{i \in \mathcal{G}} \sum_{c=1}^{v_i} \mathcal{T}_i(c). \tag{2}$$

The detailed algorithm, taken from [1], is depicted in Figure 3.



```
Input: $\alpha$, $C$ and $\mathcal{V}$.
Output: An optimal $\alpha$-fair allocation $\mathbf{w}^\alpha$.
  $\mathbf{w}^\alpha \leftarrow \vec{0}$.
  $\chi_{\max} \leftarrow 0$.
  for $i \in [\mathcal{G}]$ do
      $\mathbf{v} \leftarrow \vec{0}$.
      for $v_i \in \{0, \ldots \mathcal{V}\}$ do
          Set $v_i$ in $\mathbf{v}$ and compute $f_i(v_i)$.
          $ub_i \leftarrow v_i$.
          $lb_i \leftarrow v_i$.
          for $j \neq i, j \in [\mathcal{G}]$ do
              Update $lb_j$ and $ub_j$ using $f_i(v_i)$, $\alpha$ and $C_j$.
              $v_j \leftarrow lb_j$.
          if $\Sigma_{i \in [\mathcal{G}]} v_i > \mathcal{V}$ then
              continue.
          $\mathcal{V}_r = \mathcal{V} - \Sigma_{i \in [\mathcal{G}]} v_i$
          for $t = 1, \ldots, \mathcal{V}_r$ do
              $j^* \in \arg\max_{j \in [\mathcal{G}]} \left(\mathcal{T}_j(v_j + 1) - \mathcal{T}_j(v_j)\right)$ s.t. $v_j < ub_j$.
              $v_{j^*} \leftarrow v_{j^*} + 1$.
          $\chi(\mathbf{v}) = \Sigma_{i \in [\mathcal{G}]} \Sigma_{c=1}^{v_i} \mathcal{T}_i(c)$.
          if $\chi(\mathbf{v}) > \chi_{\max}$ then
              $\chi_{\max} \leftarrow \chi(\mathbf{v})$.
              $\mathbf{w}^\alpha \leftarrow \mathbf{v}$.
  return $\mathbf{w}^\alpha$.
```

Figure 3: Algorithm to compute an optimal fair allocation. From [1].

3.1 Extension of the algorithm

We first relax the assumption that the distribution of crime follows a Poisson distribution [1], as the empirical distributions of the number of crimes in Bogotá per locality do not follow such a distribution (Figure 2). Consequently, we use a Kernel Density Estimation to estimate in a flexible way the weekly distribution of crime in each locality of Bogotá. This distribution is then used to calculate the expected number of crimes discovered in each locality $\chi(v)$ given a police assignment v using equation 2. Note that the right tail of the crime distribution $\mathcal{T}_i(v_i) = \mathbb{P}_{c_i \sim \hat{C}_i}[c_i > v_i]$ is now computed with respect to the learnt distributions $\hat{C}_i$.

Furthermore, we modified the greedy rule to allocate the available patrols. Instead of using the rule proposed by [1]

$$\underset{i \in [\mathcal{G}]}{\operatorname{argmax}} \left(\mathcal{T}_i(v_i^t + 1) - \mathcal{T}_i(v_i^t) \right)$$

in which an additional police unit is assigned to the region in $\mathcal{G}$ with the highest probability mass in its crime distribution between v^t and $v^t + 1$, we assign patrols with the rule:

$$\underset{i \in [\mathcal{G}]}{\operatorname{argmax}} \left(\mathcal{T}_i(v_i^t + 1) \right)$$

such that the police allocation is increased by one unit in the region with the highest probability of crime occurrence greater than v^t. This change allows police to be assigned to regions with a low probability of observing exactly $v^t + 1$ crimes but with a high probability of observing a number of crimes greater than $v^t + 1$.

4 Results: Police allocation for Bogotá

We fit a Kernel Density Estimation (KDE) model for each locality to estimate the weekly crime distribution. The bandwidth for each of the kernels in the different localities was chosen using 10-fold cross validation. We then run the proposed modified algorithm to allocate 1051 police patrols among the 19 localities in Bogotá for different fairness thresholds α.

We compare our results with (i) an approximation to the current allocation used in Bogotá based on the number of quadrants in each locality, (ii) a naive allocation in which patrols are assigned according to the proportion of crime in each region, and (iii) a greedy allocation without taking into account any fairness requirement. We further study each of the changes introduced to the algorithm and compare the solution obtained to (iv) the original [1] algorithm assuming Poisson distributions (but computing the expected number of crimes using the distribution estimated with KDE), and (v) with the greedy allocation rule used by [1].

The allocations obtained by the different allocation strategies outlined above are shown in Figure 4. The current allocation based on the number of quandrants in each locality differs the most from all the allocations found. The naive allocation and the proposed modified algorithm have similar solutions as both take into account the crime distribution throughout the city. The proposed algorithm generates an allocation with a higher expected number of observed crimes an $\alpha \approx 0.02$, that is with a maximum difference in the probabilities of discovering crime between two localities of approximately 2%. Although the solution of the different methodologies does not differ much and most localities end up having a similar number of patrols assigned, slight changes in the solutions of each of the models represent either a fairer model or a higher expected number of crimes being discovered. The Appendix provides the solutions obtained in Table 1.

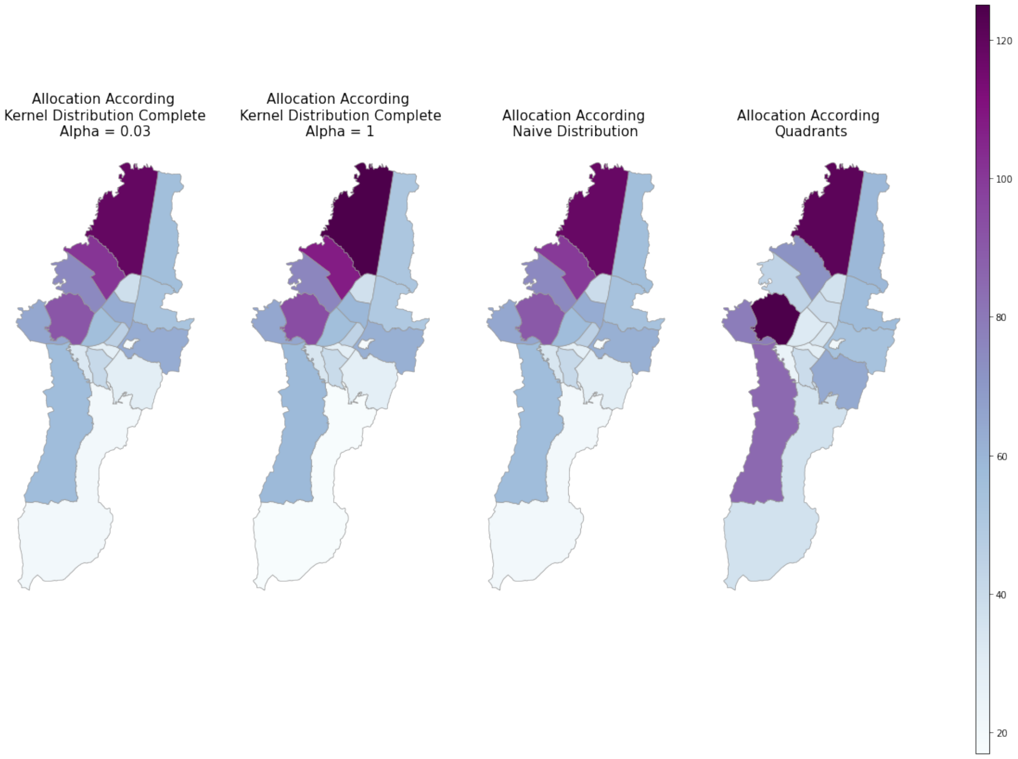


Figure 4: Allocation of police patrols using different methodologies.

To evaluate the different allocation configurations we use the expected number of crime discovered associated to the resulting allocation:

$$\chi(v) = \sum_{i \in \mathcal{G}} \sum_{c_i}^{v_i} \mathcal{T}_i(c),$$

where $\mathcal{T}_i$ is the right tail of the distribution estimated through KDE.

The results for different values of the fairness parameter α are shown in Figure 5. Both models using the kernel density functions (with different greedy allocation rules) exhibit a similar behaviour and seem to stabilize for $\alpha < 0.03$. The Poisson model has a different behavior as the expected number of crimes discovered decreases as the α value increases. We believe that this behavior is a consequence of the greedy allocation rule used in [1]. It should be noted that the Naive model in which patrols are assigned according to the proportion of crime in each region have a low α value (0.022) representing a fair allocation, but has a lower expected number of crimes discovered than the proposed model.

On the other hand, the kernel density model that takes into account the right tail of the estimated distribution to allocate the remaining police units seems to have a better performance than any of the other models for all the α values. This model allocates the patrols fairly while maximizing the average number of crimes discovered overall. The current allocation based on the number of quadrants in each locality has an $\alpha = 0.54$ and an average crime discovered equal to 983.59, representing an unfair allocation with a low expected number of crimes discovered (not shown in figure 5). This is the most deficient allocation out of all the models tested.

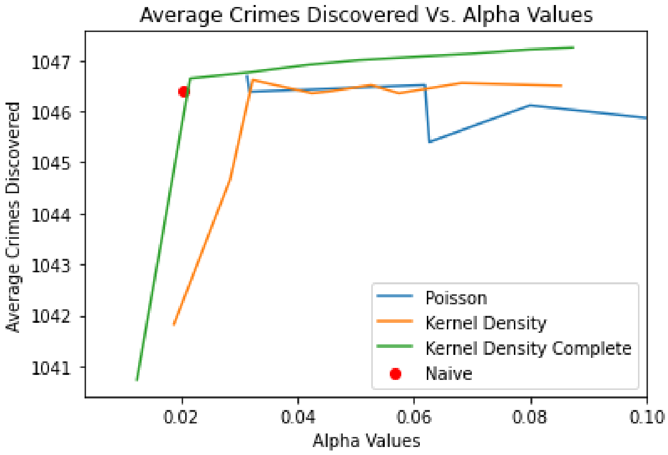


Figure 5: Allocation for the different methodologies over different fairness thresholds α.

5 Conclusions

Policy makers armed with new machine learning tools are making progress in crime prediction. Nevertheless, algorithms may be biased and reproduce existing police patrolling biases that may have disparate impacts for different populations. This paper shows in a real setting how to take a simple predictive policing algorithm (e.g., a KDE model) and use it for fairly allocating scare police resources in such a way that we still have a good predictive model and mitigate the unfair allocation of police resources as described in the paper. This paper addresses a central question in predictive police patrolling for any city independent of their police deployment strategy.

We extend the algorithm proposed by [1] in two main ways. We first relax the assumption that the distribution of crime follows a Poisson distribution, as the empirical distributions of the number of crimes in Bogotá per locality do not follow such a distribution (Figure 2). Consequently, we use a Kernel Density Estimation (KDE) model to flexibly learn the distribution of crime occurrence in each region and use it in the allocation algorithm of [1]. Second, we modify the greedy assignment rule such that the algorithm assigns an additional patrol to the region with the highest probability of exhibiting more crimes than its current allocation, rather than the region with the highest marginal probability of having an additional crime.

We run the modified algorithm for different fairness thresholds α and compare our results with an approximation to the current allocation strategy used in Bogotá, a naive allocation in which patrols are assigned according to the proportion of crime in each region, and with a greedy allocation without taking into account any fairness requirement. The results clearly show that the current allocating police strategy in Bogotá is dominated by our allocating mechanism even when we use a simple machine learning model to make crime predictions (KDE): the model discovers mores crimes and mitigates unfair allocations.

However, at least two issues are left for future research. First, although the modified algorithm [1] works well with real data for Bogotá, which clearly suggests policy makers should take it seriously, we do not provide general performance guarantees as [1] does in the particular setting in which crime follows a Poisson distribution. This is clearly unrealistic in our case as shown in 2 and given all the evidence from more sophisticated machine learning algorithms. Second, in this model censored data plays a passive role since, by construction, we do not allow discovered crimes to bias our algorithm (there is no training as police is deployed). This means we do not allow for feedback loop and take our model as ground truth. Clearly this is not the case in a real setting where one has to discover the true model and at the same time deploy police that discovers crimes. We leave this learning problem of sophisticated machine learning models for future research.

References

[1] H. Elzayn, S. Jabbari, C. Jung, M. Kearns, S. Neel, A. Roth, and Z. Schutzman, "Fair algorithms for learning in allocation problems," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019, pp. 170–179.

[2] G. O. Mohler, M. B. Short, S. Malinowski, M. Johnson, G. E. Tita, A. L. Bertozzi, and P. J. Brantingham, "Randomized controlled field trials of predictive policing," *Journal of the American statistical association*, vol. 110, no. 512, pp. 1399–1411, 2015.

[3] S. Greengard, "Policing the future," *Communications of the ACM*, no. 3, pp. 19–21, 2012.

[4] B. mayor's office. (2019) In bogotá, crimes will be predicted with artificial intelligence. [Online]. Available: https://bogota.gov.co/mi-ciudad/seguridad/modelo-de-prediccion-de-crimenes-en-bogota

[5] Semana. (2020) Bogota: dynamic quadrants, among the new measures to reduce insecurity. [Online]. Available: https://www.semana.com/nacion/articulo/bogota-cuadrantes-moviles-entre-las-nuevas-medidas-para-frenar-la-inseguridad/682206/

[6] K. Lum and W. Isaac, "To predict and serve?" *Significance*, vol. 13, no. 5, pp. 14–19, 2016.

[7] D. Ensign, S. A. Friedler, S. Neville, C. Scheidegger, and S. Venkatasubramanian, "Runaway feedback loops in predictive policing," *arXiv preprint arXiv:1706.09847*, 2017.

[8] G. Mohler, R. Raje, J. Carter, M. Valasik, and J. Brantingham, "A penalized likelihood method for balancing accuracy and fairness in predictive policing," in *2018 IEEE international conference on systems, man, and cybernetics (SMC)*. IEEE, 2018, pp. 2454–2459.

[9] C. Urcuqui, Christian, S. Moreno, Juan, A. Montenegro, Carlos, J. Riascos, Alvaro, and M. Dulce, "Accuracy and fairness in a conditional generative adversarial model of crime prediction," *7th International Conference on Behavioural and Social Computing (BESC 2020), Bournemouth, UK.*, 2020.

[10] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," *In Advances in neural information processing systems*, pp. 3315–3323, 2016.

[11] C. N. Police. (2010) Institutional strategy for citizen security: National plan for community policing by quadrants. [Online]. Available: https://www.oas.org/es/sap/dgpe/innovacion/banco/ANEXO%20I.%20PNVCC.pdf

Appendix

Locality	Poisson	Kernel Density	Kernel Density Complete	Naive	Quadrants
Antonio Nariño	29	28	28	28	29
Barrios Unidos	39	38	38	39	36
Bosa	65	64	65	65	79
Candelaria	18	18	18	18	21
Chapinero	54	53	53	54	57
Ciudad Bolívar	57	56	57	57	85
Engativá	100	101	101	100	72
Fontibón	75	75	75	75	44
Kennedy	91	91	91	90	125
Los Mártires	45	44	45	45	34
Puente Aranda	57	56	56	57	32
Rafael Uribe Uribe	41	40	41	41	39
San Cristobal	29	29	29	29	64
Santa Fé	62	63	63	62	54
Suba	117	119	119	118	121
Teusaquillo	63	64	63	63	39
Tunjuelito	33	33	33	34	25
Usaquén	56	57	56	56	59
Usme	20	20	20	20	36

Table 1: Number of patrols allocated to each locality given the different methodologies using $\alpha = 0.03$.

MODELING A 3G MOBILE PHONE BASE RADIO USING ARTIFICIAL INTELLIGENCE TECHNIQUES

Eduardo Calo[1, 5], Gabriel Vaca[1], Cristina Sánchez[2],
David Jines[3], Giovanny Amancha[3], Ángel Flores[3],
Alex Santana G[3] and Fernanda Oñate[4]

[1]Carrera Electricidad, Instituto Superior María Natalia Vaca, Ambato, Ecuador
[2]Carrera Mecánica Automotriz,
Instituto Superior María Natalia Vaca, Ambato, Ecuador
[3]Carrera Electrónica, Instituto Superior María Natalia Vaca, Ambato, Ecuador
[4]Instituto Superior Tecnológico Pelileo-Baños, Baños, Ecuador
[5]UNIR, Logroño, La Rioja, España

ABSTRACT

The main objective of this work is to be able to use artificial intelligence techniques to be able to design a predictive model of the performance of a third-generation mobile phone base radio, using the analysis of KPIs obtained in a statistical data set of the daily behaviour of an RBS. For the realization of these models, various techniques such as Decision Trees, Neural Networks and Random Forest were used. which will allow faster progress in the deep analysis of large amounts of data statistics and get better results. In this part of the work, data was obtained from the behaviour of a third-party mobile phone base radio generation of the Claro operator in Ecuador, it should be noted that. The data are KPIs of the daily and hourly performance of a radio base of third generation mobile telephony, these data were obtained through the operator's remote monitoring and management tool Sure call PRS. To specify this practical case, several models were generated based on in various artificial intelligence technique for the prediction of performance results of a mobile phone base radio of third generation, the same ones that after several tests were creation of a predictive model that determines the performance of a mobile phone base radio. As a conclusion of this work, it was determined that the development of a predictive model based on artificial intelligence techniques is very useful for the analysis of large amounts of data in order to find or predict complex results, more quickly and trustworthy.

KEYWORDS

Neural Networks, Performance, Radio Base, Random Forest, Throughput.

1. INTRODUCTION

Deep learning is one of the techniques more evolution has had within the field of Intelligence Artificial, thanks to this advance a development has begun to very notorious in various sectors such as financial, vehicle, telecommunications, understanding of natural language, translators of languages and many more sectors worldwide, streamlining processes optimization and having a future vision of them, optimizing resources.

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 389-407, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112330

The purpose of this work is to use these intelligence techniques artificial in a practical case of real life, such as the analysis of the performance of an RBS and predict its behaviour or results to be obtained after the analysis of its operating parameters, the same ones that in the telecommunications area will help us in the optimization of resources and in the improvement of a telephone network third generation, the analysis of this performance was carried out at the level of data traffic in telephone stations, predicting its behaviour with the change of different variables.

To specify this practical case, several models were generated based on in various artificial intelligence techniques such as Random Forest, Decision Trees and Neural Networks, for the prediction of performance results of a mobile phone base radio of third generation, the same ones that after several tests were creation of a predictive model that determines the performance of a mobile phone base radio.

1.1. In the investigation "Intelligent Techniques in the assignment of dynamic spectrum for cognitive wireless networks":

In this work, a number of techniques of artificial intelligence that help provide solutions in the management of electromagnetic spectrum in wireless networks, taking into account that mobile telephony is within wireless networks due to the use and management of frequencies found in the spectrum, these investigated techniques allow to analyse parameters such as spectrum availability, energy consumption, channel division, necessary requirements for the user, taking into account that all These parameters have as a final result the existence of a transmission and reception wireless communication system optimal data. [1].

1.2. In the research "Cellular mobile telephony: origin, evolution, perspectives"

Worldwide, the evolution in mobile telephony has had great events, such is the case that the use of mobile devices has potentially increased use of mobile phones initially relied on him sending voice signals and text messages in what corresponds to the first generation of mobile telephony, from there onwards there have been various types of evolutions that range from the first generation to the fifth generation, from the third generation the development is very significant that the itself allows the use of mobile phones not only for transmission determined which is the model that would give us the best results when working With this type of data, the variables with which we worked, have relation to data traffic measurements within radio bases, both for data upload and download traffic.

2. STATE OF THE ART

As part of the techniques proposed for the development of this the work evaluates the use of Artificial Intelligence as a means of function of receiving the signals in an RBS both for frequencies of rise and fall in the two types of technologies. [2]

2.1. In the research "Third generation cellular networks CDMA2000 and WCDMA":

After the emergence of second-generation technology and due to the increase in users and therefore saturation of the network and need to implement new services in what has to do with mobile telephony, third generation cellular networks appear with its CDMA 2000 and WCDMA models, the first of these models is the link to second generation and was in charge of the link of compatibility and migration between second and third generation, while WCDMA focuses on increasing the number of users, as the picture shows [3]

In CDMA2000 there are two modulation alternatives for the downlink and these are: multicarrier modulation (MC) and direct spread (DS) that use 3 carriers of 1.25 MHz or 5 MHz; while in WCDMA it uses a single 5 MHz carrier.[4]

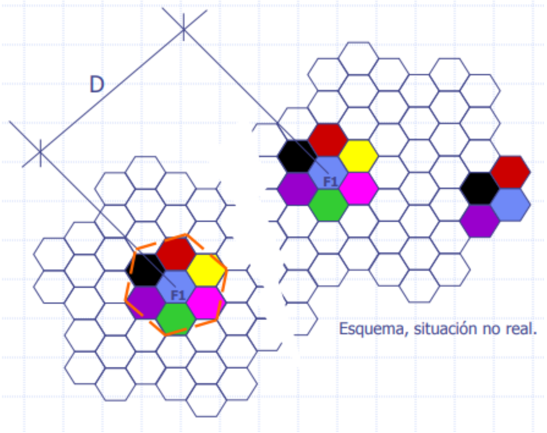


Figure 1. Third generation cellular networks CDMA2000 and WCDMA [3]

2.2. In the research “Intelligent Systems applied to Data"

The emergence of artificial intelligence and the use of all its techniques and tools have resulted in the emergence of intelligent systems the same ones that are capable of learning and execute actions automatically based solely on examples of some type of functioning adapting to learning very easily, as the picture shows, neural networks have been used in various investigations for data congestion analysis in various types of networks such as ATMs, as well as in the part IT security, obtaining good results. [5]

The discipline in computer science given by the mathematician Alan Turing began the birth of Artificial Intelligence that aims to question whether machines have the ability to think and is currently known as the Turing test. To obtain a good design of the topology of a network with restrictions using the tools of artificial intelligence, it is necessary to build an algorithm that analyses each and every one of the possible solutions. [6]

Figure 2. Intelligent Systems applied to Data [4]

2.3. In the research "Neural networks and traffic prediction":

The use of artificial intelligence tools allows us to predict data traffic, as the picture shows, classify the types of traffic and recognize the existing variables in the analysis of information traffic of a network, These AI methods are additional alternatives that allow us to an easy way to solve this problem, in this case it applies a neural network that based on the weights of neurons and after executing several iterations gives us predictions with data according to expectations. [7]

The different ways that exist to predict traffic consist of extracting the underlying relationships of the previous values and they are also used to extrapolate and predict future behavior. [8]

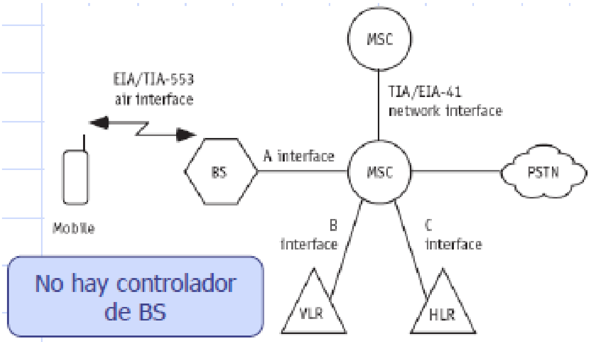


Figure 3. Neural networks and traffic prediction [5]

2.4. In the investigation “Prediction of the demand for fixed telephony through artificial neural networks"

The high increase in users both for fixed telephony and for mobile telephony has made the statistics in bases regarding the demand for the acquisition of these services increases, as the picture shows, the operators that provide this type of service are in need of use tools that provide estimates of next values growth of users and demand for the service. [9]

The adoption of a new technology generally follows a growth pattern of a logistics curve in which low growth with few users is initially identified, followed by high accelerated growth in moderate time intervals. [10]

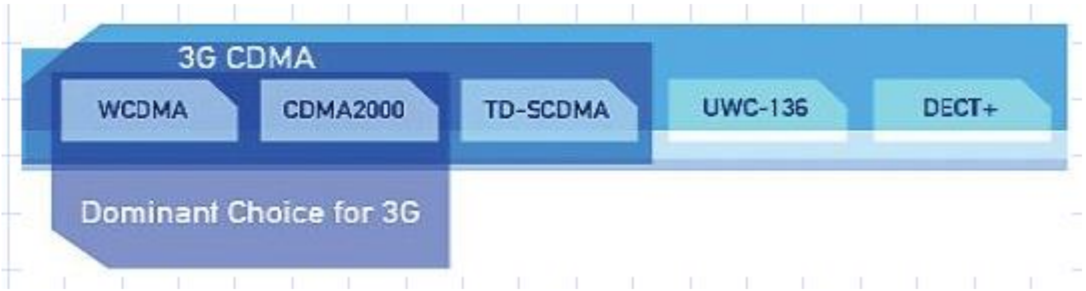


Figure 4. Prediction of the demand for fixed telephony through artificial neural networks [6]

3. WORK METHODOLOGY

Because the proposed objective of the project is the generation of various predictive models of the performance of a base radio third-generation mobile telephony, exactly targeted to traffic of data, based on machine learning algorithms within artificial intelligence, and after an analysis of the techniques existing within this branch and the type of data that the PRS tool (RBS management and monitoring). [11]

a) For the development and implementation of said models of Predicting the performance of a mobile phone base radio is used the Python Programming Language, due to its variability and the large number of intelligence tools and libraries.
b) Use of the free platform Google Colaboratory for development of the models, this is a cloud service, which provides us with a Jupyter Notebook that we can access with a web browser without import if we use Windows, Linux or Mac, has as great advantages since being an online tool suitable for this type of jobs offers high-performance technical features such as adequate RAM, possibility of activating a GPU.
c) A general analysis was performed to determine the properties statistics of the data and especially of the variables to be predicted, when use such information what we determine is the type of data with those that we are going to work on, in addition, it was determined that for the variables to predict there is a very large difference between their values because when compare the maximum, minimum and the mean you can notice that they exist values that can be very small with respect to others of the same variable.
d) Because the range of the values of the variables to be predicted is extremely broad, it was necessary to include a stage of data normalization in order to avoid a bias in the implementation of the model, after testing and analysing the results, the normalization method known as MinMaxScaler, the same one that generated an optimal normalization of the data with which we should work.[12]
e) Three models were implemented for each variable and one model final where an estimate of the three variables was attempted at the once with a single model. For this, an order was followed, being the first to always make the selection of the variables to use according to their correlation.
f) After analysing the results of this last model, they determined that the first variable has no relationship with the other two so when using a single model to estimate all three, the other Variables impair performance in estimating this.

4. DEVELOPMENT

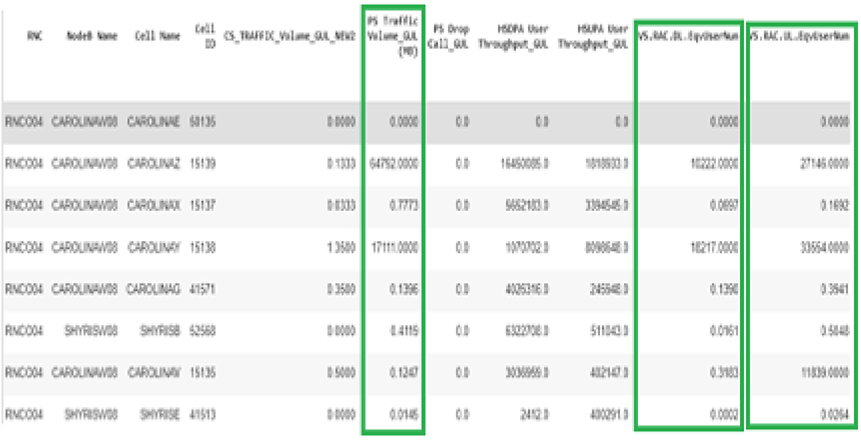
4.1. Obtaining Data

In this part of the work, data was obtained from the behaviour of a third-party mobile phone base radio generation of the Claro operator in Ecuador, it should be noted that.

The data are KPIs of the daily and hourly performance of a radio base of third generation mobile telephony, these data were obtained through the operator's remote monitoring and management tool Sure call PRS. [13]

4.2. Selection Of Variables To Predict

The predictor variables that were selected for the elaboration from this job they were: VS.RAC.DL.EqvUserNum, VS.RAC.UL.EqvUserNum, PS TraficVolume_GUL, this set of variables help us to have a general and statistical idea of the behaviour of mobile phone base radios with respect to the data traffic existing in it.

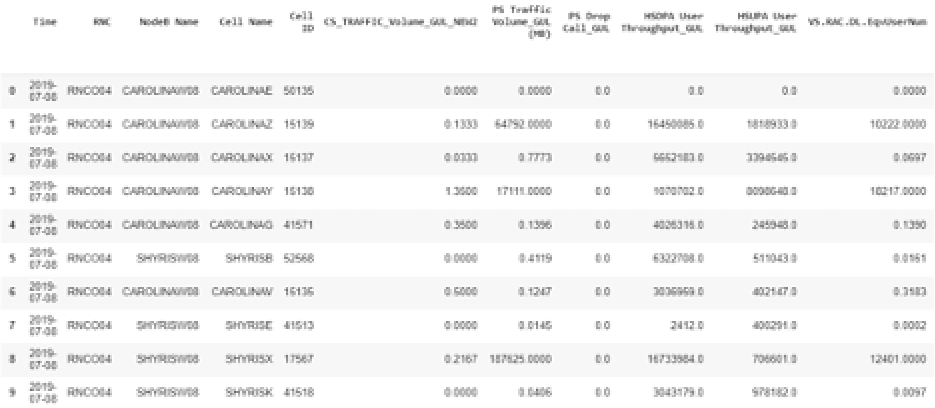


RNC	NodeB Name	Cell Name	Cell ID	CS_TRAFFIC_Volume_GUL_NEW2	PS Traffic Volume_GUL (MB)	PS Drop Call_GUL	HSDPA User Throughput_GUL	HSUPA User Throughput_GUL	VS.RAC.DL.EqvUserNum	VS.RAC.UL.EqvUserNum
RNCO04	CAROLINAW08	CAROLINAE	50135	0.0000	0.0000	0.0	0.0	0.0	0.0000	0.0000
RNCO04	CAROLINAW08	CAROLINAZ	15139	0.1333	64752.0000	0.0	16450085.0	1818933.0	10222.0000	27146.0000
RNCO04	CAROLINAW08	CAROLINAX	15137	0.0333	0.7773	0.0	5652183.0	3394545.0	0.0697	0.1692
RNCO04	CAROLINAW08	CAROLINAY	15138	1.3500	17111.0000	0.0	1070702.0	8098648.0	10217.0000	33554.0000
RNCO04	CAROLINAW08	CAROLINAG	41571	0.3500	0.1396	0.0	4026316.0	245948.0	0.1390	0.3941
RNCO04	SHYRISW08	SHYRISB	52568	0.0000	0.4119	0.0	6322708.0	511043.0	0.0161	0.5848
RNCO04	CAROLINAW08	CAROLINAW	15135	0.5000	0.1247	0.0	3036959.0	402147.0	0.3183	11839.0000
RNCO04	SHYRISW08	SHYRISE	41513	0.0000	0.0145	0.0	2412.0	400291.0	0.0002	0.0264

Figure 5. Graph of the selection of variables to predict (Google Colaboratory)

4.3. Statistic Analysis

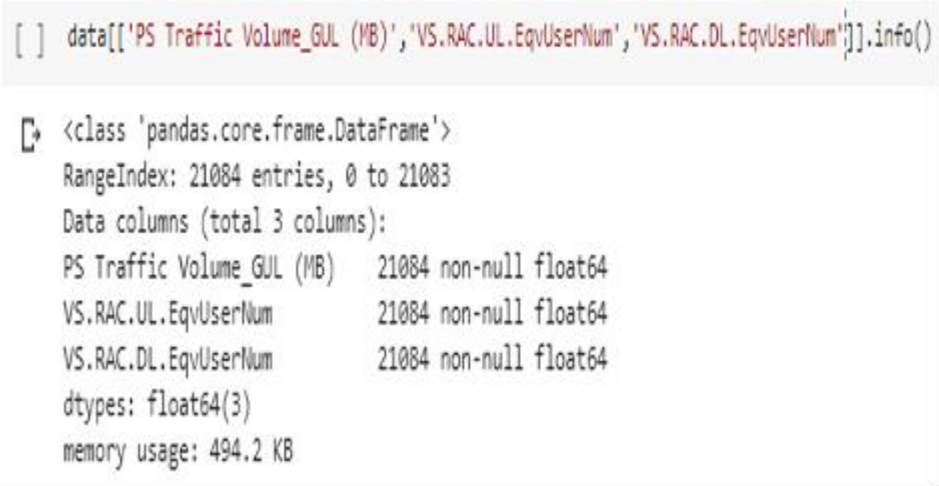
First, a general analysis was carried out to determine the statistical properties of the data and especially of the variables a predict.



	Time	RNC	NodeB Name	Cell Name	Cell ID	CS_TRAFFIC_Volume_GUL_NEW2	PS Traffic Volume_GUL (MB)	PS Drop Call_GUL	HSDPA User Throughput_GUL	HSUPA User Throughput_GUL	VS.RAC.DL.EqvUserNum
0	2019-07-08	RNCO04	CAROLINAW08	CAROLINAE	50135	0.0000	0.0000	0.0	0.0	0.0	0.0000
1	2019-07-08	RNCO04	CAROLINAW08	CAROLINAZ	15139	0.1333	64752.0000	0.0	16450085.0	1818933.0	10222.0000
2	2019-07-08	RNCO04	CAROLINAW08	CAROLINAX	15137	0.0333	0.7773	0.0	5652183.0	3394545.0	0.0697
3	2019-07-08	RNCO04	CAROLINAW08	CAROLINAY	15138	1.3500	17111.0000	0.0	1070702.0	8098648.0	10217.0000
4	2019-07-08	RNCO04	CAROLINAW08	CAROLINAG	41571	0.3500	0.1396	0.0	4026316.0	245948.0	0.1390
5	2019-07-08	RNCO04	SHYRISW08	SHYRISB	52568	0.0000	0.4119	0.0	6322708.0	511043.0	0.0161
6	2019-07-08	RNCO04	CAROLINAW08	CAROLINAW	15135	0.5000	0.1247	0.0	3036959.0	402147.0	0.3183
7	2019-07-08	RNCO04	SHYRISW08	SHYRISE	41513	0.0000	0.0145	0.0	2412.0	400291.0	0.0002
8	2019-07-08	RNCO04	SHYRISW08	SHYRISX	17567	0.2167	187625.0000	0.0	16733984.0	706601.0	12401.0000
9	2019-07-08	RNCO04	SHYRISW08	SHYRISK	41518	0.0000	0.0406	0.0	3043179.0	978182.0	0.0097

Figure 6. Graph of the data set variables (Google Colaboratory)

Using this information, it was determined that the type of data with those that were worked on, in this case the three variables to be predicted were of the type floating point and 64-bit meaning that they contained a decimal format.

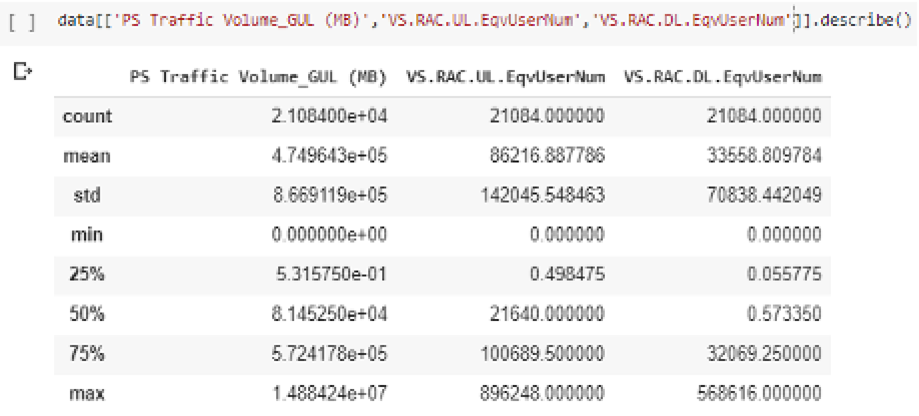


```
[ ] data[['PS Traffic Volume_GUL (MB)','VS.RAC.UL.EqvUserNum','VS.RAC.DL.EqvUserNum']].info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21084 entries, 0 to 21083
Data columns (total 3 columns):
PS Traffic Volume_GUL (MB)    21084 non-null float64
VS.RAC.UL.EqvUserNum          21084 non-null float64
VS.RAC.DL.EqvUserNum          21084 non-null float64
dtypes: float64(3)
memory usage: 494.2 KB
```

Figure 7. Graph of the data type of the variables to be predicted (Google Colaboratory)

Furthermore, what was found is that for the variables to be predicted there were a very big difference between their values because when comparing the maximum, minimum and mean it can be noted that there are values that can be very small with respect to others of the same variable. [14]



```
[ ] data[['PS Traffic Volume_GUL (MB)','VS.RAC.UL.EqvUserNum','VS.RAC.DL.EqvUserNum']].describe()
```

```
       PS Traffic Volume_GUL (MB)  VS.RAC.UL.EqvUserNum  VS.RAC.DL.EqvUserNum
count                2.108400e+04          21084.000000          21084.000000
mean                 4.749643e+05          86216.887786          33558.809784
std                  8.669119e+05         142045.548463          70838.442049
min                  0.000000e+00              0.000000              0.000000
25%                  5.315750e-01              0.498475              0.055775
50%                  8.145250e+04          21640.000000              0.573350
75%                  5.724178e+05         100689.500000          32069.250000
max                  1.488424e+07         896248.000000         568616.000000
```

Figure 8. Graph of the existing values in the variables (Google Colaboratory)

Moreover, simple graphics prints were also made of the data to visually analyse how the data are distributed and observe if there could be outliers or also known data as outliers.

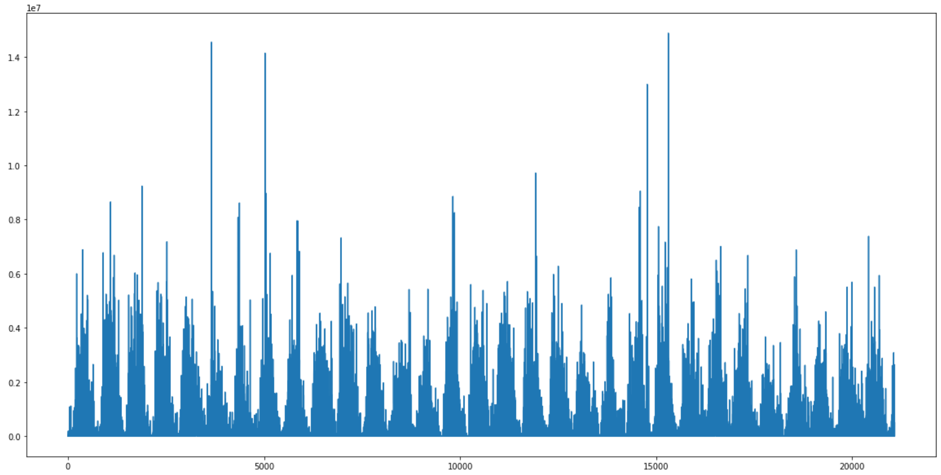


Figure 9. Graph of the data distribution of the variable PS Trafic (Google Colaboratory)

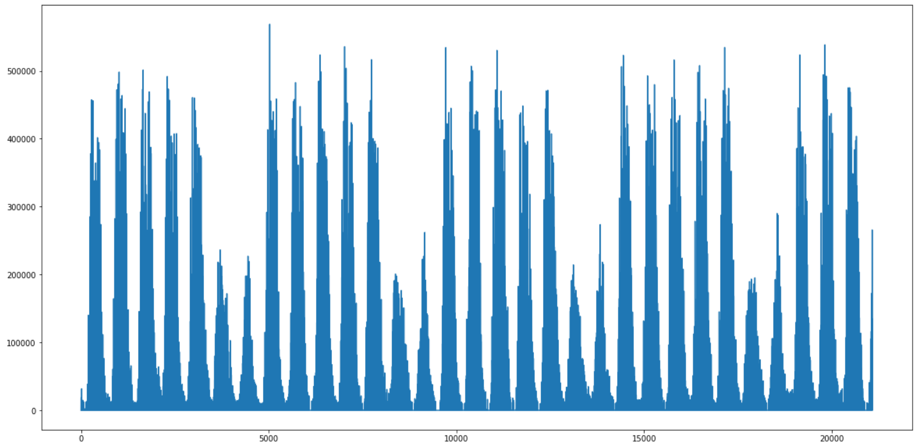


Figure 10. Graph of the data distribution of the variable VS.RAC.DL.EqvUserNum (Google Colaboratory)

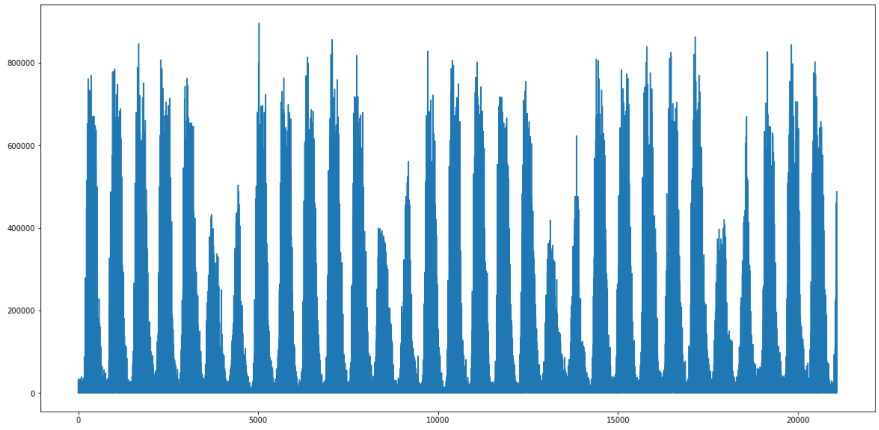


Figure 11. Graph of the data distribution of the variable VS.RAC.UL.EqvUserNum (Google Colaboratory)

4.4. Correlation

Since in the database with which we worked, there was a large number of variables, around a hundred, that could be used to estimate the target variables, to determine the ones with the most information that can contribute, an analysis of the correlation between all the variables with special emphasis on target variables.

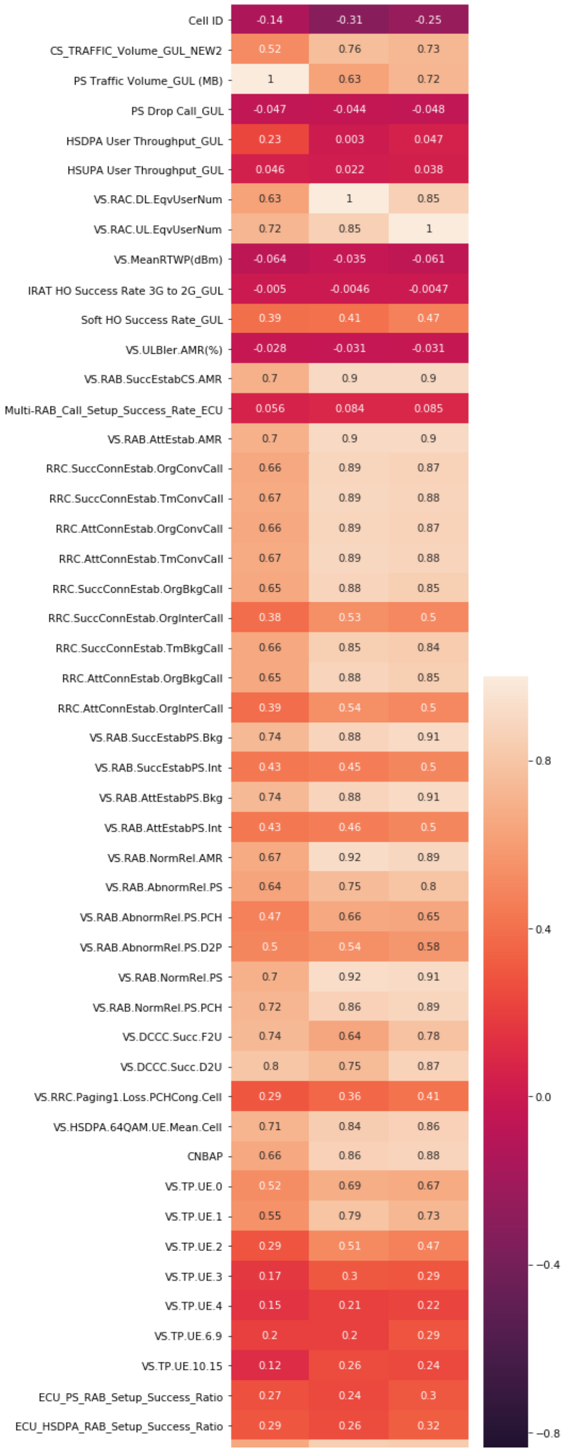
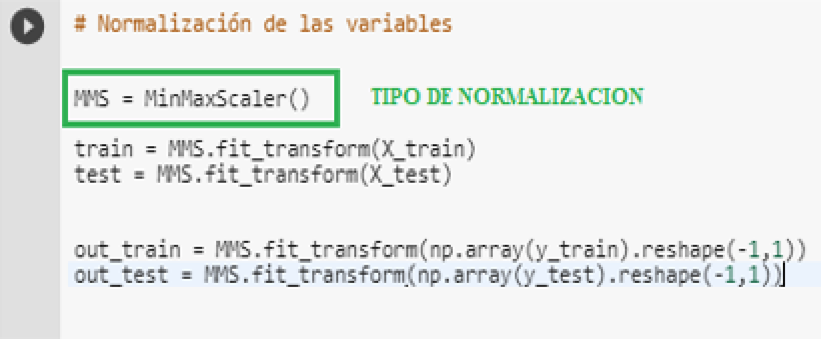


Figure 12. Variable correlation graph (Google Colaboratory)

4.5. Data Normalization

Because the range of the values of the variables to be predicted is extremely broad, it was necessary to include a stage of data normalization in order to avoid a bias in the implementation of the model, after testing and analysing the results, the normalization method known as **MINMAXSCALER,**



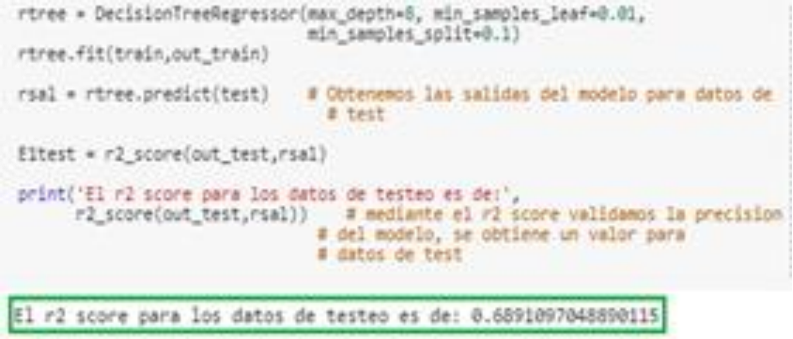
```
# Normalización de las variables

MMS = MinMaxScaler()

train = MMS.fit_transform(X_train)
test = MMS.fit_transform(X_test)

out_train = MMS.fit_transform(np.array(y_train).reshape(-1,1))
out_test = MMS.fit_transform(np.array(y_test).reshape(-1,1))
```

Figure 13. MinMaxScaler normalization graph (Google Colaboratory)

5. MODELS - VARIABLE PS TRAFIC

5.1. Decision Trees

Using decision trees, the hyperparameters by increasing and decreasing depth, parameters with respect to the sheets and others, in order to find the best ones. [13] As we can see in the graph the result that is displayed is the value of r2 score, this value is a metric that tells me the accuracy in prediction calculations of variables that as we can see for this model is a value of 0.6891097048890115



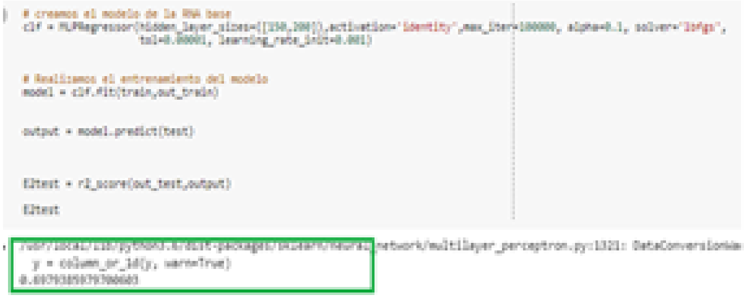
```
rtree = DecisionTreeRegressor(max_depth=8, min_samples_leaf=0.01,
                              min_samples_split=0.1)
rtree.fit(train,out_train)

rsal = rtree.predict(test)    # Obtenemos las salidas del modelo para datos de
                                # test

Eltest = r2_score(out_test,rsal)

print('El r2 score para los datos de testeo es de:',
      r2_score(out_test,rsal))    # mediante el r2 score validamos la precision
                                 # del modelo, se obtiene un valor para
                                 # datos de test
```

```
El r2 score para los datos de testeo es de: 0.6891097048890115
```

Figure 14. R2 score graph for the Decision Tree (Google Colaboratory)

5.2. Neural Network

In the same way, for the neural network model, various tests modifying the number of neurons in the hidden layer and the number of hidden layers, as well as their function of activation, its learning rate, its optimization algorithm, number of epochs and overtraining control, looking for optimize the value of your hyperparameters for better performance [15], the optimizer that was used was an LBFGS optimizer, the r2 Score was 0.6979385979700603



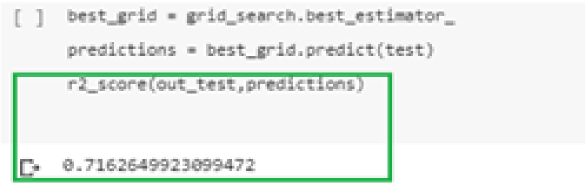
```
# creamos el modelo de la RNA base
clf = MLPRegressor(hidden_layer_sizes=([150,200]),activation='identity',max_iter=100000, alpha=0.1, solver='lbfgs',
                   tol=0.00001, learning_rate_init=0.001)

# Realizamos el entrenamiento del modelo
model = clf.fit(train,out_train)

output = model.predict(test)


R2test = r2_score(out_test,output)

R2test
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/neural_network/multilayer_perceptron.py:1321: DataConversionWa
  y = column_or_1d(y, warn=True)
0.6979385979706683
```

Figure 15. Graph of r2 score for the Neural Network (Google Colaboratory)

5.3. Random Forest

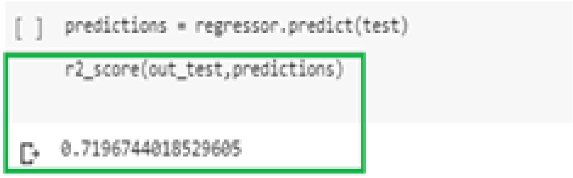
Finally, with the Random Forest technique what was done was a search using grid search to find your best hyperparameters, prior to this test were performed modifying the parameters manually and finding in which range is where better performance has the model similar to the process used for the other models, the r2 Score was 0.7162649923099472



```
[ ]  best_grid = grid_search.best_estimator_
     predictions = best_grid.predict(test)
     r2_score(out_test,predictions)

     0.7162649923099472
```

Figure 16. Graph of r2 score for Random Forest (Google Collaboratory)

Once this was determined, a finer adjustment was made. using grid search and using only parameters in ranges where better performance was found with an r2 Score 0.7196744018529605



```
[ ]  predictions = regressor.predict(test)
     r2_score(out_test,predictions)

     0.7196744018529605
```

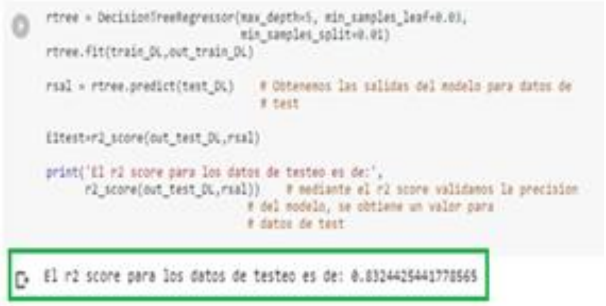
Figure 17. R2 score plot for Random Forest with fine adjustment (Google Colaboratory)

A similar procedure was used for the other two variables. During the training and testing of the models, the only thing that changed is that in these no atypical data were found as in the case of the data traffic variable. [16]

5.3.1. Variable VS.RAC.DL.EqvUserNum

a) Decision trees

As we can see in the graph the result that is displayed is the value of r2 score, this value is a metric that tells me the accuracy in prediction calculations of variables that as we can see for this model is a value of 0.8324425441778565



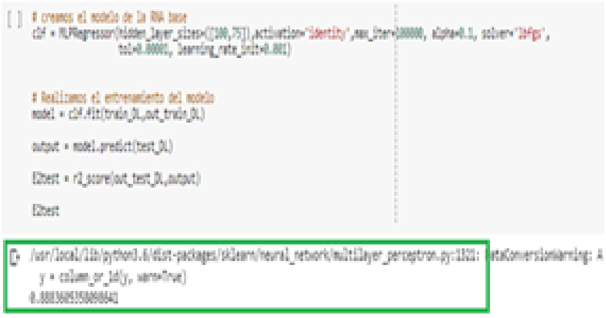
```
rtree = DecisionTreeRegressor(max_depth=5, min_samples_leaf=0.03,
                              min_samples_split=0.01)
rtree.fit(train_DL,out_train_DL)

rsal = rtree.predict(test_DL)    # Obtenemos las salidas del modelo para datos de
                                 # test

E1test=r2_score(out_test_DL,rsal)

print('El r2 score para los datos de testeo es de:',
      r2_score(out_test_DL,rsal))    # mediante el r2 score validamos la precision
                                   # del modelo, se obtiene un valor para
                                   # datos de test
```

```
El r2 score para los datos de testeo es de: 0.8324425441778565
```

Figure 18. R2 score graph for the Decision Tree (Google Collaboratory)

b) Neural Network

R2 Score of 0.8883605358098641



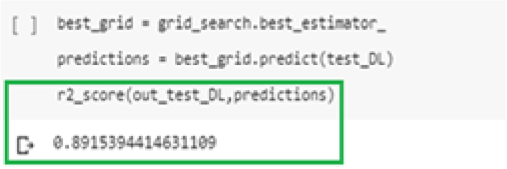
```
[ ] # creamos el modelo de la RNA base
    clf = MLPRegressor(hidden_layer_sizes=([100,75]),activation='identity',max_iter=100000, alpha=0.1, solver='lbfgs',
                       tol=0.00001, learning_rate_init=0.001)


    # Realizamos el entrenamiento del modelo
    model = clf.fit(train_DL,out_train_DL)

    output = model.predict(test_DL)

    E2test = r2_score(out_test_DL,output)

    E2test
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/neural_network/multilayer_perceptron.py:1321: DataConversionWarning: A
  y = column_or_1d(y, warn=True)
0.8883605358098641
```

Figure 19. R2 score graph for the Neural Network (Google Colaboratory)

c) Random Forest

R2 score was 0.8915394414631109

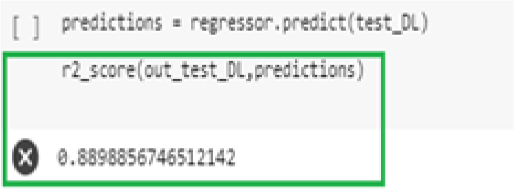


```
[ ] best_grid = grid_search.best_estimator_

    predictions = best_grid.predict(test_DL)

    r2_score(out_test_DL,predictions)
```

```
0.8915394414631109
```

Figure 20. R2 score plot for Random Forest (Google Colaboratory)

Once this was determined, a finer adjustment was made. using grid search and using only parameters with ranges regulated obtaining an r2 Score of 0.8898856746512142



```
[ ] predictions = regressor.predict(test_DL)

    r2_score(out_test_DL,predictions)
```

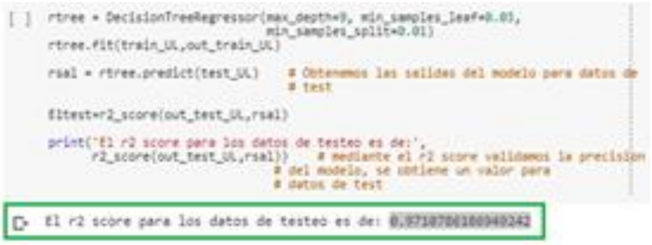
```
0.8898856746512142
```

Figure 21. Graph of r2 score for Random Forest with fine adjustment (Google Colaboratory)

5.3.2. Variable VS. RAC.UL.EqvUserNum'

a) Decision trees

As we can see in the graph the result that is displayed is the value of r2 score, this value is a metric that tells me the accuracy in prediction calculations of variables that as we can see for this model is a value of 0.9710786186949242. [20]



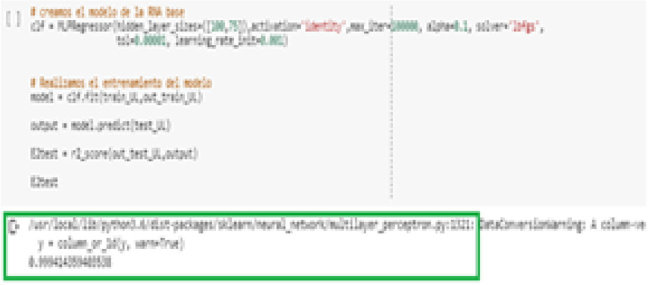
```
[ ]  rtree = DecisionTreeRegressor(max_depth=9, min_samples_leaf=0.01,
                                    min_samples_split=0.01)
     rtree.fit(train_UL,out_train_UL)

     rsal = rtree.predict(test_UL)     # Obtenemos las salidas del modelo para datos de
                                       # test

     E1test=r2_score(out_test_UL,rsal)

     print('El r2 score para los datos de testeo es de:',
           r2_score(out_test_UL,rsal))    # mediante el r2 score validamos la precision
                                        # del modelo, se obtiene un valor para
                                        # datos de test
```

```
El r2 score para los datos de testeo es de: 0.9710786186949242
```

Figure 22. R2 score graph for the Decision Tree (Google Colaboratory)

b) Neural Network

Obtaining an R2 Score of 0.999414359403538



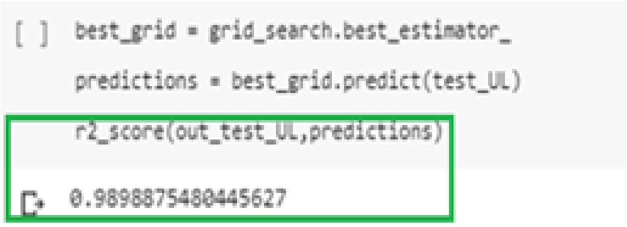
```
[ ]  # creamos el modelo de la RNA base
     clf = MLPRegressor(hidden_layer_sizes=([100,75]),activation='identity',max_iter=10000, alpha=0.1, solver='lbfgs',
                   tol=0.00001, learning_rate_init=0.001)


     # Realizamos el entrenamiento del modelo
     model = clf.fit(train_UL,out_train_UL)

     output = model.predict(test_UL)

     E2test = r2_score(out_test_UL,output)

     E2test
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/neural_network/multilayer_perceptron.py:1321: DataConversionWarning: A column-ve
  y = column_or_1d(y, warn=True)
0.999414359403538
```

Figure 23. R2 score graph for the Neural Network (Google Colaboratory)

c) Random Forest

R2 score was 0.9898875480445627

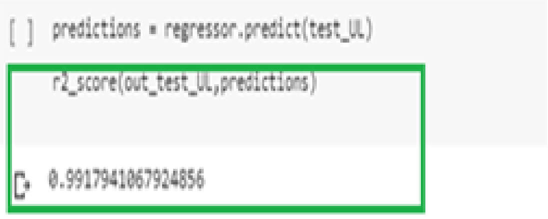


```
[ ]  best_grid = grid_search.best_estimator_

     predictions = best_grid.predict(test_UL)

     r2_score(out_test_UL,predictions)
```

```
0.9898875480445627
```

Figure 24. R2 score plot for Random Forest (Google Colaboratory)

Once this was determined, a finer adjustment was made. using grid search and using only parameters in ranges where a better performance was found, and an r2 Score of 0.9917941067924856

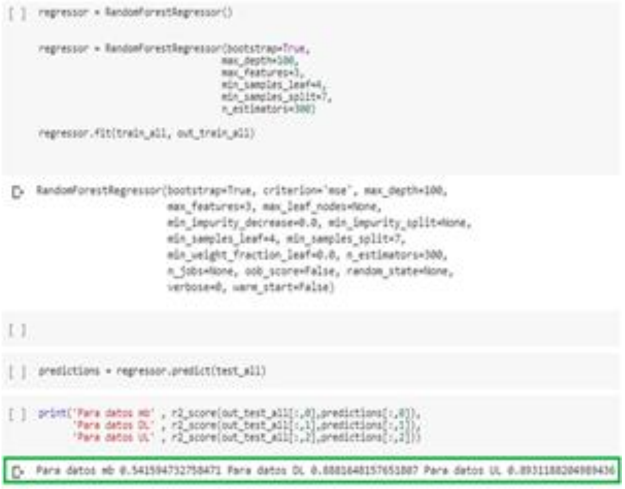


```
[ ] predictions = regressor.predict(test_UL)

    r2_score(out_test_UL,predictions)

    0.9917941067924856
```

Figure 25. R2 score plot for Random Forest with fine adjustment (Google Colaboratory)

5.4. Model to estimate three variables

Once the experiments have been carried out with each variable separately, it was decided to carry out an additional experiment using random forest, since it was the one that best estimated all the variables, to estimate all variables at once and not having an independent model for each variable. In this case, for the selection of variables was chosen for the variables that have a correlation greater than 0.8 with respect to the variables to be predicted. [17]



```
[ ] regressor = RandomForestRegressor()

    regressor = RandomForestRegressor(bootstrap=True,
                                      max_depth=100,
                                      max_features=3,
                                      min_samples_leaf=4,
                                      min_samples_split=7,
                                      n_estimators=300)

    regressor.fit(train_all, out_train_all)

    RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=100,
                          max_features=3, max_leaf_nodes=None,
                          min_impurity_decrease=0.0, min_impurity_split=None,
                          min_samples_leaf=4, min_samples_split=7,
                          min_weight_fraction_leaf=0.0, n_estimators=300,
                          n_jobs=None, oob_score=False, random_state=None,
                          verbose=0, warm_start=False)

[ ]

[ ] predictions = regressor.predict(test_all)

[ ] print('Para datos mb' , r2_score(out_test_all[:,0],predictions[:,0]),
          'Para datos DL' , r2_score(out_test_all[:,1],predictions[:,1]),
          'Para datos UL' , r2_score(out_test_all[:,2],predictions[:,2]))

    Para datos mb 0.5415594732758471 Para datos DL 0.8881648157651807 Para datos UL 0.8931188204989436
```

Figure 26. R2 score graph for the three-variable model (Google Colaboratory)

6. RESULTS

6.1. Result of the Model and Testing Variable PS TRAFIC

Regarding the testing process, to validate the models, we used the test data set and the target variable was estimated. Then the exit of the model was compared with the real data using the metric of the r2 score which is widely used in this type of model where regressions are performed, this metric allows us to identify how efficient is the estimate with respect to all the test data.

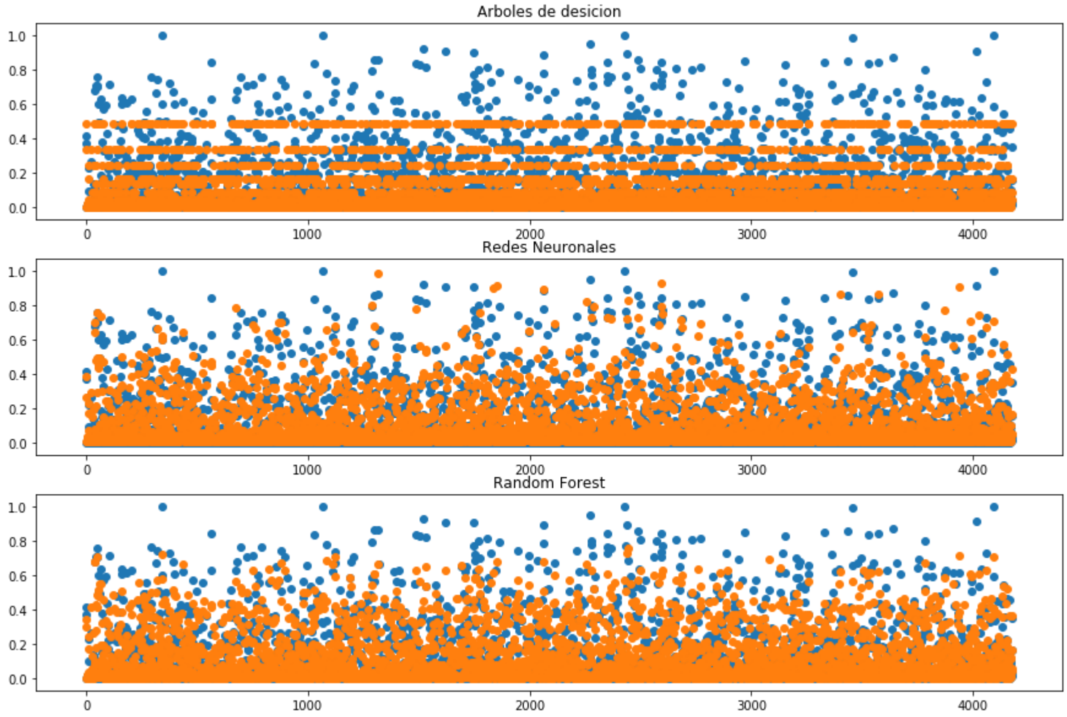


Figure 27. Plot of original and predicted data in each model (Google Colaboratory)

6.2. Outcome of the model and testing variable Vs. RAC.DL.EqvUserNum

Graphs were made of the results obtained and the real data for each model, for said variable, here it could be noted that the decision trees do not have the ability to fully estimate the objective variables, which does not happen with the other models.

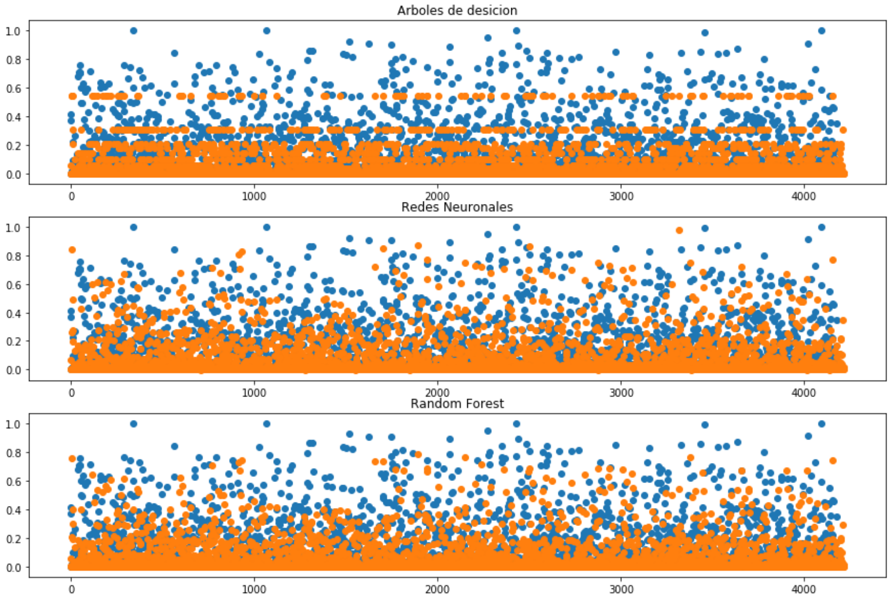


Figure 28. Plot of original and predicted data in each model (Google Colaboratory)

6.3. Outcome of the model and testing variable Vs. RAC.UL.EqvUserNum

Graphs were made of the results obtained and the real data for each model, for said variable, here it could be noted that the decision trees do not have the ability to fully estimate the objective variables, which does not happen with the other models. [19]

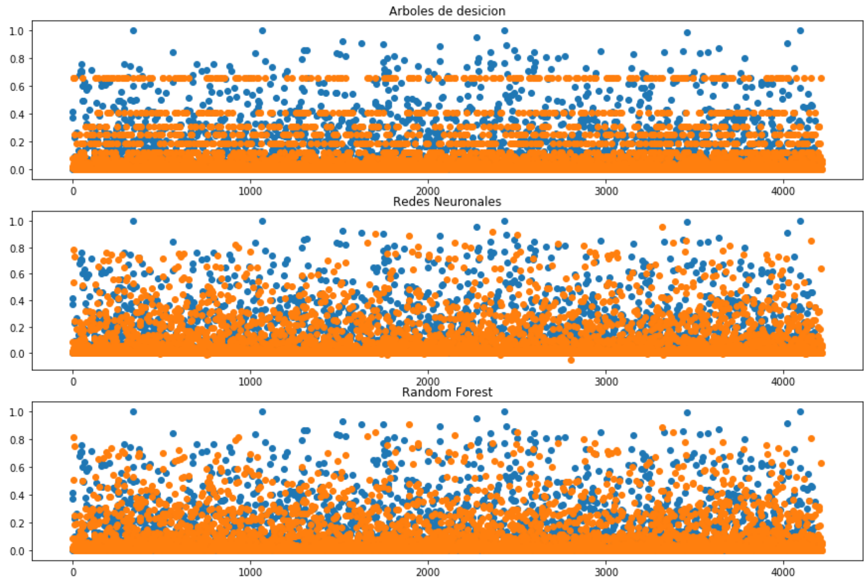


Figure 29. Plot of original and predicted data in each model (Google Colaboratory)

6.4. Discussion and Analysis of Results

As a final part of the development and implementation of the models applied Artificial Intelligence, the numerical analysis of values given in each of the tests, this in order to have a overview and summary of the values obtained, for each of the proposed models and depending on each of the variables of analysis, as detailed below: [18]

Table 1. Analysis of results for all models (Google Collaboratory)

VARIABLE	MODELO	NORMALIZACION	CORRELACIÓN	R2 SCORE
PS TRAFIC MB	Arboles de Decisión	MinMaxScaler	>=0.72	0.6891097048890115
PS TRAFIC MB	Redes Neuronales	MinMaxScaler	>=0.72	0.6979385979700603
PS TRAFIC MB	Random Forest	MinMaxScaler	>=0.72	0.7162649923099472
	Ajuste fino	MinMaxScaler	>=0.72	0.7196744018529605
VS.RAC.DL.EqvUserNum	Arboles de Decisión	MinMaxScaler	>=0.8	0.8324425441778565
VS.RAC.DL.EqvUserNum	Redes Neuronales	MinMaxScaler	>=0.8	0.8883605358098641
VS.RAC.DL.EqvUserNum	Random Forest	MinMaxScaler	>=0.8	0.8915394414631109
	Ajuste fino	MinMaxScaler	>=0.8	0.8898856746512142
VS.RAC.UL.EqvUserNum	Arboles de Decisión	MinMaxScaler	>=0.8	0.9710786186949242
VS.RAC.UL.EqvUserNum	Redes Neuronales	MinMaxScaler	>=0.8	0.999414359403538

VS.RAC.UL.EqvUserNum	Random Forest	MinMaxScaler	>=0.8	0.9898875480445627
	Ajuste fino	MinMaxScaler	>=0.8	0.9917941067924856
Modelo para estimar tres variables	Random Forest	MinMaxScaler	>=0.8	**PS TRAFIC MB=** 0.541594732758471 **VS.RAC.DL.EqvUserNum=** 0.8881648157651807 **VS.RAC.UL.EqvUserNum=** 0.8931188204989436

7. Conclusions

The initial analysis allowed defining which variables shared some type of relationship with the variables we wanted to predict, when using the correlation as a measure of similarity could be filtered to the variables of greater importance and thus use these for the models.

The results were analysed where it was found that the model that the best performance was the model based on random forest, with compared to the other models that were tested, the model based in neural networks was the second that worked best and that of trees decision-maker the one that had the worst performance compared to the others.

It was concluded that neural networks and random forests have had the best results because they have a structure with a greater ability to generalize knowledge.

In the model created to represent all three variables at the same time, the performance of the traffic variable was the one that could not be represented correctly, this behaviour was thought to occur because the other two variables between them share a higher relationship than with that of traffic.

It was concluded that the use of this type of technique is relevant as its use could be extended to applications such as sensors virtual variables, where variables that are complex of measure (due to sensor costs or difficult to access) through variables make them easier to measure.

Future Lines of Work

As part of the improvement of this work, it is planned in the future to implement the ARIMA method for the prediction of this type of data, taking into account that ARIMA is a tool that allows working with time series, whether stationary or non-stationary, as we could see at the beginning of our work the data set that was used contains countless variables, in which the variable that relates times or schedules of greater data traffic in an RBS could also be inserted.
In the future this work may be very helpful in the optimization and monitoring of communications networks, whether mobile or fixed, determining their behaviour and performance more quickly.

Acknowledgements

The authors thank the higher education institutions that contributed to the research with their authors, as place of work such us: Instituto Superior Tecnológico María Natalia Vaca, Ambato, Ecuador, Instituto Superior Tecnológico Pelileo-Baños, Baños, Ecuador

REFERENCES

[1] C. Salgado, «Tecnura,» 15 Mayo 2016. [En línea]. Available: http://www.redalyc.org/jatsRepo/2570/257047577010/index.html.ERICSSON, «QUALCOMM,»

[2] E UNIO 2019. [En línea]. Available: http://itu-apt.org/wrc4p/prez/qualcomm.pdf.

[3] E. contributors, «Telefonía móvil 1G,» 17 Enero 2018. [En línea]. Available: https://www.ecured.cu/index.php?title=Telefon%C3%ADa_m%C3%B3vil_1G&oldid=3049358

[4] Behcet Sarikaya 2000, Packet Mode in Wireless Networks: Overview of Transition to Third Generation, IEEE Comm. Mag. 38 (9): 164-172.

[5] 4GHS, «4GHS,» [En línea]. Available: https://www.4ghs.com/4ghs-network-information.

[6] Hopcroft John E., Ullman Jeffrey D. 1993 "Introduccion to Automata Theory Languages, and Computation".

[7] GALIPIENSO, A., ISABEL, M., Cazorla Quevedo, M. A., Colomina Pardo, O., ESCOLANO RUIZ, F. R. A. N. C. I. S. C. O., & LOZANO ORTEGA, M. A. (2003). Inteligencia artificial: modelos, técnicas y áreas de aplicación. Editorial Paraninfo.

[8] M. Turcaník 2009, "Traffic lights control using recurrent neural networks," Science &Mililtary. vol. 2,

[9] I. Rohde & Schwarz USA, «Rohde & Schwarz USA, Inc.,» [En línea]. Available: https://www.mobilewirelesstesting.com/challenges-in-testing-multistandard-radio-base-stations/.

[10] BELL, D.K., DE TIENNE, D.H. Y JOSHI, S.A. (2003): «Neural networks as statistical tools for business researchers», Organizational Research Methods, Vol. 6, No 2, pp. 236-265

[11] J. B. V., «DESARROLLO Y SIMULACIÓN DE UNA ESTACIÓN BASE,» 10 noviembre 2002. [En línea]. Available: https://scielo.conicyt.cl/pdf/rfacing/v10/art02.pdf.

[12] J. Sánchez García, «e-Gnosis,» 1 Enero 2005. [En línea]. Available: http://www.redalyc.org/articulo.oa?id=73000304.

[13] L. C. Corbalán, «SECIDI,» 1 Noviembre 2006. [En línea]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/4139/Documento_completo__.pdf?sequence=1&isAllowed=y

[14] M. COCA, «IESE,» 2007. [En línea]. Available: http://www.iese.umss.edu.bo/uploads/docs/revista_1277232118.pdf#page=153.

[15] N. S. T. ÁLVAREZ, «TECNURA,» 29 Agosto 2011. [En línea]. Available: http://www.redalyc.org/articulo.oa?id=257020887009.

[16] Olabe, X. B. (1998). Redes Neuronales Artificiales y sus aplicaciones. Publicaciones de la Escuela de Ingenieros.

[17] O. R. Gámez, «Redalyc,» 1 Marzo 2005. [En línea]. Available: http://www.redalyc.org/articulo.oa?id=181517913002.

[18] Pedro Malagón, Patricia Arroba, Samira Briongos, Alex Mauricio Santana, and José M. Moya. 2020. Modeling tree-structured I2C communication to study the behavior of a dielectric coolant in a two-phase immersion cooling system.

[19] In Proceedings of the 2020 Summer Simulation Conference (SummerSim '20). Society for Computer Simulation International, San Diego, CA, USA, Article 33, 1–12. https://dl.acm.org/doi/10.5555/3427510.3427544

[20] R. Gámez2005, «Telefonía móvil celular: origen, evolución, perspectivas,» Ciencias Holguín, p. 9.

AUTHORS

Eduardo Luis Calo Villalva - Master's Degree in Artificial Intelligence (Universidad Internacional de la Rioja), Electronics and Communications Engineer (Technical University of Ambato), teacher in the Electricity career at the Instituto Superior Tecnológico María Natalia Vaca.

Gabriel Alejandro Vaca Ortega - Master's Degree in Systems Management (Universidad de las Fuerzas Armadas ESPE), Electronics and Communications Engineer (Technical University of Ambato), coordinator and teacher in the Electricity career at the Instituto Superior Tecnológico María Natalia Vaca.

Cristina Del Rocio Sanchez Lara (Universidad de Fuerzas Armadas ESPEL - Latacunga), Mechatronics Engineer. Project execution and automation engineer. Instructor of Educational Robotics with Mechatronic Legos. Full-time professor at the Instituto Superior Tecnológico Maria Natalia Vaca in the automotive area – in the city of Ambato.

David Alejandro Jines Espín. **-** Master in Technologies for the management and teaching practice (Pontificia Universidad Católica del Ecuador Ambato Headquarters), Electronics andCommunications Engineer (Technical University of Ambato). SNNA Leveling Teacher at the Technical University of Ambato. Currently Professor of the electronics career at the Instituto Tecnologico Superior Maria Natalia Vaca.

Washington Giovanny Amancha Punina. **-** Master's Degree in Artificial Intelligence (Universidad Internacional de la Rioja), Electronics and Communications Engineer (Universidad Técnica de Ambato). IT Analyst at Tata Consultancy Services (TCS). Professor of Instituto Superior Tecnológico SECAP.I currently work as a coordinator and teacher in the Electronica career at the Instituto Superior Tecnológico María Natalia Vaca.

Angel Arturo Flores Lescano – Master in Management information systems (Universidad Autónoma de los Andes – Ambato), Electronic and Communications Engineer (Universidad Técnica de Ambato), Electronics professor at the Instituto Superior Tecnológico Maria Natalia Vaca.

Alex Mauricio Santana Gallo. **-** Master's Degree in Electronic Systems Engineering (Universidad Politécnica de Madrid), Visiting Researcher HCT-LAB Universidad Autónoma de Madrid 2016 (Universidad de Fuerzas Armadas ESPE), Electronics and Instrumentation Engineer (Universidad de Fuerzas Armadas ESPE). CEO of the company IA-KUNTUR S.A.S. B.I.C. dedicated to the development of artificial intelligence and data science located in the city of Latacunga- Ecuador. Currently teaching the electronic career at the Instituto Superior Tecnológico María Natalia Vaca.

María Fernanda Oñate Pico. **-** Master in Higher Education (Universidad de Fuerzas Armadas ESPE), Teacher in technical specialty (Instituto Superior Tecnológico Pelileo-Baños), Mathematics Teacher (La Salle Ambato Unit), Mechatronics Engineering(Universidad de Fuerzas Armadas ESPE), FINANCIAL MANAGER of the company IA-KUNTUR S.A.S. B.I.C. dedicated to the development of artificial intelligence and data science located in the city of Latacunga-Ecuador.

DEVELOPING E-LEARNING SYSTEM TO SUPPORT VISUAL IMPAIRMENT DURING COVID-19 PANDEMIC

Piya Techateerawat

Engineering Faculty, Thammasat University, Thailand

ABSTRACT

E-learning is a common tool to support the education in variety of scenarios. As the education content can be prepared by the group of specialists, but the skilled teachers are limited in remote area. Also, the contents in most curriculum are planned to distribute to limited skilled people. The gap of education can be full-filled with E- learning system. However, the conventional E-learning is high cost system and not appropriated for rural area. Also, open-source system is complicated to implement and configure in dedicated curriculum. This research is proposed the customized design of E-learning system for supporting visual impaired student with dynamic design during Covid-19 pandemic. The limitation has challenged the research to sharing and testing among the stakeholders. The system is designed and implemented from actual requirements from teachers and students in university level. As a result, the result show student involvement and deliver more content and knowledge to the visual impaired students. The feedback from actual usage also is evaluated.

KEYWORDS

E-learning, Android, Visual Impaired, Covid-19.

1. INTRODUCTION

Electronic learning (E-learning) is a general system to support the learning process and assist the content delivering to the specified learner. The key features of E-learning are to let learner access contents promptly with less constraints e.g. remote area, limited time, lack of teacher and large number of learners. Since network performance and equipment are easily access currently, a number of learners' request more of E-learning system. Also, learners expect to use E-learning for knowledge sharing and on-demand contents.

This paper shows the customized solution of E-learning system that can be implemented in adaptive content with specified need (visual impaired students) The solution also can be applied to other scenarios for other organizations.

2. E-LEARNING SYSTEM

E-learning is shown as critical system for content sharing in business, research and academic. Since the market share of E-learning system reached $2 trillions in 2001, many organizations implement the system for accessible and extra class support [1, 2].

David C. Wyld et al. (Eds): NLP, MLTEC, CLBD, SEAPP, NeTIoT, VLSIE, ITCS, ARIA, DMS - 2021
pp. 409-416, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112331

The main benefit of E-learning system is cost effective, flexibility, accessibility and content distribution. As the system are content centralized and using the technology platform for access anywhere-anytime. Then overall, it reduced the cost of staff, operation task and open the opportunity to more people [3]. It also adds more features that can provide online chat, AI enquiry, reminder, online quiz and content suggestions.

In general, large system is approaching with framework [4-5]. As developing time can be reduced significantly by using the built-in library. The security is conformed by the framework library as well as updating regularly with framework update. The important key is maintenance as framework using the standard model following with guidance.

The one of challenge of implementing E-learning is cost that increased 83 percent from 1998-2003 [6]. This is a struggle for many organizations especially in education system. However, the need of E-learning also pushes to one of the largest share markets [7]. For, specific small organization and limited budget company are difficult to access to the commercialized platform as well as the open source system also required the skilled person to configure and understand that need follow the scheme of open source package.

3. FRAMEWORK

Android framework is a structure that prepare for programmer to develop application. Adobe Illustrator and Figma are the of tools to design on UI and character based. SQLite is use for database of quiz and content storage. This assist for both user and backend side to communicate and interact and share contents over the network. Most of Android framework is based on JDK model, this scheme let programmer customize the framework to meet the requirement by using the universal programming language and layout organization that is compatible to designated devices. Also, most of the framework is based on open source, so it can minimize the cost of license and maintenance.

The advantage of using this framework is let the structure more robust and more secure from the prepared package compared to developing from bare bone. In addition, the framework has prepared the libraries and themes for developer to choose and apply. So, implementation can be rapid and more convenience to meet the requirement.

SQLite provides the up-to-date content structure that can add, delete and modify the content of application on the fly. These support the upcoming content update from the education curriculum. The special request is also to add the simple design features. The project designs the main character model of the simple page and support the handle the flexible of mobile device. This has been focussed for visual impaired student to handle the learning by themselves.

4. PLANNING & DEVELOPMENT

This E-learning system is based on the actual case from Thammasat University. The group of students and teacher in 2020 is based on the research. The development and evaluation are also based on the actual activity that interacts with students.

The design must be compatible to the new mobile devices (e.g. smart phone and tablets). The contents in each screen should be simplified with one main idea.

The purpose of this E-learning system is to support the actual class and student has sufficient knowledge in using computer technology. The requirement and design of this system is survey from in-class students.

As this system need to support visual impaired students, the UI has brainstorm with the user and concluded the need to use the text-to-speech module. As text-to-speech has been developed for support need and assisted to deliver contents to audience [8] as shown in Fig 1. The method has been developed with support new equipment and software [9] as shown in Fig 2. In this system, all stakeholders are discussed in focus group and need to balance between instructor, student and family who support the student with lead to use the API to support the main content similar to strategy in fig 3 [10]. As a result, students are required to access to most standard contents as deliver in actual classroom.

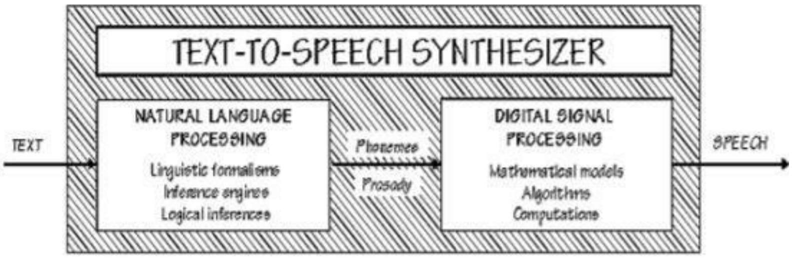


Fig. 1. Text-to-speech [8]

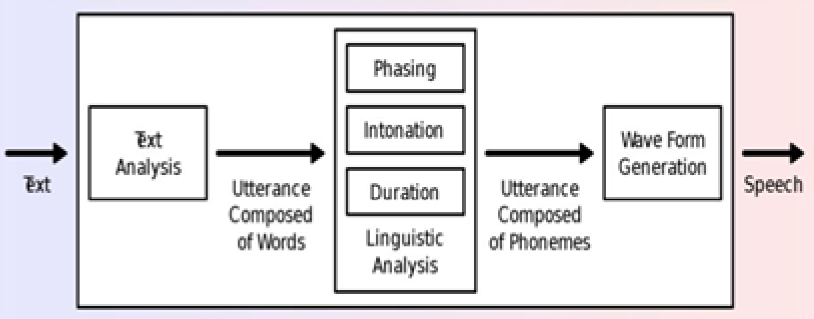
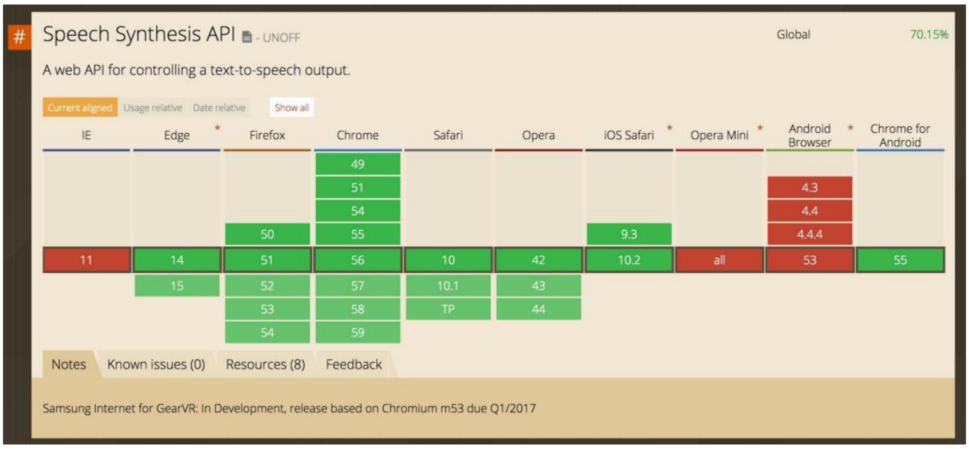


Fig. 2. Text-to-speech System [9]


Speech Synthesis API - UNOFF

Global 70.15%

A web API for controlling a text-to-speech output.

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			49						
			51					4.3	
			54					4.4	
		50	55			9.3		4.4.4	
11	14	51	56	10	42	10.2	all	53	55
	15	52	57	10.1	43				
		53	58	TP	44				
		54	59						

Samsung Internet for GearVR: In Development, release based on Chromium m53 due Q1/2017

Fig. 3. Text-to-speech API support [10]

4.1. Requirement

- The system must be convenient for visual impaired student or family to apply and access.
- Students which do not enrol may apply and register for the interest class.
- Every user can browse the contents in subject.
- The interface should be kids friendly.
- The objective must follow the core curriculum.
- The quiz must the task of activities in the class.
- The flow of application should let the student feel enjoy.

4.2. Application Design

This application is focused to deliver standard content to visual impaired students. As all stakeholder discussed in planning phase. The structure has been designed to support the need of all stakeholders. The ER diagram are also support from the need as shown in fig. 4. Next, the user interface is also break into small step for user to follow step-by-step guideline which replaced overall view information in modern design as shown in Fig. 5. Also, the main contents are brought to highlight focus which in main step e.g. lecture in audio file format and put in main focus of application and ready-to-play as shown in Fig 6.

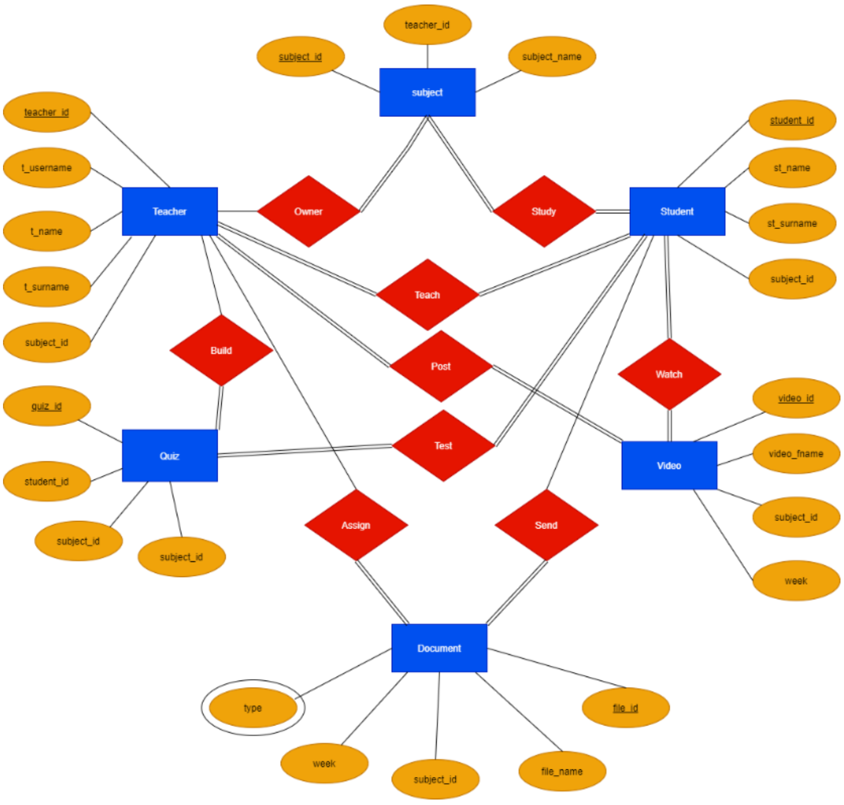


Fig. 4. ER Diagram

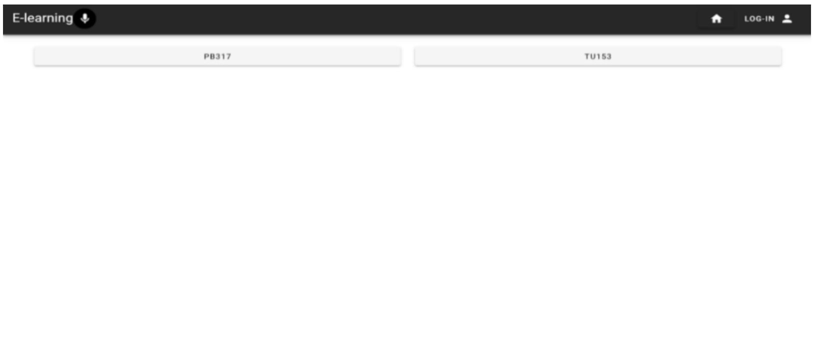
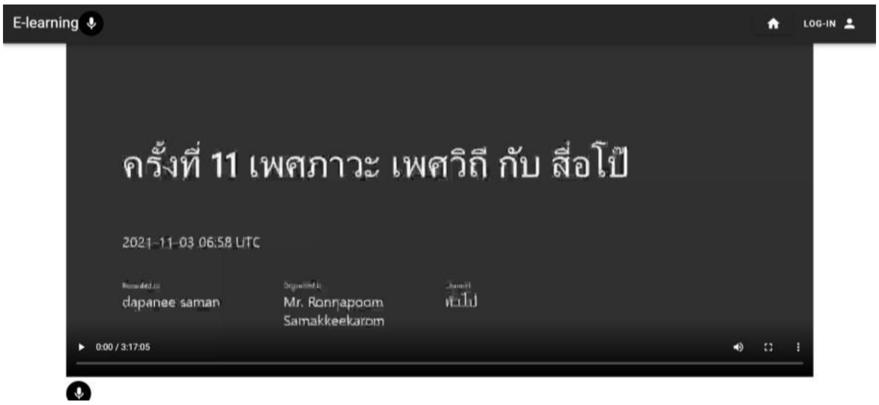


Fig. 5. Simple Content Design



ครั้งที่ 11 เพศภาวะ เพศวิถี กับ สื่อโป๊

2021-11-03 06:58 UTC

dapanee saman

Mr. Ronnapoom Samakkeekarom

ทั่วไป

Fig. 6. Main Content are prioritized in audio format.

In quiz section, the quiz is based on the learning knowledge by pulling up the question from SQLite and arrange by the selected topics. When the user finishes the quiz, simple mark output is given. The quiz allows the user to re-do in the case the gain the understanding. Fig 7 shows the instructor mode that teacher can create the quiz in standard mode to support all class students. However, fig 8 shows that in visual impaired student has quiz in text-to-speech and use voice command to do the quiz.

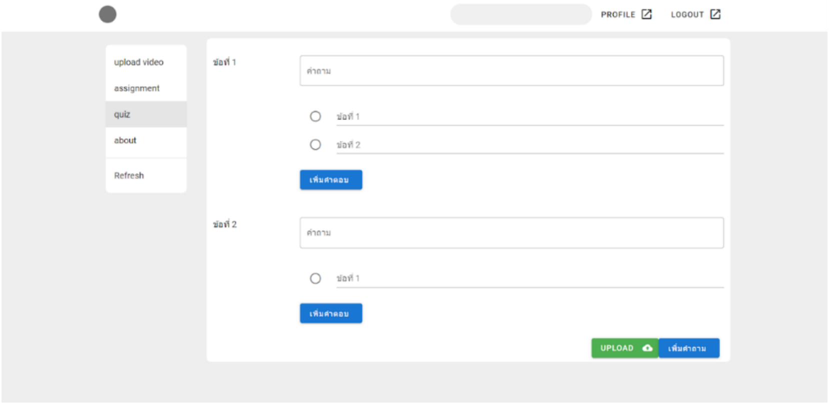
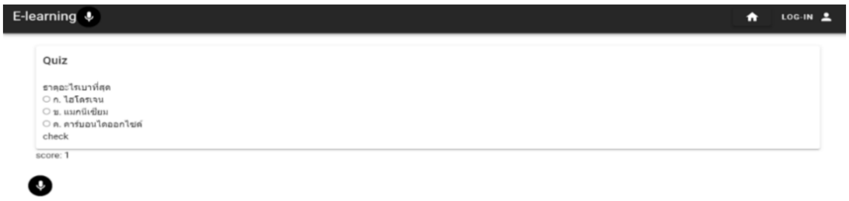


Fig. 7. Quiz preparation for instructor.



Quiz

ธาตุอะไรเบาที่สุด
○ ก. ไฮโดรเจน
○ ข. แมกนีเซียม
○ ค. คาร์บอนไดออกไซด์
check

score: 1

Fig. 8. Quiz mode with voice command.

In quiz section, the quiz is based on the learning knowledge by pulling content from SQLite and using Genymotion to adjust the final result.

5. Result

The result of system is based on the actual testing in university students from Thammasat University. The group of visual impaired students are using this application along with class study and giving the test result and feedback as follows:

Administrator mode:

- Log-in to the system.
- Registering to the system.
- Access to E-learning contents & downloadable contents.
- Some E-learning subject provide optional quiz.
- Search function.
- Forum for learner to share/discuss knowledge.
- File organization Add, Modify, Delete.
- Profile management.
- History organization (Review Content).

User mode:

- Topic selective.
- Content Summary
- Quiz
- Simple UI
- Design from user requirements.

The test result is based on the given time allocation of teacher to the students where the two weeks allocation on everyday task. The average time per slot is 88 minutes as shown in table 1. Also, the support family also given the involvement and provide the feedback result for overall usage as shown in table 2.

Table 1. Summation of Usage Time

	Male	Female
Average usage per week	9	9.7
Average Time per usage in class (minutes)	82	94
Willing to continue to use.	95.71%	97.5%
Average time per usage at home (minutes)	58	64

Table 2. Feedback Result

Feedback (out of 5)	Average	SD
Quality of content	3.95	0.1
To support the class knowledge	3.95	0.1
Quality of application	3.93	0.15
Effective to learning process	3.97	0.07

6. Discussion

As this system is created based on the actual user (visual impaired students) requirement and reviewed by the same group student, our purpose is to gain more contribution of student. The objective is gaining more attention and support the in-class education.

As shown in Table 1, our objective is to evaluate the contribution of learner. As the learners does not intention to use the E-learning before this research. To motivate learners to contribute the new learning tools are shown in visual impaired students. The result also shows the continuous of the involvement for all study period in both school and home. This usage time includes both quiz and learning content based on 2020 students.

Manyinvolvements also require network traffic and computer resource. With this need, in our research has a large scale of computer resource in research lab. However, in real case, the system requires implementation resource in the organization. Therefore, with limited resource and recent technology, cloud server is suggested for the proposed system.

The quality from feedback is shown as above average due to the design and requirement of their choices, however they suggest that the improvement on the next version of application would be expected.

Overall, the customized system shows the significant improved in main objective of E-learning system. This also provides the solution that new trend of users needs the information in their own need format and their own need. The alternative solution is served their task achieved in this area.

7. CONCLUSION

E-learning is challenged in every organization to customize for appropriate for each organization. The difference in objective, members' background and knowledge leads to different expectation. The customization system is a solution with a high cost of out-of-the-box software.

This paper suggest the customized E-learning system based on Android framework. The specific learning group are based visual impaired students. The features of this system are content sharing, online quiz and feedback system to the responding teacher in school. The implementation of deployed system is based on user's requirement that tend to meet with expectation.

The trend of user involvement is improving the participation in most of area especially support special need students.

ACKNOWLEDGEMENTS

We would like to thank Faculty of Engineering, Thammasat University, Thammasat University, Optical and Quantum Communication Research Lab and National Research Council of Thailand.

REFERENCES

[1] Cisco Systems, Cisco Systems IQ Atlas, Cisco, 2001.

[2] K. Fry, "E-learning markets and providers: some issues and prospects", Education + Training, Vol. 43 No. 4/5 pp. 233-9, 2001.

[3] S.M. Furnell, P.D. Onions, U. Bleimann, U. Gojny, M. Knahl, H. F. Roder and P.W. Sanders, "A security framework for online distance learning and training", Internet Research: Electronic Networking Applications and Policy, Vol. 8 No. 3, pp. 236-42, 1998.

[4] S. Alexander, "E-learning developments and experiences", Education + Training, Vol. 43 No. 4/5 pp. 240-8, 2001.

[5] S.H. Garrison and D.J. Borgia, "Using an Internet based distance learning to teach introductory finance", Campus-Wide Information Systems, Vol. 16 No. 4, pp. 136-9, 1999.

[6] D. Lance, " Venture captial viewpoints and E-learning futures", The Business of E-learning: Bringing Your Organization in the Knowledge Economy, Univeristy of Technology Sydney, 2000.

[7] P. Henry, "E-learning Technology, Content and Services", Education + Training, Vol. 43 No. 4, pp. 249-55, 2001.

[8] Itunuoluwa Isewon, Jelili Oyelade, Olufunke Oladipupo. "Design and Implementation of Text To Speech Conversion for Visually Impaired People" CORE, 2021.

[9] Raiyetunbi, Oladimeji Jude, Ayeh Emmanuel. "An Interactive Cloud Based User Oriented, Dynamic and Intelligent Text-To-Speech Module" Easpublisher, 2021.

[10] Wittawat Patcharinsak. "Conversation App with Web Speech API" konoesite.com. [Online]. Available: https://konoesite.com/สรา-conversation-app-ง่ายๆด้วย-web-speech-api- be54db5505f4. [Accessed: November. 9, 2020].

AUTHORS

Piya Techateerawat is a lecturer of Computer Engineering at Thammasat University. He received his B.Eng. from University of New South Wales, Australia with Honors in 2004. He continued his PhD study at Royal Melbourne Institute of Technology University, Australia, where he obtained his PhD in Wireless Sensor Network Security. His current interests involve applications of Sensor Network, Security and Quantum Cryptography.

AUTHOR INDEX