Artificial Intelligence and Applications

David C. Wyld,
Dhinaharan Nagamalai (Eds)

Computer Science & Information Technology

- 9th International Conference on Artificial Intelligence and Applications (AIAP 2022), February 26~27, 2022, Vancouver, Canada
- 11th International Conference on Information Theory (IT 2022)
- 11th International Conference on Mobile & Wireless Networks (MoWiN 2022)
- 12th International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2022)
- 15th International Conference on Network Security & Applications (CNSA 2022)
- 9th International Conference on Bioinformatics and Bioscience (ICBB 2022)

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: 978-1-925953-62-6
DOI: 10.5121/csit.2022.120401- 10.5121/csit.2022.120410

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

Preface

9th International Conference on Artificial Intelligence and Applications (AIAP 2022), February 26~27, 2022, Vancouver, Canada, 11th International Conference on Information Theory (IT 2022), 11th International Conference on Mobile & Wireless Networks (MoWiN 2022), 12th International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2022), 15th International Conference on Network Security & Applications (CNSA 2022) and 9th International Conference on Bioinformatics and Bioscience (ICBB 2022) was collocated with 9th International Conference on Artificial Intelligence and Applications (AIAP 2022). 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 AIAP 2022, IT 2022, MoWiN 2022, CCSEIT 2022, CNSA 2022 and ICBB 2022 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, AIAP 2022, IT 2022, MoWiN 2022, CCSEIT 2022, CNSA 2022 and ICBB 2022 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 AIAP 2022, IT 2022, MoWiN 2022, CCSEIT 2022, CNSA 2022 and ICBB 2022.

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

Abbas Al-Thaedan,	Al-Muthanna University, Iraq
Abdalhossein Rezai,	University of Science and Culture, Iran
Abdel-Badeeh M. Salem,	Ain Shams University, Egypt
Abdelhadi Assir,	Hassan 1st University, Morocco
Abdellatif I. Moustafa,	Umm AL-Qura University, Saudi Arabia
Abderrahim Siam,	University of Khenchela, Algeria
Abderrahmane Ez-Zahout,	Mohammed V University, Morocco
Abdullah,	Chandigarh University, India
Abhishek Shukla,	R D Engineering College, India
Addi AIT-MLOUK,	Cadi Ayyad university, Morocco
Addisson Salazar,	Universitat Politècnica de València, Spain
Afaq Ahmad,	Sultan Qaboos University, Oman
Ahmad A. Saifan,	Yarmouk University, Jordan
Akhil Gupta,	Lovely Professional University, India
Ali Abdrhman Mohammed Ukasha,	Sebha University, Libya
Ali Al-Sabbagh,	Ministry of communucation, Iraq
Ali Asghar Rahmani Hosseinabadi,	University of Canada, Canada
Altynbek Sharipbay,	Eurasian national university, Kazakstan
Amal Azeroual,	Mohammed V University, Morocco
Amari Houda,	Networking & Telecom Engineering, Tunisia
Amel Ourici,	University Badji Mokhtar of Annaba, Algeria
Amizah Malip,	University of Malaya, Malaysia
Anita Yadav,	Harcourt Butler Technical University, Kanpur
Ankur Singh Bist,	Chief AI Scientist, India
Anuj Singal,	GJU S &T, HISAR, India
Aridj Mohamed,	Hassiba Benbouali University Chlef, Algeria
Assem Abdel Hamied Moussa,	Chief Eng egyptair, Egypt
Assem mousa,	University of Bristol, England
Atanu Nag,	IFTM University, India
Atilla Evcin,	University of Afyon Kocatepe, Turkey
Baihua Li,	Loughborough University, UK
Bhagyashree SR,	ATMECE, Mysore
Bin Zhao,	JD.com Silicon Valley R&D Center, USA
Bouchra Marzak,	Hassan II University, Morocco
Brahim Lejdel,	University of EL-Oued, Algeria
Cheng Siong Chin,	Newcastle University, Singapore
Cherkaoui Leghris,	Hassan II University of Casablanca, Morocco
Christian Mancas,	Ovidius University, Romania
Cyrus Azimi,	Tehran University of Medical Sciences, Iran
Dinesh Reddy,	SRM University, India
Dinyo Omosehinmi,	colossus technology, Nigeria
Dumitru Dan Burdescu,	University of Craiova, Romania
Ebenezer Bonyah,	Kumasi polytechnic, Ghana

Technically Sponsored by

Computer Science & Information Technology Community (CSITC)

Artificial Intelligence Community (AIC)

Soft Computing Community (SCC)

Digital Signal & Image Processing Community (DSIPC)

9th International Conference on Artificial Intelligence and Applications (AIAP 2022)

11th International Conference on Information Theory (IT 2022)

11th International Conference on Mobile & Wireless Networks (MoWiN 2022)

12th International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2022)

15th International Conference on Network Security & Applications (CNSA 2022)

9th International Conference on Bioinformatics and Bioscience (ICBB 2022)

LIGHTBLUE: NURTURE YOUR PERSONAL CHATBOT

Xiang Zhang[1], Yan Liu[1], Gong Chen[1] and Sheng-hua Zhong[2]

[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
[2]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China

ABSTRACT

Chatbot has long been an important research topic in artificial intelligence and attracts lots of attention recently. Despite significant advancements in language ability, the interactions between users and chatbots are rather generic, short-term, and transnational. It has always been challenging to develop truly personal chatbots and even more difficult to establish long-term, affective connections. This paper first brings up "nurture" as a new interaction mode with chatbots. We introduce the nurture framework and accordingly design the learning algorithm and nurture functions. Then we present LightBlue – a platform that allows non-professionals to nurture personal chatbots from scratch. Experiments on both closed- and open-domain tasks validate the proposed framework and demonstrate a promising method for facilitating long-term interaction between users and chatbots.

KEYWORDS

Personal Chatbot, Conversation Agent, Nurture, Human-chatbot Interaction, Long-term Relationship

1. INTRODUCTION

Chatbot, a computer program which can communicate with humans in natural language, has been put into study since the birth of Artificial Intelligence (AI) [1]. It is also called as machine conversation system, virtual agent, dialogue system, and chatterbot [2]. A chatbot can be programmed to mimic humans, answer questions, serve as a personal assistant, and so on [3]. In recent years, the applications of chatbots boom in industry, emerging in many fields including education, social media, finance, catering, etc [4,5,6]. Siri as one of the most representative chabot products, has attracted billions of users [6,7,8].

The work on chatbots can be divided into two groups: task-oriented chatbots and general chatbots. Task-oriented chatbots are designed to serve specific task, such as a hotel booking or a technical support service, and the conversation cannot go beyond the topic scope of the system. General chabots are designed to pass the Turing test, or engage in social chit-chat with users [9], with no specific target or topic scope. They can be also used to engage users in open-domain human-computer conversations for entertainments or emotional companionship [10]. However, the current communication with chatbots are superficial and generic; users are more likely to regard them as tools or strangers [11].

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 01-21, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120401

Personal chatbots begin to receive the public attention in recent years. Rather than serving a broad audience, personal chatbots cater to a specific user, aiming to conduct a personal and in-depth conversation: with intimate knowledge of the user's preferences, needs, and habits, it provides necessary information and assistance [12]. This direction, which provides end-users with increased flexibility and control over their own personal chatbots, is yet to be explored. For example, companion chatbot, a kind of chatbot pursues long-term interaction and affectional connections with specific users, is still in its infancy [5,11].

The inherent difficulty in developing personal chatbots is balancing generality and specificity. On the one hand, it should be sufficiently flexible to be customized by all users; on the other hand, it should know the specific user to a certain degree to facilitate long-term and cumulative communication.

To provide a flexible conversation mechanism with both generality and specificity, the prerequisite is to be aware of the user's personal information. However, collecting user's information for each conversation not only requires a huge amount of work, but also is unavailable in many cases. Recently, many efforts have been put into building datasets providing users' information for dialogue samples. For example, PERSONA-CHAT dataset from Facebook AI research [13] contains over 10,000 dialogs on more than 1,000 different personas. However, the dialogs are collected from crowd-sourcing workers, thus can not fully represent real user-bot interaction. And the designed personas are hard to address the massive number of user profiles in real world.

User-specific conversation requires every sentence to be processed in a specific context, while the diversity of end-users makes the content of conversations unpredictable. The answers cannot be prepared in advance, as different users may interpret the same sentence differently. Furthermore, people would change their identities, preferences and status with time and the environment. Instead of preparing massive amounts of conversation samples, we emphasize the learning ability of a personal chatbot -- A personal chatbot should be able to continuously learn and grow from the interactions, and adapt to new changes after deployment.

Alan Turing has once raised the question in his proposal "Computing Machinery and Intelligence" – "*Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's?*" [14]. The nature-versus-nurture debate concerning the development of human intelligence has lasted for a long time [15]. However, "nurture" hasn't received enough attention in today's data-driven machine learning methods. Inspired by Turing's "children and education" viewpoint, we propose to nurture a chatbot by users. That is, we first create a simple and general chatbot model. After deployment, it can be nurtured by end-users to become a personal chatbot. We define "nurture" as a new interaction mode with chatbots, which differentiates from traditional interactions in the following aspects:

- The learning algorithm grows gradually during the nurture process.
- Users can guide the algorithm to develop in a particular direction through the nurture process.

In this way, the learning algorithm is able to learn about the particular user and grow after deployment. By considering the end-users' initiative to engage in a cumulative and in-depth conversation, "nurture" contributes to the development of personal chatbots in the following aspects:

- In biological and social sciences, it is assumed that intelligence develops through learning culturally valued knowledge and skills in a human, social environment [16]. There are also powerful evidences showing the critical role of nurture in cognitive development [17].
- Nurture can be helpful to develop a close, interpersonal relationship, such as social bonds, between users and chatbots.
- Nurture plays a significant role in how humans learn to converse. In addition to passively observing others' conversations, we also actively adjust and correct our speech in response to feedback woven throughout our own conversations [18,19]. It has been proved that interaction, rather than exposure, is necessary for successful language acquisition [20,21].

In this paper, we define the nurture framework in detail and design the corresponding learning algorithm for nurture. Then we integrate the learning algorithm and nurture functions into a chatbot nurture platform called LightBlue, which enables nonprofessional users to nurture personal chatbots. There are three main contributions of this work:

- This paper is the first work of introducing the nurture framework as a new interaction mode between users and chatbots.
- Under the nurture framework, we further propose a corresponding learning algorithm for nurture and design a bunch of nurture functions.
- In this paper, we develop LightBlue – a chatbot nurture platform, integrating the learning algorithm and nurture functions.

The rest of the paper is organized as follows. We start by reviewing related work. Then we describe the nurture framework, the learning algorithm, nurture functions and the LightBlue chatbot nurture platform. Two experiments are conducted to verify the effectiveness of the nurture framework and we report the results in the following section. We visualize and analyze our experimental results and then discuss the features of the proposed framework. Finally, we conclude this paper.

2. RELATED WORK

In this section, we review the chatbot applications and technologies. There are several studies on learning after deployment and user-chatbot interactions that are related to our nurture framework design.

2.1. Chatbot Applications and Technologies

Following the first chatbot ELIZA [22] in 1960s, researchers have devoted to the study on chatbots. Many of the early works aimed to pass the Turing test [14]. For example, PARRY simulated paranoid behaviors and successfully fooled its judges [23]. In 1995, the Loebner Competition held its first unrestricted Turing test with no limits on the topics discussed. More general-purpose chatbots have been developed, such as MegaHAL [24], CONVERSE [25], ELIZABETH [26], A.L.I.C.E [27] and Mitsuku (Loebner prize winner in 2013, 2016, 2017, 2018) [28]. With the emergence of messaging platforms, not only have chatbots for socializing and small talk (e.g. A.L.I.C.E, Cleverbot, Simsim, Tay) gained popularity, but task-oriented chatbots have also attracted a lot of attention [29]. Bagousse estimates that chatbots in customer service will save businesses up to 11 billion dollars by 2025 [30]. Chatbot bot has never been so popular as recent days.

During the last half century, chatbot technologies have varied from keywords matching, rule-based models, retrieve-based models, to generative models [31]. With the advancement of

computational power and the availability of big data, the current trend is to use neural networks to generate responses. One of the largely used structure in neural language generation is Sequence-to-Sequence (Seq2Seq) [32] model, adopted from machine translation. For example, [33] uses Sina Weibo dataset to train a neural responding machine based on Seq2Seq framework; Neural attention (or alignment) are then proposed to improve the Seq2Seq models by associating salient items in the source sequence with the generated item in the target sequence [34,35,36]; Zhou et al.[37] adds memory mechanism to address the emotion factor in large-scale conversation generation.

Deep learning has contributed a lot to develope intelligent chatbots in the sense of generating more human-like sentences and talking in a natural way [31]. However, these models are usually developed in the laboratory by professional engineers. No matter targeting a specific task or open-domain conversation, they're usually trained using large corpora of crowd-sourced or scraped conversations [38]. Thus, there are two common limitations: First, they tend to give generic answers – some "Jack of All Trades"-type answers will be applied to a different question, such as "*I don't know.*" [7,10]; Second, they usually lack consistent personality due to the large amount of mixed training data. And we wouldn't call these chatbots truly "understand" the dialogue but rather create an illusion of intelligence and empathetic understanding [39]. Since they lack the ability to further develop or learn about the user during interaction, it is hard to have long-term and deep conversation. Due to the same reason, users often lose interest after a short interaction [40].

2.2. Learning After Deployment

For personal chatbots, it's not feasible to collect a large amount chatting data between one particular user and the chatbot. Since users are free to communicate with the system in any way they wish, it is impossible to pre-plan the interaction logic and domain knowledge in advance [41]. A simple intuition is to allow chatbots to develop after deployment. For example, Huang et al. [42] proposes Evorus, a crowd-powered chatbot with can automate itself during deployment. Though Evorus learns to automatically select high-quality response over time, it learns from paid crowd workers rather than end-users. In another word, it can not learn or update without crowd oversight, which is not only labor-extensive, but also involve privacy issues.

A common way to learn from users is to require user's feedback during the interaction [13,43]. Typically, existing methods require paid annotators to provide scalar rewards or to use particular templates to so that the feedback can be used by the model [44,45,46]. For example, some chatbots learn actively during conversation in the question answering (QA) setting [47,48]. Some works also allow chatbots to learn directly from the natural dialogues, e.g., Hancock et al. [19] proposes the self-feeding chatbot capable of extracting new training examples from the conversations in which it participates. However, most of the work are targeting task-oriented chatbots or a specific scenario, but not designed for a general setting. And none of them allow users to develop chatbots from scratch, meaning that they still need large dataset to pre-train the model before deployment.

While learning from multiple paid workers or crowd-sourcing feedback may be effective for a specific task, the feedback can be chaotic and even conflicting when applied to a broader context. Tay, Microsoft's Twitter-based chatbot which learned from massive Twitter users, was forced to shut down for being incredibly racist [49].

2.3. User-Chatbot Interactions

Learning from a particular user can be very different from learning from crowd users in terms of data volume, dialogue content, interaction depth or consistency. Additional modes of interaction for efficient learning should be investigated. Current research on user-chatbot interactions focuses primarily on interface design in order to facilitate usage or improve the user experience. Jain et al. [50] provides a context view for chatbots to address a mismatch between the chatbot's state of understanding (also called context) and the user's perception of the chatbot's understanding. It also supports intuitive interactions with context values, allowing users to modify them simply and efficiently. Candello et al. [51] studies the influence of typefaces on the perception of humanness in chatbots. There's also an increasing interest in multi-modal interaction, such as combining visual information [52,53,54]. Some work also considers the interaction in special cases. Seering et al. [55] study chatbots supporting or engaging in group or multiparty interactions. And [56] explores the repair preferences for conversational breakdowns. Luger et al. [8] reveals a number of design challenges arising from the gap between user expectations and actual experience, such as how a chatbot can reveal its current state, or how to design system feedback and clearly communicate the goal of the system. However, the interaction between users and the underlying learning algorithm is still underexplored.

Though personal chatbots are not designed for specific tasks, the conversations will become specific and personal when deployed to the end-users. Thus, we introduce a new interactive mode – nurture, which enables non-professional users to train the learning algorithm using simple functions and to obtain personal chatbots.

3. SYSTEM DESIGN

In this section, we first define the nurture framework as a new interaction mode between users and chatbots. Following the nurture framework, we propose a novel computational model - pattern matching model for nurture, along with a set of simple yet effective nurture functions. Algorithms of those functions are explained formally. Finally, we put the proposed framework into practice by creating the LightBlue chatbot nurture platform, which can be directly used by non-professional end-users.

3.1. Nurture Framework

The basic idea of "nurture" is to allow end-users to guide the development of the intelligent algorithm and help it adapt to a new environment. Intuitively, it can be conducted in the same way that humans provide feedback, such as correcting incorrect responses, indicating a preference for certain actions, and awarding rewards. As a new interaction mode, "nurture" allows end-users to participate in the development of intelligent algorithms and enables joint development of engineers and non-professional users. In contrast to conventional user-chatbot interaction, which views chatbots as the interface to other functions, "nurture" goes beyond the interface and emphasizes interaction with the underlying learning algorithm. We define the nurture framework by three key components:

- Learning algorithm: an algorithm with a simple and general initial state, and can be nurtured in an open and dynamic environment through long-term interaction.
- Nurture functions: functions that can be used by nonprofessional users to guide the development of the learning algorithm.
- Feedback information: feedback from the chatbot that enables users to monitor the effect of nurture actions and the chatbot's state, and adjust their nurture strategy accordingly.

3.2. Pattern Matching Model for Nurture

Following the nurture framework, the learning algorithm should be able to grow in the open and dynamic environment. Specifically, in the application of personal chatbots, the algorithm should be able to adapt and grow through the interaction with unfamiliar users. Meanwhile, users can guide the development of the algorithm on a long-term and open-ended scale.

Here we adopt the classical idea of the pattern matching method used in chatbots, and develop it into a pattern matching model for nurture. The pattern matching technology can be traced back to the first chatbot ELIZA [22], which matches responses to pre-defined keywords. Based on this simple idea, we further develop heuristic pattern generation and update methods, allowing the chatbot to learn from new data incrementally. Its simplicity and transparency allow us to see how the new growth characteristics affect the original algorithm and to observe the role of nurture.

3.2.1. Basic concepts and symbols

Figure 1 shows the important components in the pattern matching model. The following are detailed explanations of basic concepts in the model:

- *Message*: the user's input utterance.
- *Response*: the chatbot's output utterance to the user.
- *Basic unit*: word or punctuation.
- *Pattern*: a basic unit or a concatenation of basic units.
- *Goodness*: the degree to which a response is considered to be a good one.
- *Link*: the weighted edge between a pattern and a response.
- *Link activation*: the process by which the weight of a link is increased.
- *Number of activation(s)*: The number of link activation(s).
- *Dialog turn:* a message from the user and a following response from the chatbot.

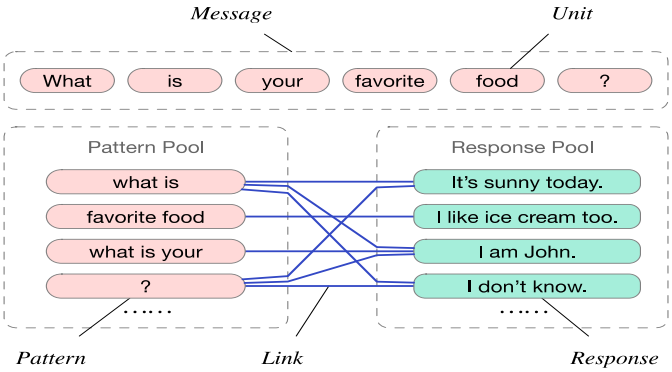


Figure 1. The architecture of pattern matching model for nurture.

Table 1 contains a list of frequently used notations in this paper.

Table 1. Summary of frequently-used notations.

Symbol	Definition
P	A set of patterns
p_i	The pattern i in set P
R	A set of responses

r_j	The response j in the set R
$l(p_i)$	Length of p_i (i.e. the number of basic units it contains)
$g(r_j)$	The goodness of response r_j
$w(p_i, r_j)$	The weight of the link between $p_i$and r_j
$s(p_i, r_j)$	The strength between p_i and r_j
t	The dialog turn index

3.2.2. Chatting

Chatting is the most fundamental function of a chatbot model, which outputs in the form of natural language based on the dialog history. Figure 2 illustrates the workflow of the chatting function using the example input message "*What is chatbot?*". The chatting function can be divided into two sessions: *response retrieval* and *update process*.

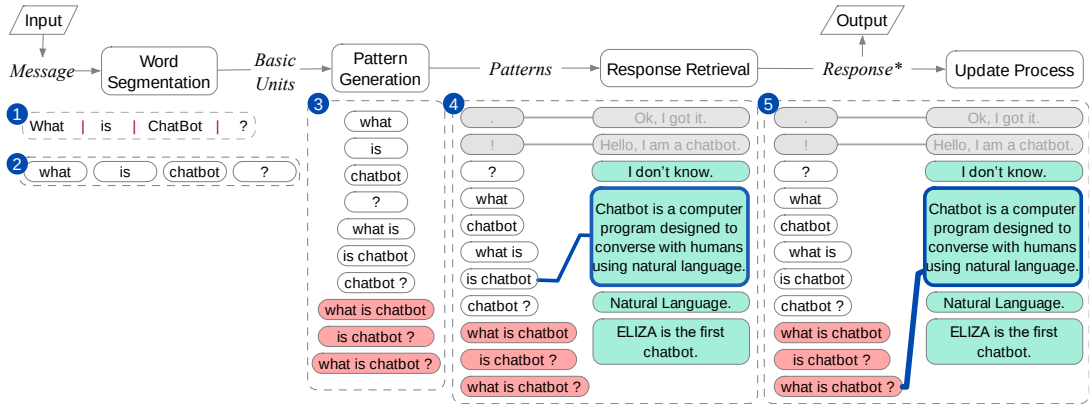


Figure 2. The workflow of the chatting function, illustrated with the input message "What is a chatbot?". The sentence is first divided into four basic units (words and punctuation). Following that, these basic units are used to generate related patterns. And links of those generated patterns are used to retrieve response*, which is also the output of the chatbot. The links between generated patterns and response* will be strengthened in the final stage.

In the *response retrieval* session, the user's input message is first decomposed into basic units. Then we define a heuristic function to generate patterns based on the basic units:

1. Each basic unit is a pattern.
2. If two patterns are adjacent to each other in the message, they are concatenated to form a new pattern.

Thus, for an input message, we can generate a set of patterns $P = \{p_i\}(i = 1, \dots, n)$ according to the above rules. Then through the links between patterns and responses, we can get a set of m related responses $R = \{r_j\}(j = 1, \dots, m)$ from the response pool. For the link between pattern *pi* and response r_j at dialog turn t, we represent its weight by $w_t(p_i, r_j)$. Besides, we define the strength between pattern p_i and response r_j as follows:

$$s_t(p_i, r_j) = \frac{w_t(p_i, r_j)}{\sum_{k=1}^{m} w_t(p_i, r_k)} \times l(p_i) \qquad (1)$$

where $l(p_i)$ is the number of basic units in p_i. Then, we define the goodness of response r_j at dialog turn t as follows:

$$g_t(r_j) = \max_{i \in \{1,\dots,n\}} s_t(p_i, r_j) \quad (2)$$

Based on the goodness $g_t(r_j)$, the best response r^* is recognized as follows:

$$r^* = argmax_{r_j \in \{r_1,\dots,r_m\}} g_t(r_j) \quad (3)$$

Then, the best response r^* is selected as the output response to the user.

After retrieval, here comes the *update process* session. When the response r^* is approved by the user, all patterns in set *P* and responses in set *R* will be linked. Then the link activation is performed for all links of r^* by updating the weight $w_t(p_i, r^*)(i = 1, \dots, n)$ as follows:

$$w_{t+1}(p_i, r^*) = w_t(p_i, r^*) + 1, \quad (4)$$

The weights of links between all patterns and responses are initialized as follows:

$$w_0(p_i, r_j) = 0 \quad (5)$$

where $i = 1, \dots, n$ and $j = 1, \dots, m$. By the *update process*, the algorithm continues to self-renewal during chatting which enables self-learning without supervision.

3.2.3. Nurturing Functions

Based on social scientists' research on learning in human communication [57], we further designthree nurturing functions for the proposed nurturable pattern matching learning algorithm: *Change*, *Like*, and *Analysis*. This section discusses the details.

Via the *Change* function, users can override the chatbot's response r^* with a preferred response r'. If r' is a new response that is not in the current response set *R* the chatbot will add r' into *R*. The change function will increase the link weight associated with the preferred response r' iteratively until the goodness of r' exceeds that of r^*. The algorithm of the change function is provided in Algorithm 1.

Algorithm 1 Change function

Input: $r^*, \psi, f(\cdot), w_t(p_i, r')$
Output: $w'_t(p_i, r')$

if $r' \notin R$ **then**
  $R = R \cap \{r'\}$
end if
compute $g_t(r^*)$ by Eq. (2)
compute $g_t(r')$ by Eq. (2)
while $g_t(r') \leq g_t(r^*)$ **do**
  for i = 1, ..., n **do**
    $w_t(p_i, r') = w_t(p_i, r') + 1$
  end for
  compute $g_t(r')$ by Eq. (2)
end while
for $i = 1, \dots, n$ **do**
  $w'_t(p_i, r') = w_t(p_i, r')$

Algorithm 2 Like function

Input: $r^*, \psi, f(\cdot), w_t(p_i, r^*)$
Output: $\boldsymbol{w'_t(p_i, r^*)}$

for $v = 1, \dots, \psi$ **do**
  for $i = 1, \dots, n$ **do**
    $w_t(p_i, r^*) = w_t(p_i, r^*) + f(v)$
  end for
end for
for $i = 1, \dots, n$ **do**
  $w'_t(p_i, r') = w_t(p_i, r')$
end for
return $w'_t(p_i, r')$

Algorithm 3 Analysis function

Input: response of ChatBot r^*
Output: pattern p^* that results in r^*

14: **end for** 15: **return** $w_t'(p_i, r')$	1: **for** $i = 1, \dots, n$ **do** 2: compute $s_t(p_i, r^*)$ by Eq. (1) 3: **end for** 4: $p^* = \underset{p_i \in \{p_1, \dots, p_n\}}{\text{argmax}} \; s_t(p_i, r^*)$ 5: **return** p^*

Via the *like* function, user can encourage the good response of the chatbot. If a response is liked by the user, the link weight associated with that response is increased, and the chatbot is more likely to output that response in the future when a similar message is given. The like function can be used multiple times to enhance its effectiveness. The algorithm of the like function is provided in Algorithm 2. The weighted function $f(\cdot)$ in Algorithm 2 can be determined by the user. For example, a simple case is linear weighting: $f(v) = 1$ for $v = 1, \dots, \psi$, which means that every *like* is equally important.

Analysis is a function used to provide the explanation of the chatbot's response. The user can viewthe pattern that results in the output response r^* via the analysis function. The algorithm of *Analysis* is provided in Algorithm 3.

3.3. LightBlue – Chatbot Nurture Platform

In this section, we describe how we integrate the learning algorithm and nurture functions into the nurture platform – LightBlue. The details of implementation are explained, including its interface, interaction process and the process of user nurturing. With LightBlue end-users are able to nurture the chatbot while chatting with it.

3.3.1. Interface

Figure 3 shows the interface of the LightBlue platform, which includes the chatting window, function buttons, chatbot image, and vocabulary. It can be mainly divided into two parts. On the left is the user-chatbot conversation interface, including the conversation history, the chatting box, nurture functions and common operation buttons. On the right shows the state of the learning algorithm, including the chatbot image and learned vocabulary. As users nurture the chatbot, we visualize the evolution of the learning model through the chatbot's image and vocabulary repertoire.

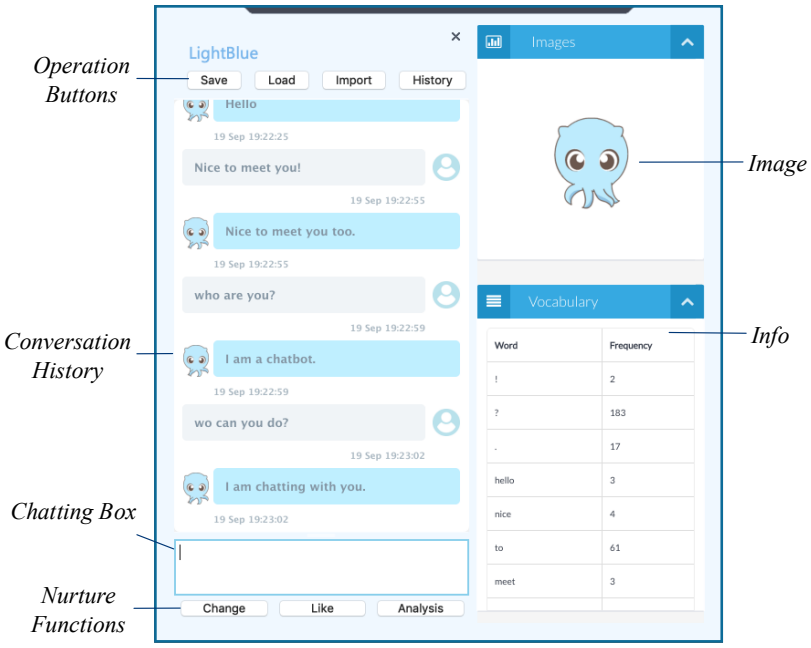


Figure 3. The interface of LightBlue. The left part contains the chatting window and functional buttons; and the right part contains an image of the chatbot and its vocabulary, which visualizes the progress of the nurture.

3.3.2. Interaction Process

This subsection describes how users interact with the underlying algorithm through the LightBlue platform. When the program is launched, a new initialized chatbot is automatically created. The*Save* and *Load* buttons allow you to save the current chatbot model and load an existing chatbot model. Users use the chatting box to send messages, and the chatbot responds using the chatting function described previously. The new dialogs will then be included in the conversation's history.As the conversation goes, the algorithm will learn and grow. And the right anthropomorphic chatbot image reflects the changing state. The right lower box contains more specific information.

3.3.3. Nurture Functions

We design three buttons to assist users in nurturing the chatbot based on the nurture function defined in the previous section. Users can click *like* button to express their preferences for the chatbots' responses. After receiving likes from users, the chatbot uses Algorithm 2 to increase the link weight of the preferred response. To change a response, the user can enter a preferred answer in the chatting box and click the *change* button. Then Algorithm 1 will be performed to revise the response. The *Analysis* button allows the user to determine why the chatbot provides a particular response. An example result of the analysis are showed in Figure 4, explaining the relevant information leading to the response.

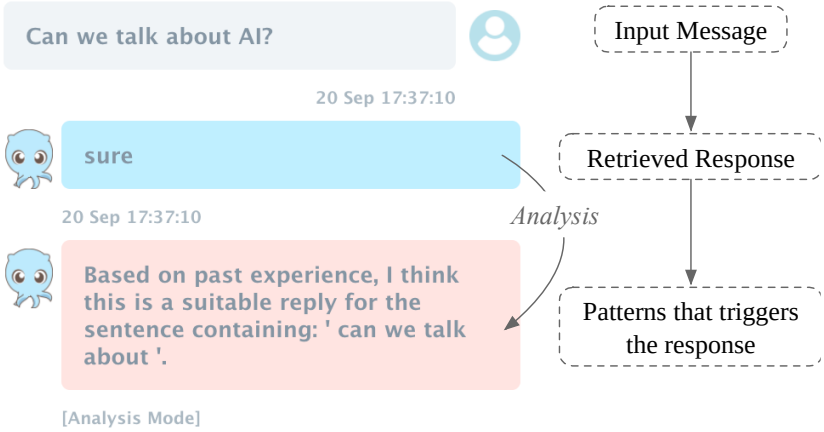


Figure 4. An example of the analysis function in LightBlue. The feedback from analysis shows the most contributing pattern "*can we talk about*" that leads to the retrieved response "*sure*".

4. RESULT

We conduct experiments in both closed-domain and open-domain settings to demonstrate the feasibility of the nurture framework for developing personal chatbots. In the closed-domain experiment, non-professional users are required to communicate with chatbots about a given topic and material. After verifying the effectiveness of the nurture framework, we further design an open-domain experiment to test if chatbots can adapt to different users and be applied to different task scenarios through nurture.

4.1. Initialization

As showed in Figure 5, every chatbot is initialized with three patterns (i.e. period, exclamation mark, and question mark), three responses (i.e. "*Ok, I got it*", "*Hello, I am a chatbot*", and "*I don't know*") and corresponding links between them. Initially, all links have a weight of 1. Thus, in the initial state, the vocabulary contains only three words: period (.), exclamation mark (!), and question mark (?).

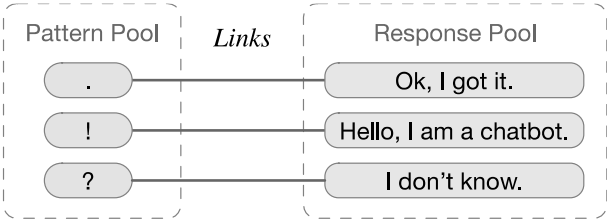


Figure 5. The initial state of Pattern Matching Model. It contains three patterns, three responses, and corresponding links between them.

4.2. Closed-domain Experiment

In this experiment, we selected "*A brief history of artificial intelligence*" as a specific topic, and provided all participants with a 2,000-word article and a 1,000-word vocabulary. Participants were asked to nurture their chatbots to use words in the vocabulary to answer questions related to the content of the article. After two weeks of training, 34 chatbots were collected. The

conversation history between users and chatbots is typically measured in thousands of turns, with a vocabulary ranging from hundreds to thousands.

4.2.1. Response Quality

To determine whether the chatbot is capable of accurately answering questions, we invite three volunteers to read the article and then pose questions about the content. According to the content relevance, grammar correctness, and vocabulary, we finally screen 132 questions and randomly assign three chatbots to answer each question.

To grade the quality of the chatbots' responses, three categories are used: *correct*, *partly correct or relevant to the question*, and *not relevant to the question*. Three volunteers are assigned to categorize each response. To avoid ambiguity, we count only the responses that were assigned to the same category by all three volunteers.

Then we define the correct rate as the proportion of correct responses to all responses, and relative rate as the proportion relevant responses (including correct responses) to all the responses. The evaluation results of all 357 responses are shown in the Figure 6, where 83.5% responses are related to the questions and 66.4% responses correct with the best chatbot has the relative rate of 91.7% and correct rate 83.3%.

Figure 6. 66.4% of responses from 34 chatbots to 132 questions are correct, are 83.5% are related to the questions.

4.2.2. Growth

To examine the growth ability of chatbots, we choose a chatbot with an average performance and also an average length of dialog history (about 3500 turns) from the 34 collected chatbots. Then we replicate the nurture process and test it with the same questions across time periods. With the increase of chatting dialogues, the chatbot's performance has improved (see Figure 7).

Due to the fact that users place a different emphasis on nurture, the growth rates of chatbots may vary at different stages. However, there is no obvious bottleneck of the growth, which demonstrates the robustness of our algorithm to different nurture methods and the characteristics of sustainable growth.

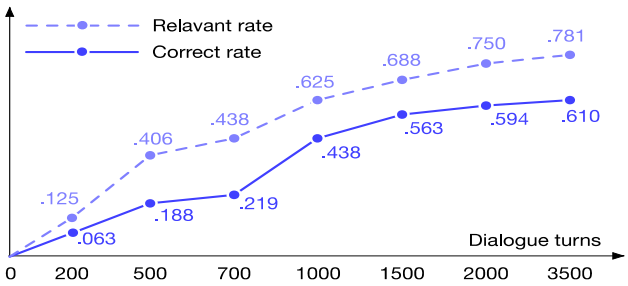


Figure 7. The response quality of the chatbot as nurture process goes.

4.2.3. Diversity

Even when the topic and content are the same, people's expressions vary widely due to the diversity of human language. We calculate the average rate of overlap between the conversation histories of each chatbot. Figure 8 shows the result, with an average overlap rate of 8.03%. And the overlap rate between conversation corpus and testing questions is even lower (less than 1% on average). Achieving high response quality, our proposed nurture framework demonstrates its prominent superiority in dealing with various conversational styles. Table 2 shows several examples where chatbots give diverse but reasonable responses to questions with objective answers in the material. These two questions are basic and simple questions which have clear answers in the article, and chatbots give out diverse responses. It's also an important observation that the same chatbot maintains a consistent conversational style. For example, Chatbot 14 tends to answer the question in a more casual manner, using examples or descriptions.

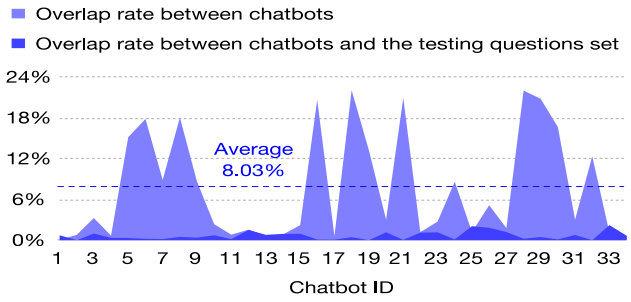


Figure 8. Overlap rate of chatbots' conversation history; and overlap rate between chatbots' conversation history and testing question set.

4.3. Open-Domain Experiment

In the open-domain experiment, we distribute LightBlue platform to 101 participants (mean age = 20.7, 36 females) for 21 days. The participants can nurture chatbots according to their preferences with unlimited topics, while basic greeting skills are required for all chatbots. To test the nurtured chatbots fairly and easily, we also provided a restricted vocabulary of 2024 words drawn from the common English vocabulary. After the experiment, we collected 98 chatbots, with conversation histories ranging from hundreds to tens of thousands turns.

Table 2. Diverse answers to the same questions related to the material. For the question with a clear answer in the article, human-nurtured chatbots are capable of providing diverse yet reasonable responses.

Question	***What is a virtual assistant?***
Chatbot 10	Virtual assistant is capable of accessing data and answering questions.
Chatbot 9	Serving as virtual assistants, the majority of these chatbots perform customer service or customer engagement functions, answering questions or offering suggestions for purchase.
Chatbot 14	Siri 2010, Google Now 2012, Alexa 2015, Cortana 2015, and Google Assistant 2016.
Question	***What is AI?***
Chatbot 8	It is the theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages.

Chatbot 21	Artificial Intelligence: a.k.a machine intelligence, is intelligence demonstrated by the machine to display human behaviour.
Chatbot 14	If the user doesn't know they're speaking to a bot.

4.3.1. Diversity

We test 98 chatbots using six commonly used English greetings. The results in Table 3 shows that,on average, 85% of responses are appropriate, and the average overlap rate of responses is about 14%. This reflects the diversity of training data and training methods from different users, which results in the diversity of chatbots.

Table 3. Correct rate and overlap rate of responses to common greetings.

Greetings	**Correct rate**	**Overlap rate**
Hello!	0.88	0.32
Good morning!	0.88	0.18
Nice to see you.	0.79	0.07
How are you?	0.93	0.06
How's everything?	0.86	0.04
Have a nice day!	0.77	0.21
Average	**0.85**	**0.14**

4.3.2. Consistency

Nowadays, one of the most common challenges or complaints about chatbots is their lack of consistency. That is, responses to different questions may be inconsistent. We select 8 common topics which may result in inconsistency, including food, education, weather, mood, hobbies, color, living information and friends. We ask more than ten questions about each topic, and the same questions may be repeated to the chatbot if they yield different responses. If all of the responses to a specific topic are consistent, the chatbot is considered consistent in that topic. Given that not all chatbots are capable of discussing the eight topics, we only examine those reasonable and related responses.. It is worth noting that for LightBlue, we test all the nurtured chatbots, and the consistent answers are those consistent within one chatbot. We regard *LightBlue* as a consist chatbot on one topic if all answers are consist (which can be different for different chatbots). For the chatbot from the Internet, we only test the model once.

Table 4 summarizes the results of the consistency evaluation, demonstrating that human-nurtured chatbots (*LightBlue*) always provide consistent responses, whereas the average consistency rate for general Internet chatbots is low (below 50%). As a task-oriented chatbot, *Siri* also maintains a consistent personality. However its topic scope is somehow limited compared to others since it is not designed for general chatting. And in many cases, especially unrelated to pre-defined tasks, it tends to respond with "*I'm sorry, I'm afraid I don't have an answer to that.*".

Table 4. Consistent rates of responses to contextual topics.

Chatbot	**Topics**	**Consistent Answers**	**Consistent Rate**
LightBlue	8	8	**1**
Mitsiku [58]	8	3	0.38
Cleverbot [59]	8	3	0.38

Eviebot [60]	8	4	0.50
SimSimi [61]	8	1	0.13
Siri [62]	5	5	**1**

4.3.3. Adaptability to different task scenarios

In addition to basic greetings, users can train chatbots to perform a variety of tasks based on their preferences. By their self-feedback after the experiment, the application scenarios designed for personal chatbots include domain knowledge (e.g. math), entertainment (e.g. playing games, telling jokes), personal assistant (e.g. notes), adviser (e.g. career advice), customer service (e.g. booking tickets), chit-chat, and emotional companions (see Table 5). The majority of users expressed satisfaction with their chatbots' performance in the designed scenario. This again demonstrates the robustness of the nurture framework in adapting to a variety of tasks, corpora, and users.

Table 5. Customized tasks in open-domain experiment.

Customized tasks	**Number of chatbots**	**Average history length (turns)**
Chit-chat	50	1020
Personal assistant	4	2613
Emotional companion	11	4978
Domain knowledge	15	5038
Entertainment	4	1198
Adviser	5	1969
Customer service	6	3104

4.3.3. Social Bond

While it is difficult to quantify the social bond directly, it is reflected in the depth, quantity, and sentiment of dialogues. To better explore the connections between users and chatbots, we examine only chatbots that have been nurtured for more than two weeks.

Throughout the experiment, users are asked to record a new session whenever they change the subject or end the conversation. During two or three weeks of nurturing, our chatbots averaged 21 conversation-turns per session (CPS), which is larger than the CPS of most chatbots conversations between users and nearly equal to the CPS of the 6th generation of XiaoIce [63]. For comparison, the CPS of the 1st generation of XiaoIce is 5. Because each session's conversational content is focused on a single subject, the large CPS indicates that the conversation has progressed to a certain level of depth.

We further examine how the amount of conversation changes throughout the nurture process. To eliminate the influence of start and end periods, we take a week in the middle as the observation window. Figure 9 shows the daily conversation turns between users and chatbots during 19th Nov to 25th Nov. With the process of nurture, the conversation between users and chatbots increases gradually. It also indicates that the users' interests are gradually increasing and their bonds are getting tighter.

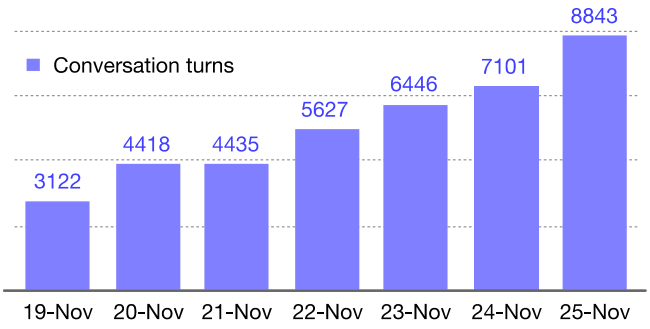


Figure 9. The number of daily conversation turns between users and chatbots from 19th Nov to 25th Nov.

To analyse the sentiment of the conversation, we examine the frequency with which positive and negative words are used. As a reference, the Sentiment Lexicon [64] is used, which contains a list of approximately 6800 positive and negative opinion or sentiment words in English. For simplicity, we refer to the conversation history of all chatbots nurtured on the LightBlue platform as the LightBlue Corpus. As shown in Table 6, we compare the positive and negative word frequencies in the LightBlue Corpus to those in other commonly used datasets for chatbot training. The result shows that datasets containing human-chatbot dialogues (LightBlue and ConvAI) have a lower Negative-Positive ratio, implying a higher level of positivity. And LightBlue has the highest Positive Frequency and the lowest Negative-Positive ratio. It is noteworthy that for both positive frequency and negative frequency, LightBlue Corpus is higher than other datasets. The high frequency of positive and negative words indicates that users are more emotionally engaged when interacting in a nurture-based context.

Table 6. The frequency of positive words, the frequency of negative words, and the Negative-Positive ratio in LightBlue and public datasets.

Dataset	Positive Frequency	Negative Frequency	N/P Ratio
LightBlue	0.244	0.039	0.159
Corpus ConvAI [65]	0.051	0.012	0.233
Twitter [66]	0.038	0.028	0.734
Cornell movie dialogues [67]	0.019	0.017	0.896
Open subtitles [68]	0.028	0.020	0.724

5. DISCUSSION

This chapter summarizes the features of LightBlue and discusses its limitations.

5.1. Features of LightBlue Chatbot Nurture Platform

Through LightBlue chatbot nurture platform, non-professional users can easily nurture and customize a chatbot. Despite of the simple design of the learning algorithm and small amount of dialogues, human-nurtured chatbots achieve satisfying performance in both closed-domain and open-domain experiments. Below we further summarize the features of LightBlue and how the nurture framework benefits developing personal chatbots.

1. Adaptability and Growth

Users can nurture chatbots to perform a variety of tasks via the LightBlue platform, with application scenarios ranging from entertainment to personal assistants. The learning algorithm also demonstrates growth characteristics during the nurturing process. And there's no obvious bottleneck of the growth trend as the conversation corpus grows. The capacity for sustained growth will also facilitate long-term interaction between users and chatbots.

2. Robustness

Through the two experiments (closed-domain and open-domain), we find that LightBlue is robust to a variety of tasks, nurture corpora, nurture methods, and end-users. That is, it does not require professionals or particular process to nurture a chatbot.

3. Diversity and Consistency

The main challenges faced by nowadays general chatbots are uniform answer and inconsistent personality. Our chatbots, when nurtured by different end users, can be naturally personalized and maintain diversity, overcoming the uniform responses caused by big data. Additionally, because the conversation corpus originates from a single user, it naturally has a distinct style and is less prone to inconsistencies.

4. Social functions

According to statistics, the public will lose interest in a general chatbot after a short interaction. However, the participants in our two experiments continue to communicate with their chatbots for 2-3 weeks, with conversations ranging from hundreds to tens of thousands of turns. Not only does their conversation frequency increase over time, but they also maintain a high CPS (around 21), which is significantly higher than the average user-chatbot conversation on the Internet. And the interaction involves more emotion than common dialogues. This suggests a new social role for chatbots in our lives, lays the groundwork for long-term user-chatbot relationships, and even facilitates emotion functions such as affective bonding.

5.2. Limitations

In this work, we only implement one learning algorithm – the pattern matching model for nurture. As a retrieval algorithm, It has inherent limitations such as the inability to generate new questions and answers. At the same time, the hyperparameters and initialization of the model may influence the experimental results. Further investigation of experimental environments is necessary.

Because the experiment participants are all college students who are reasonably familiar with computer operations and share similar cultures and habits, the experiment results may be influenced by group characteristics. Additional subjects with diverse backgrounds, such as middle school students, workers, and the elderly, will be needed to evaluate the robustness of the LightBlue platform.

6. CONCLUSION

This work introduces "nurture" as a new mode of interaction between humans and chatbots, and defines the nurture framework for developing personal chatbots. We further propose a novel learning algorithm and design nurture functions accordingly. Then we implement the LightBlue

chatbot nurture platform and distribute it to non-professional users. Experiments in both closed-domain and open-domain are designed to verify the effectiveness of the nurture framework. Through the LightBlue platform, non-professional users can develop diverse and personalized chatbots. They're able to perform diverse tasks with small amount of training data while remaining robust to end-users, application scenarios, and nurture methods. Nurture, as an interaction mode, demonstrates a new way for chatbots to perform social functions and promotes long-term connections between users and chatbots.

ACKNOWLEDGEMENTS

This work is supported by Innovation and Technology Fund from the Innovation and Technology Commission of Hong Kong, project number ITS/110/19.

REFERENCES

[1] S. A. Abdul-Kader and J. Woods, "Survey on chatbot design techniques in speech conversation systems," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 7, 2015.

[2] B. A. Shawar and E. Atwell, "Chatbots: are they really useful?," in *Ldv forum*, vol. 22, pp. 29–49, 2007.

[3] K. Ramesh, S. Ravishankaran, A. Joshi, and K. Chandrasekaran, "A survey of design techniques for conversational agents," in *International Conference on Information, Communication and Computing Technology*, pp. 336–350, Springer, 2017.

[4] R. Dale, "The return of the chatbots," *Natural Language Engineering*, vol. 22, no. 5, pp. 811–817, 2016.

[5] S. Chandel, Y. Yuying, G. Yujie, A. Razaque, and G. Yang, "Chatbot: efficient and utility-based platform," in *Science and Information Conference*, pp. 109–122, Springer, 2018.

[6] R. M. Thomas, S. Punna, M. C. K. K. Reddy, and B. RamanaMurthy, "Survey on artificially intelligent chatbot," *Journal of Applied Science and Computations*, vol. 6, no. 1, pp. 85–94, 2019.

[7] M. Nuruzzaman and O. K. Hussain, "A survey on chatbot implementation in customer service industry through deep neural networks," in *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, pp. 54–61, IEEE, 2018.

[8] E. Luger and A. Sellen, "Like having a really bad pa: the gulf between user expectation and experience of conversational agents," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5286–5297, ACM, 2016.

[9] B. Chantarotwong, "The learning chatbot," *Final year project.[Online]: http://courses. ischool. berkeley. edu/i256/f06/projects/bonniejc. pdf*, 2006.

[10] R. Yan, ""chitty-chitty-chat bot": Deep learning for conversational ai.," in *IJCAI*, vol. 18, pp. 5520–5526, 2018.

[11] L. Clark, N. Pantidi, O. Cooney, P. Doyle, D. Garaialde, J. Edwards, B. Spillane, E. Gilmartin, C. Murad,C. Munteanu, *et al.*, "What makes a good conversation?: Challenges in designing truly conversational agents," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 475, ACM, 2019.

[12] F. Daniel, M. Matera, V. Zaccaria, and A. Dell'Orto, "Toward truly personal chatbots: on the development of custom conversational assistants," in *Proceedings of the 1st International Workshop on Software Engineering for Cognitive Services*, pp. 31–36, ACM, 2018.

[13] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2204–2213, 2018.

[14] C. Machinery, "Computing machinery and intelligence-am turing," *Mind*, vol. 59, no. 236, p. 433, 1950.

[15] R. J. Sternberg and E. Grigorenko, *Intelligence, heredity and environment*. Cambridge University Press, 1997.

[16] S. Scarr, "Biological and cultural diversity: The legacy of darwin for development," *Child Development*, vol. 64, no. 5, pp. 1333–1353, 1993.

[17] R. Winston and R. Chicot, "The importance of early bonding on the long-term mental health and resilience of children," *London journal of primary care*, vol. 8, no. 1, pp. 12–14, 2016.
[18] M. A. Bassiri, "Interactional feedback and the impact of attitude and motivation on noticing l2 form," *English Language and Literature Studies*, vol. 1, no. 2, p. 61, 2011.
[19] B. Hancock, A. Bordes, P.-E. Mazare, and J. Weston, "Learning from dialogue after deployment: Feed yourself, chatbot!," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3667–3684, 2019.
[20] J. Berko-Gleason, "Insights from child language acquisition for second language loss," *The loss of language skills*, vol. 23, no. 1, pp. 12–23, 1982.
[21] M. G. Werts, M. Wolery, A. Holcombe, and D. L. Gast, "Instructive feedback: Review of parameters and effects," *Journal of Behavioral Education*, vol. 5, no. 1, pp. 55–75, 1995.
[22] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.
[23] K. M. Colby, *Artificial paranoia: a computer simulation of paranoid process*. Pergamon Press, 1975.
[24] J. L. Hutchens, "How to pass the turing test by cheating," *School of Electrical, Electronic and Computer Engineering research report TR97-05. Perth: University of Western Australia*, 1996.
[25] B. Batacharia, D. Levy, R. Catizone, A. Krotov, and Y. Wilks, "Converse: a conversational companion," in *Machine conversations*, pp. 205–215, Springer, 1999.
[26] B. A. Shawar and E. Atwell, *A comparison between Alice and Elizabeth chatbot systems*. University of Leeds, School of Computing research report 2002.19, 2002.
[27] B. AbuShawar and E. Atwell, "Alice chatbot: Trials and outputs," *Computación y Sistemas*, vol. 19, no. 4, pp. 625–632, 2015.
[28] R. Higashinaka, K. Imamura, T. Meguro, C. Miyazaki, N. Kobayashi, H. Sugiyama, T. Hirano, T. Makino, and Y. Matsuo, "Towards an open-domain conversational system fully based on natural language processing," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 928–939, 2014.
[29] P. B. Brandtzaeg and A. Følstad, "Why people use chatbots," in *International Conference on Internet Science*, pp. 377–392, Springer, 2017.
[30] A. L.Bagousse,"Customer Service Chatbots Are Redefining Customer Service," 2021. https://blog.smart-tribune.com/
[31] R. Yan, Y. Song, X. Zhou, and H. Wu, "Shall i be your chat companion?: Towards an online human-computer conversation system," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 649–658, ACM, 2016.
[32] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, pp. 3104–3112, 2014.
[33] L. Shang, Z. Lu, and H. Li, "Neural responding machine for short-text conversation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1577–1586, 2015.
[34] Z. Yin, K.-h. Chang, and R. Zhang, "Deepprobe: Information directed sequence understanding and chatbot design via recurrent neural networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2131–2139, ACM, 2017.
[35] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.
[36] A. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A decomposable attention model for natural language inference," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2249–2255, 2016.
[37] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu, "Emotional chatting machine: Emotional conversation generation with internal and external memory," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[38] I. V. Serban, R. Lowe, P. Henderson, L. Charlin, and J. Pineau, "A survey of available corpora for building data-driven dialogue systems: The journal version," *Dialogue & Discourse*, vol. 9, no. 1,pp. 1–49, 2018.
[39] M. Dibitonto, K. Leszczynska, F. Tazzi, and C. M. Medaglia, "Chatbot in a campus environment: Design of lisa, a virtual assistant to help students in their university life," in *International Conference on Human-Computer Interaction*, pp. 103–116, Springer, 2018.

[40] J. Zhang, N. M. Thalmann, and J. Zheng, "Combining memory and emotion with dialog on social companion: A review," in *Proceedings of the 29th international conference on computer animation and social agents*, pp. 1–9, ACM, 2016.
[41] R. Yan, Y. Song, and H. Wu, "Learning to respond with deep neural networks for retrieval-based human-computer conversation system," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 55–64, 2016.
[42] T.-H. K. Huang, J. C. Chang, and J. P. Bigham, "Evorus: A crowd-powered conversational assistant built to automate itself over time," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 295, ACM, 2018.
[43] J. Li, A. H. Miller, S. Chopra, M. Ranzato, and J. Weston, "Learning through dialogue interactions by asking questions," *arXiv preprint arXiv:1612.04936*, 2016.
[44] V. Rieser and O. Lemon, *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation*. Springer Science & Business Media, 2011.
[45] H. Zhang, H. Yu, and W. Xu, "Listen, interact and talk: Learning to speak via interaction," *arXiv preprint arXiv:1705.09906*, 2017.
[46] B. Liu, G. Tür, D. Hakkani-Tür, P. Shah, and L. P. Heck, "Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems," in *NAACL-HLT*, 2018.
[47] J. E. Weston, "Dialog-based language learning," in *Advances in Neural Information Processing Systems*, pp. 829–837, 2016.
[48] J. Li, A. H. Miller, S. Chopra, M. Ranzato, and J. Weston, "Dialogue learning with human-in-the-loop," *arXiv preprint arXiv:1611.09823*, 2016.
[49] R. Price, "Microsoft is deleting its AI chatbot's incredibly racist tweets," *Business Insider*, 2016.
[50] M. Jain, R. Kota, P. Kumar, and S. N. Patel, "Convey: Exploring the use of a context view for chatbots," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 468, ACM, 2018.
[51] H. Candello, C. Pinhanez, and F. Figueiredo, "Typefaces and the perception of humanness in natural language chatbots," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 3476–3487, ACM, 2017.
[52] B. Huber, D. McDuff, C. Brockett, M. Galley, and B. Dolan, "Emotional dialogue generation using image-grounded language models," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 277, ACM, 2018.
[53] Y. Matsuyama, A. Bhardwaj, R. Zhao, O. Romeo, S. Akoju, and J. Cassell, "Socially-aware animated intelligent personal assistant agent," in *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*, pp. 224–227, 2016.
[54] D. DeVault, R. Artstein, G. Benn, T. Dey, E. Fast, A. Gainer, K. Georgila, J. Gratch, A. Hartholt, M. Lhommet, *et al.*, "Simsensei kiosk: A virtual human interviewer for healthcare decision support," in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*,pp. 1061–1068, International Foundation for Autonomous Agents and Multiagent Systems, 2014.
[55] J. Seering, M. Luria, G. Kaufman, and J. Hammer, "Beyond dyadic interactions: Considering chatbots as community members," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 450, ACM, 2019.
[56] Z. Ashktorab, M. Jain, Q. V. Liao, and J. D. Weisz, "Resilient chatbots: Repair strategy preferences for conversational breakdowns," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 254, ACM, 2019.
[57] E. A. Schegloff, G. Jefferson, and H. Sacks, "The preference for self-correction in the organization of repair in conversation," *Language*, vol. 53, no. 2, pp. 361–382, 1977.
[58] mitsuku, "mitsuku." Web Program, Sep 2019. Retrieved Sep 21, 2019 from https://www.pandorabots.com/mitsuku/.
[59] Cleverbot, "Cleverbot." Web Program, Sep 2019. Retrieved Sep 21, 2019 from https://www.cleverbot.com/.
[60] Eviebot, "Eviebot." Web Program, Sep 2019. Retrieved Sep 21, 2019 from https://www.eviebot.com/en/.
[61] simsimi, "simsimi." Web Program, Sep 2019. Retrieved Sep 21, 2019 from https://www.simsimi.com/Home.

[62] J. R. Bellegarda, "Spoken language understanding for natural interaction: The siri experience," in *Natural Interaction with Robots, Knowbots and Smartphones*, pp. 3–14, Springer, 2014.
[63] L. Zhou, J. Gao, D. Li, and H.-Y. Shum, "The design and implementation of xiaoice, an empathetic social chatbot," *Computational Linguistics*, vol. 46, no. 1, pp. 53–93, 2020.
[64] B. Liu and M. Hu, "Opinion mining, sentiment analysis, and opinion spam detection," *Dosegljivo: https://www. cs. uic. edu/ liub/FBS/sentiment-analysis. html# lexicon.[Dostopano 15. 2. 2016]*, 2004.
[65] V. Logacheva, M. Burtsev, V. Malykh, V. Polulyakh, and A. Seliverstov, "Convai dataset of topic-oriented human-to-chatbot dialogues," in *The NIPS'17 Competition: Building Intelligent Systems*, pp. 47–57, Springer, 2018.
[66] C. Tan, L. Lee, and B. Pang, "The effect of wording on message propagation: Topic- and author-controlled natural experiments on twitter," in *Proceedings of ACL*, 2014.
[67] C. Danescu-Niculescu-Mizil and L. Lee, "Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs.," in *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.
[68] P. Lison and J. Tiedemann, "Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles," 2016.

AI-ENABLED CYBERNETIC ANALYTICS OF SECURITY MODELS FOR SMART SERIOUS GAMES-BASED MOBILE OPERATING SYSTEMS

Abid Ali[1], Faisal Jamil[2], Taegkeun Whangbo[1] and Shabir Ahmad[1*]

[1]Department of Computer Science,
University of Engineering and Technology, Taxila, Pakistan
[2]Computer Engineering Department, Gachon University, Korea

ABSTRACT

Smart serious games are games which are primarily designed for a serious job such as education and training rather than entertainment. Mobile Phones are an essential enabler of smart serious games. Over the past few years, there has been an exponential growth in the security of the cyber-physical system of multiple operating systems. Security challenges have emerged from access scenarios in conventional operating systems. However, in Mobile Operating Systems, there is a lack of systematic study about mobile security measurement. Artificial Intelligence (AI) is one of the critical features which exists inside mobile operating systems. AI enhances the working of Mobile OS functionality by providing thinking on user behavior and extending the functionality. We explore some key security models for different mobile Operating System providers. In particular, the comparison of security policies implemented by multiple mobile operating system vendors. It's challenging to deal with and study the related data models and measurements for mobile operating system platforms. This paper collects, analyzes, and provides effective decisions about the existing research on multiple mobile operating systems. We consider the built-in security, Authentication, Data Protection, Device protection, Application Security, Device Wipe, Mobile Device Management, Cooperate Managed e-mail, Active Sync Support, Device Firewall, Security Certifications, and Virtualizations. AI supports all these features to help OS to extend its functionality for effective game management. We have focused on these components that clearly understand the Mobile operating system structure and support services. General, as well as specific features, are explored about the different platforms of the operating systems. The systematic research study is applied to these platforms to capture the response, and based on these responses, and we developed results based on the feedback and research responses. This comparison aims to make the right choice for mobile users to device the best mobile with a high-security Operating System installed for mobiles partaking serious games in cyber-physical environment.

KEYWORDS

Serious Games, Mobile OS, Symbian OS, Android OS, Windows OS, Blackberry OS, iOS, Artificial Intelligence.

1. INTRODUCTION

Serious games have been evolved with the recent advances in internet of things (IoT) and Cyber-physical systems (CPS). The collection of sensors data and consume it to make informed decisions during a gameplay has taken the serious games into a whole next level, which in literature often known as smart serious games [1-2]. The use of IoT nodes, instead of

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 23-38, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120402

sophisticated controller offer a variety of advantages, such as context-awareness, pervasiveness, to name a few [3]. However, due to limited computational abilities, they suffer from security challenges. Recently, mobile operating systems are more common for personal and other uses like banking, office, and many other purposes. These devices are commonly known as smartphones. These devices are more generally used today in almost all platforms like social websites, banking, personal information keeping, and others [4]. When people try to access different personal information on their pocket phone, there is an issue discovered that each device is manufactured by other mobile companies like Apple, Samsung, Blackberry, to name a few. Each vendor and smartphone device uses a different operating system for its use. For instance, android operating systems used by Samsung, LG and HTC[5,6]. Windows operating systems are also optimized to use on various smartphones such as Nokia. Apple and Blackberry uses their own operating systems [7]. Rasberry PI, acting as IoT nodes in modern-day CPS uses a variant of android named AndroidThings. However, the optimization of size which is the focus of IoT devices leads to security challenges. For instance, a user using a smartphone, its device's video calls, GPS, Wi-Fi, and many more services are running almost all the time, and due to the internet's openness, the security of these devices is an issue. In the first quarter of 2018, a report issued from McAfee labs showed several malware attacks on smartphones that targeted the Android operating system [8]. When the number of attacks increases, there is a lot of effect on users' data and applications [9]. Different sophisticated security protocols are deployed but it has an associated overhead which at times degrade the performance of IoT nodes due their constraints abilities. Therefore, different sets of security protocols need to considered if the target devices are smart phones or IoT devices.

To this end, we investigate security models used by different operating systems in the context of smart serious games. We also discuss various techniques provided by the operating systems for ensuring the security of personal data and information. Each category of operating system has its security mechanism and thus reviewing the user's knowledge of recent smartphone operating systems' dangers, threats, vulnerabilities, and what exists is fundamental when choosing a smartphone for gameplay. In other words, the paper focuses on optimizing the existing security models in different operating systems to be used in smart serious games devices.

The rest of the paper provides: Section 2 provides the security models implemented in major mobile operating system. Section 3 defines the AI features in the context of Mobile OS. Section 4 describes the comparison based on the security vulnerability of these equipped security models. Finally, Section 5 provides a suitable operating system based on our findings from the study which can be a potential candidate for devices participating in smart serious games.

2. SECURITY MODELS DISCUSSION

2.1. Android Security Model

Since 2008, Android OS has been one of the most used operating systems around the globe. The reason behind this use is its application-free availability and ease to use interface. Despite it has been the leading OS in the market for terminal devices being used in games, according to some data sources from 2011, the security threats on these devices has increased significantly over time, leading to the amount of new malware/hackers [10, 11]. Android is an OS developed for smartphone us and can be used by different vendors. As depicted in Figure 1, the Android OS provides a boxed application and data execution environment. A well customized and groomed embedded Linux kernel system interacts with the smartphone hardware and other hardware equipment. The middleware includes Android RT, libraries, and VM, application's Application Programming Interface runs on top of Linux Kernel [12]. To simplify the running and execution

environment, an application's management interface is used via APIs. Each application executes itself under the supervision of a Dalvik Virtual Machine (DVM) that is running under the control

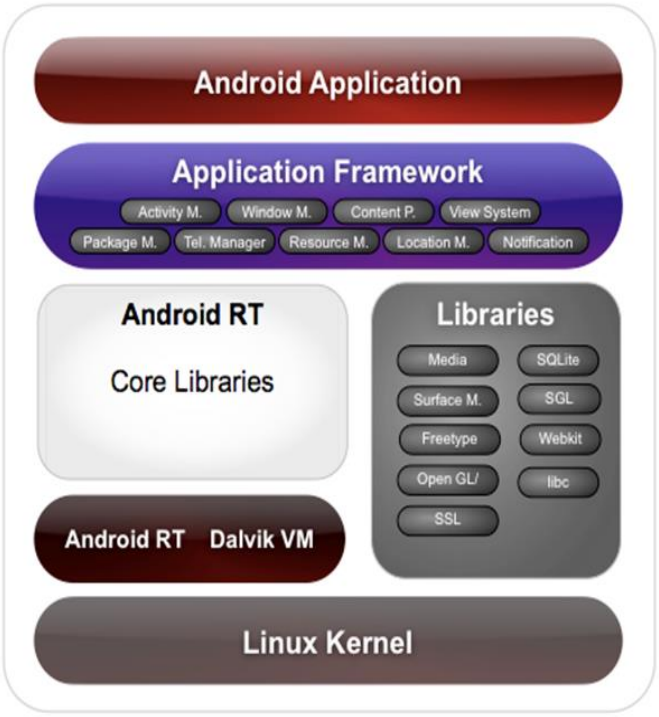


Figure 1. Android Application architecture [13]

of a unique UNIX user id (UID) [14]. The smartphone comes with a pre-installed applications, such as address book, call dialer, messenger for SMS and MMS, etc. When a user chooses the device based on the Android operating system for the applications and their data usage, the user does not know what type of attacks they can suffer while using this operating system platform. The user does not need to perform the security arrangements on these terminal devices.

The security is purely based on the operating system structure in which structure it is used for security permission. Android OS provides two different levels of security, i.e., application-level security and kernel-level security [15].

Application permissions in android OS only check and set the security and check license at the install time of the application after the install application on an android user does not change the application permission on these devices. App of the authorization are set at the install time of the application, so the user needs to keep a check on all of permission access levels at the install time when its application wants to check all of the applications needs to set the app permission when the user chooses the application at from the Android play store and wants to install that application on the device. Figure 2 shows some at from the Android play store and wants to install that application on the device. Figure 2 shows some of the access permissions on the Android OS [12-15]. Android OS provides the component level security that the whole application on the Android should be designed on these four components [16, 17].

First off, the activity component defines the application user interface; each screen of the android application defines an activity. One activity on the screen can cause the start-up of other events and for returning the fundamental values. One of the significant activities of the android

operating system is its phone dialer activity in which the user dials the call to other android devices. The service component performs the background processing. When the user interface disappears from the current interface, but the application is still running on the back, that activity is provided by the services component (like when users download a file). Services are the way of communication between different components and application interaction via some form of Remote Procure Call (RPC). Data is collected back with the help of this application and call-back services. The content provider provides the facility for database handling and stores and retrieves data by using the relational database. Every type of content provider has different authorities that describe which kind of content it takes during its execution. Some of the components need to handle the SQL queries on the database like (INSERT, UPDATE, and DELETE) for content reading and writing. Content provides stores all the activities in the database record.

The broadcast receiver component acts as the main box for sending and receiving messages from different applications, commonly in the Android OS application code broadcast messages to the intended receivers. The broadcast receivers thus subscribe to the procedures to receive the messages sent to them.

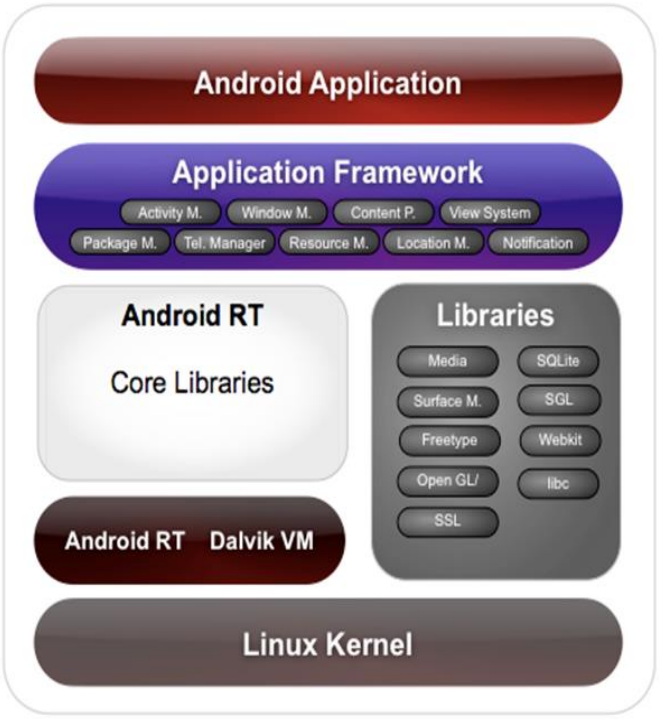


Figure 2. Android Application Permission [18]

2.2. iOS Security Model

Today iOS is also an effective and most widely used mobile operating system in the smartphone market. In the initial years of iOS, apple products were mainly used in Europe. Still, nowadays, it is also one of the widely used operating systems all around the globe, especially in Asia and Africa [19, 20]. If we talk about the security point of view, this mobile operating system provides security at the component and application levels. iOS offers one of the most robust security technology and features [21]. Apple iOS provides two types of security, both at the hardware and OS levels [22]. At the Low level of the operating system, firmware security features protect from viruses and malware attacks. High-level OS levels provide security of information at the app's

level access. That leads to preventing unauthorized use and helping thwart attacks provided on iOS.

In Figure 3, the iOS operating security model secures information while mobile usage is enabled, other installed apps from different sources, and synchronization. The complete design is based on the international standard procedure—and an apple a lot of time for enhancing the security of its products and improving the plan without any effect on the security precautions. Apple strongly focuses on having done additional design work to improve its safety. The iOS security model consists of the following architectural design [23-27]

System Architecture apple iOS is one of the best operating systems in terms of the boot loading process. Each step in the booting ensures the system is trusted cryptographically and signed by Apple itself. When the iOS is turned on the device, the application processor on the device loads the executable code from read-only memory called Boot ROM. The Boot ROM code has Apple Root CA public key, and this public key is used to authenticate Boot loader at the low level (LLB). Apple iOS assign this boot loader. It is the first step for the security measure by Apple iOS.

When the Boot Loader for the low-level completes its assigned task, this loader has the built-in functionality to load the next stage of the iOS booting process, iBoot, which then loads the final set of the booting process book kernel. This whole boot process allows iOS to load on only validated apple devices.

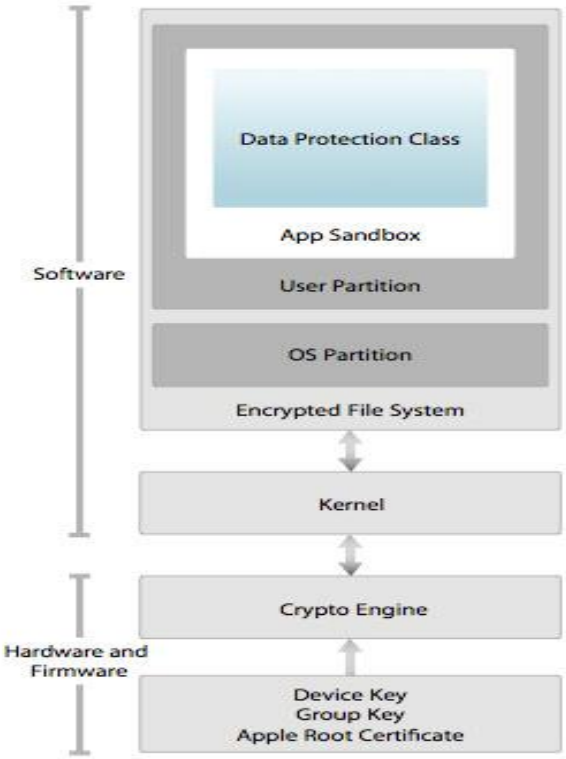


Figure 3. The security architecture of iOS [25]

Once the apple kernel is loaded, its next responsibility is to start and control all the user iOS processes and applications. Ensure that all of the apps need to be approved by the Apple iOS-

issued certificate [26]. Applications from the other sources must also be run and started by using the apple issue certificate.

Encryption and Data Protection On iOS, the encryption and protection facilities are based on hardware and software technologies. Every iOS has a path between the phone's main memory and the ROM-based storage with the help of a crypto-based key called AES 256 based key to achieve a high level of encryption and security [27]. This SHA-1 cryptographic algorithm is implemented at the hardware level to improve safety for implementing the security overhead. When the Apple mobile processor is manufactured, this 256-bit key cryptographic algorithm is fed unique ID and group ID in this hardware device. There is a unique ID and a standard group ID for all the Apple products used in the market on every Apple device.

Data protection is also an essential consideration in iOS, where all the user data is secure using a crypto key technology that is only used in the iOS platform. Whenever iOS need to secure users' data, it assigns a class to each segment. These class keys ensure when data needs to be accessed, and the key will unlock when information is accessed. When a new fine is created for the data protection, this data protection makes a unique key 256-bit in length; this key is based on a per-hardware basis and then forwarded to the AES engine. This AES engine uses a key to encrypt the file for storing it on the flash memory by sing CBC coding scheme [28]. When the file is encrypted with the SHA-1 encryption scheme, an output file is created that is Initialization Vector (IV) and stored in the shift register. iOS device passcode is one of the data protection mechanisms and for accessing the device. Every iOS device is equipped with a four-digit passcode that the user device needs to enable on its mobile platform. The whole iOS device is protected when the passcode is enabled on the device. This passcode interacts with the UID. The start screen passcode security is to make the iOS much more secure, and when a user attempts a passcode, then every time makes it safe. If a user attempts all the password combinations, then it will take approximately 80 for each attempt [29]. Using this speed, users need to try all the passcode combinations for about five and a half years. There is a six-character passcode that will take, and for the nine characters, it will take two and a half years. If the user attempts all these combinations, they will make it more secure because no one can try all the combinations. So, it is tough to break the security of the iOS passcode with encryption mechanism.

Network security including all other services iOS also provides network security services to its users. iOS provides authenticated, authorized, and encrypted communication between different devices over Wi-Fi and cellular internet services. iOS is less target for network attacks because of the limitations of ports and unavailability of the network firewall and other network utilities such as web servers, telnet, and shells. iOS uses its built-in encrypted apps for communication and other entertainment activities because these applications are very encrypted and authenticated by iOS users. These communication apps include FaceTime, iMessage, and iTunes. All the iOS devices are equipped with SSL and TSL layers for encryption of their data. iOS operating system supports three types of security measures [25]; Transport layer security, Secure socket layer and STLS. On iOS, all its internet-related applications like mail cylinder internet services and Safari use these types of encrypted communication mechanisms to ensure security between devices and the network.

Device access, one of another iOS security parameters is device access, and every smartphone has its optionality to active passcode to access the User Interface (UI) of that particular device. iOS is built-in with a four-digit passcode to access its first UI interface. This four-digit passcode is highly encrypted, and hard to guess such type of encrypted passcodes. Users can also set long alphanumeric passcode from 4 digits to longer passcode because this longer passcode is hard to guess. Every iOS device has a configuration file called XML. This XML file is based on all system configuration settings like passcode policies, Wi-Fi settings, VPS settings, e-mail settings,

exchange settings, web clips, credentials and keys, and a lot more. If a user comes and deletes this configuration XML file, all system settings will be removed and need to re-establish all locations.

2.3. Blackberry RIM (Research in Motion) OS Security Model

One of the Operating systems for mobile platforms is Blackberry. This blackberry operating system is specially designed for the enterprise's use, and individual users in terms of confidentiality, integrity authenticity Blackberry operating system is developed by Blackberry Inc and called it Blackberry RIM (Research in Motion) for their Blackberry devices. It is not a multi-platform operating system; it is only installed on Blackberry phones and smartphones [30].

Blackberry operating system is an ultimately end-to-end platform for mobile devices [31]. Here we now discuss the security model used by the Blackberry operating system. It encompasses the following subcategories; protecting data, protecting work data on personal use devices, enforcing strong access control and managing devices

Protecting data: The fundamental component of the blackberry data transection is its security for both enterprise and individual use. Blackberry provides a built-in data encrypting mechanism that provides robust data security policies. It also protects and manages the applications and devices from end-to-end security of data. Fig 4 shows the blackberry enterprise security and application access from the blackberry enterprise servers. This US-based company protects against all types of illegal entry. Data using Blackberry is strongly encrypted, and corresponding encrypted key is also used within these [32].

Messages and e-mail encryption are also handled by the Operating system using industry-standard S/MIME encryption. This operating system uses the concept of tunnelling in which multiple encryptions are done during the transmission over the network, which leads to achieving a higher level of security for these devices [33]

Enabling Work Data on Personal Use devices

Protecting users' data is the critical and comprehensive part of mobile devices, but the device's operating system only provides this protection. Blackberry allows the user to establish a quick setup for password creation to secure the data from illegal access. In Blackberry OS, the Blackberry Balance is a registered application that provides the control employee's data, companies data security, control network connection, and many more features that this application contains to process [34].

Enforcing Strong Access Control

Blackberry OS provides multiple control access features like authentication, anti-counterfeiting manufacturing, and device protection. Authentication is also taken place in blackberry devices for minimum chances for data loss [35].

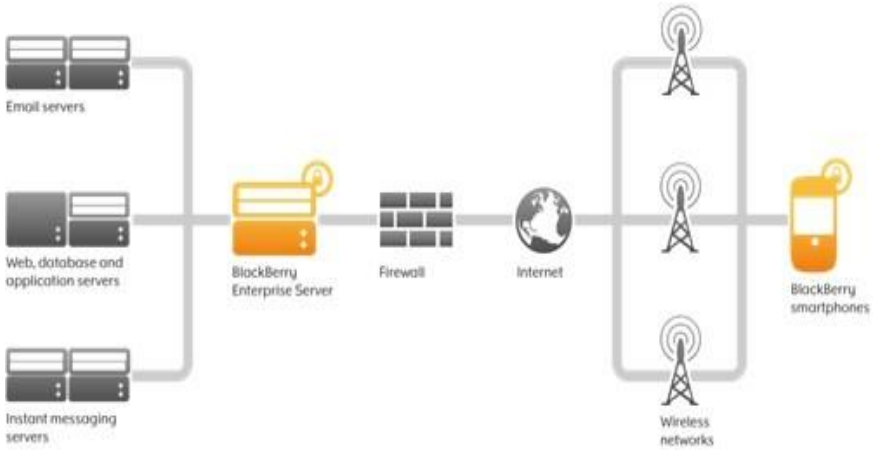


Figure 4. Blackberry enterprise security [36]

This authentication is key based, where all of the keys are kept private and secure for data protection. Fig. 5 shows the Blackberry 10 operating system application and data access mechanism. Whenever a blackberry device is activated, this device has a built-in tool that is first generated and key. It then sends this key to the operating system's server for creating and authenticating by the OS Server. Then this server sends back a certificate for authenticating the client device. Blackberry uses the management root certificate for authenticating the certificate for accessing the web services and their contents.

Managing Devices: Blackberry operating system is also one of the devices that can manage other devices like iOS, Android, and Windows phones. It has a built-in administration console to handle such a type of system. App of the managed application and data is secure from all the data and personal information applications. A trusted blackberry security model provides built-in security for blackberry devices apps developed to secure their private Virtual Private Network (VPN) [37]. Blackberry mobile device management is software used to connect and provide data sharing in a security manager for other operating systems.

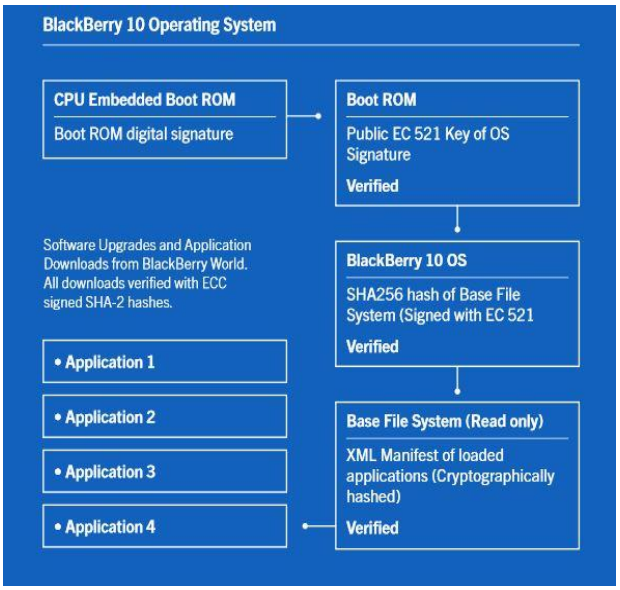


Figure 5. Application and data access mechanism in blackberry OS [38]

2.4. Windows Operating System Security

Windows operating system gained popularity from its desktop operating system, which provides a productive operating system for its users, and it is also now on the mobile platform [39]. At the start, Microsoft brought forward and developed windows seven operating systems for the Nokia smartphones. Later, they enhance user experience and get a new UI for their windows operating system, consisting of a 'Metro' based design. This new type of operating system supports the new kind of experience, which is multi-core processor support, high-resolution screen support, and also, most but not least, their more significant storage [40]. And the latest operating system is currently in use which is Windows 8.1 for mobile users [41].

The windows 8 phone consists of the following security measure taken by the windows 8 phone from a security perspective [42, 43].

Device Encryption: Windows phone provides complete internal protection against different attacks. Windows built-in BitLocker architecture gives full device encryption.

Data Encryption: Data encryption is another security measure in windows eight phones which is very important from the data point of view. When data is encrypted in the device, no other device can collect that data because it does not support it. It also helps to provide authentication between communicating parties.

Data Leak Prevention: Windows 8 phone has a built-in mechanism to prevent its data from leakage. So, Information Right Mechanism (IRM) is a way to avoid this type of information. IRM also protects e-mail and phone contents from illegal access from other devices and other applications on another phone.

Digital Signature: Windows 8 phones also provide security from other applications using a digital signature. It also helps to authenticate the information from another party.

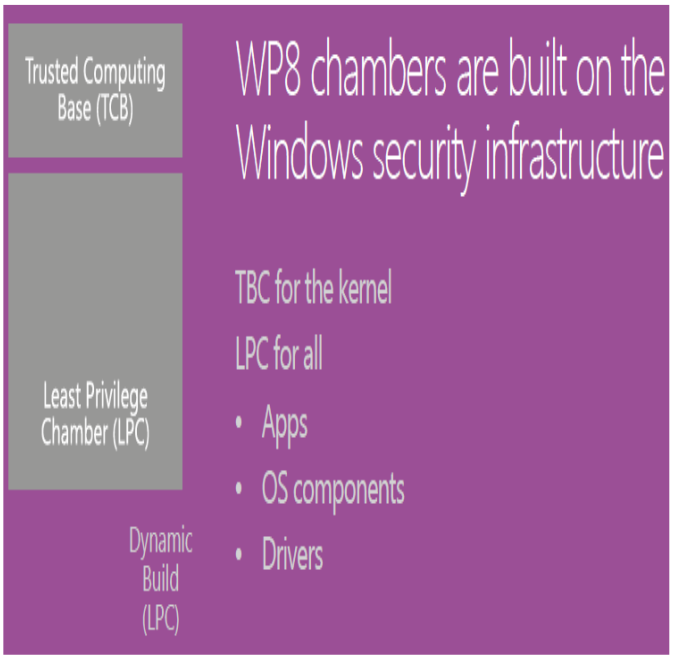


Figure 6. Windows 8 phone Application security model [41]

3. ARTIFICIAL INTELLIGENCE IN OS

Recent developments in AI and electronic technology raised the demand for these technologies in a modern OS. All current OS have AI in their User Interfaces, Human Machine Interaction features, Hardware Technology, Communication, Application support, etc. Machine to Machine Communication is possible through global services such as Internet services and other related platforms. In Modern OS, the AI component have become necessary partly because it controls the user actions, change working with the incomplete scenario, control user actions, and work with unsatisfied, incomplete, and noisy input values [43, 42]. All modern OS contains the user interface with a compelling voice as an AI feature to process and respond without typing a single word.

Besides all these, the advanced level of hardware technology also supports the next generation OS with AI features. The modern hardware contains new technology called AI on Chip (AI SOC). It includes AI sensors that effectively control the multi-user tasks with an effective computation environment [44]. Figure 7 contains such OS AI-enabled features.

AI controls user Interfaces to enhance the working of the OS for users and interactive user programming and other related facilities. Different OS platforms govern the inputs for practical reasoning and system analysis, and good communication level features. There are several reasonable devices which enhance the total working performances [45]. Figure 8 shows the AI features for the Human User interface for handling all such types of interfaces. In figure 8, CLI is Command Line Interface, GUI is Graphical User Interface, AUI is Audible User Interface, and UGI is User Gesture Interface.

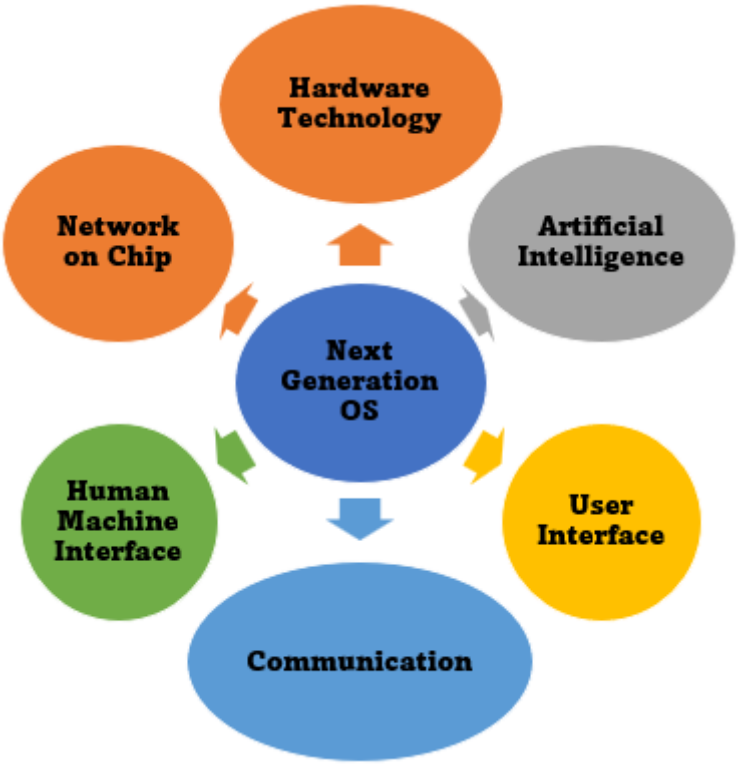


Figure 7. Operating Systems with AI features

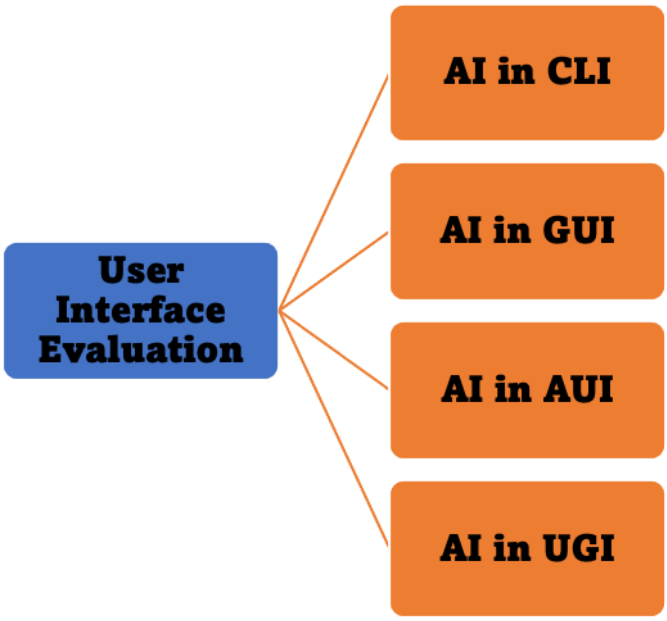


Figure 8. Evaluation of AI in OS User Interfaces

4. DISCUSSION

Every mobile operating system focused on Security, Data Protection, Authentication of users, Device Protections, Wipe of Device, Mobile Support, Concurrent E-mail management, Mobile Firewall Capabilities, Security Certifications, and Virtualization Security Certification based on descriptive technology. Every mobile device manufacturer wants appropriate mobile operating systems for their mobile devices. Figure 7 shows the usage statistics of mobile phone operating system users around the world. This gives us a clear picture that active mobile users using mobile operating systems. Based on the analysis found in the study, we announced that the most significant number of users using Android. But based on the evaluation in figure 7, according to the security, the Blackberry mobile operating system is best suitable for users. On the second, the security of the iOS for Apple mobile devices takes the second number with the second number on the mobile device selection procedure. According to the report, the most number of attacks are found yet on android devices [46].

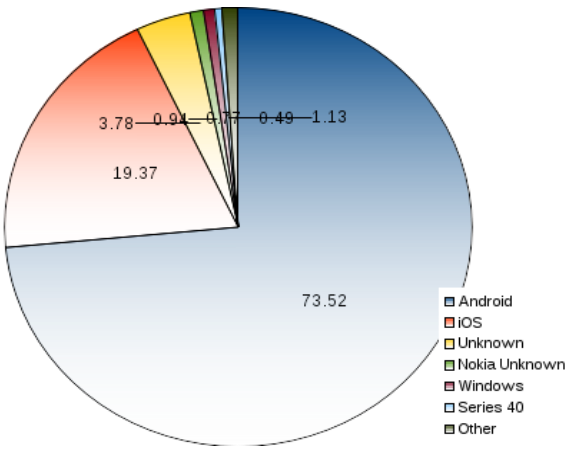


Figure 9. Mobile Operating Users Statistics

We have four primary Mobile Operating systems that have existed till now for most smartphones. Descriptive research is applied to the gathered data from multiple online and offline sources. Based on the analysis and data collected from different sources, we conclude that every type of mobile operating system has its own security mechanism for its users to secure their data from different malware and attacks either from the internal system or from the external system.

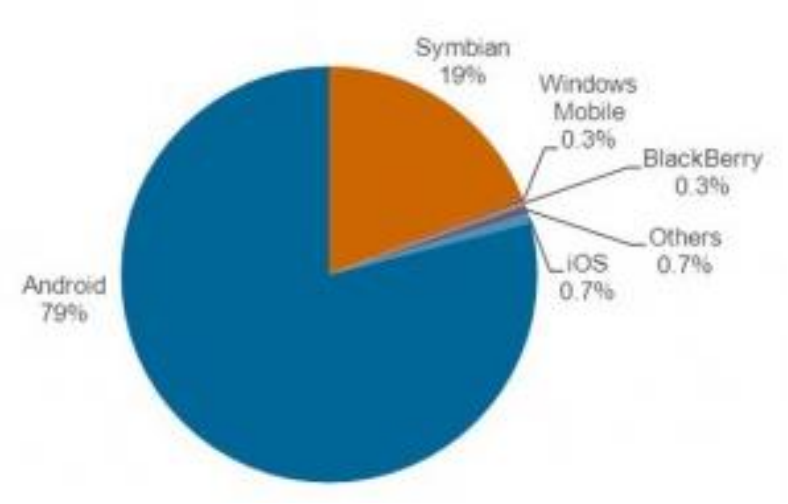


Figure 10. Mobile Phone Operating System attacks

Figure 7 describes the results based on our research. Based on our results, as Android is a free open-source mobile operating system, most of the data is handled by different programmers, changing its security. Apple iOS is, on the other hand, is another OS for mobile users, and it provides better security measures than Android because it is not open source. It offers better security as internal data, applications, as well as online security measures. Few other operating systems give other security measures and provide a better user experience as Blackberry uses its type of security measures. We must assign parametric values to the data gathered. Figure 8 shows a complete descriptive research result that shows the security according to multiple parameters listed below.

We compare our comparison based on the following security measures

1. Built-in security
2. Authentication
3. Data Protection
4. Device protection
5. Application-level security
6. Device Wipe
7. Mobile device management
8. Cooperate managed e-mail
9. Support for active sync
10. Device firewall
11. Security certifications
12. Virtualizations

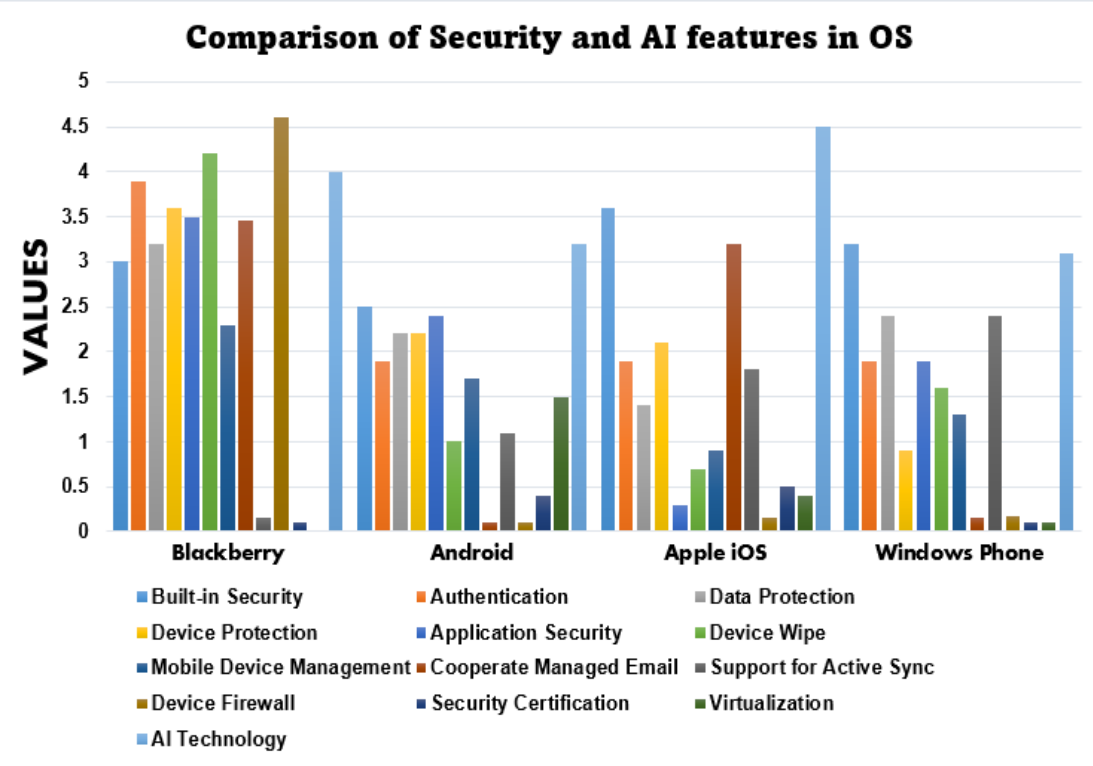


Figure 11. Cybernetic Comparison of Security Tools

5. CONCLUSIONS

Mobile Devices are remarkable inventions with more advanced Operating Systems with the most outstanding features for users to use in their daily and work purposes. Multiple applications on mobile phones extend the functionality of mobile phones. Technological advancements bring more elements to the smartphone with the penalty of features. This paper focuses on major vital points that are very helpful for the organization to adapt to the secure and well-groomed operating system from all of the operating systems in the market. Users of the operating system choose the best features of mobile devices with highlighted parts. All Operating systems of mobile devices effectively manage the processing power, unit of processing (processor), data storage and processing capabilities, network selection, and internet resource allocations. In support of security, the AI also enables multiple intelligence features in Mobile Operating Systems. AI controls the working of Mobile OS with its numerous features to get practical and under control. Users have some priority features when choosing the operating system from all operating systems in their interest. It's like an easy interface (user-friendly), secure, professional motions, multiple functionalities, official activities, entertainment, financial research, news. The priority of the OS design is to provide faculty to the users, so it controls and AI coordinates the user values and provides adequate user interface coordination. User priority is significant for choosing the best operating system for their personal use and their business purposes..

ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (grant number: 2021R1I1A1A01045177)

REFERENCES

[1] Ahmad, S., Khan, F., & Whangbo, T. K. (2022). Performance Evaluation of Topological Infrastructure in Internet-of-Things-Enabled Serious Games.

[2] Ahmad, Shabir, Faisal Mehmood, Faheem Khan, and Taeg K. Whangbo. 2022. "Architecting Intelligent Smart Serious Games for Healthcare Applications: A Technical Perspective" Sensors 22, no. 3: 810. https://doi.org/10.3390/s22030810

[3] Ahmad, Shabir, et al. "An Adaptive approach based on resource-awareness towards power-efficient real-time periodic task modeling on embedded IoT devices." Processes 6.7 (2018): 90.

[4] Lipp, M., et al. Armageddon: Cache attacks on mobile devices. in 25th {USENIX} Security Symposium ({USENIX} Security 16). 2016.

[5] Bhushan, A., A Comparative Analysis of Consumer Behavior of Nokia and Samsung Mobile Users. International Journal of Industrial Organization, Forthcoming, 2016.

[6] Schlagkamp, S., et al., Understanding user behavior: from HPC to HTC. Procedia Computer Science, 2016. 80: p. 2241-2245.

[7] Riadi, I., R. Umar, and A. Firdonsyah, Identification Of Digital Evidence On Android's Blackberry Messenger Using NIST Mobile Forensic Method. International Journal of Computer Science and Information Security (IJCSIS), 2017. 15(5): p. 155-160.

[8] McAfee, L., McAfee Labs 2018 Threats Predictions. Mission College Boulevard, Santa Clara, CA, 2017.

[9] Zhou, Y. and X. Jiang. Dissecting android malware: Characterization and evolution. in 2012 IEEE symposium on security and privacy. 2012. IEEE.

[10] Enck, W., M. Ongtang, and P. McDaniel, Understanding android security. IEEE security & privacy, 2009. 7(1): p. 50-57.

[11] Bhat, P. and K. Dutta, A survey on various threats and current state of security in android platform. ACM Computing Surveys (CSUR), 2019. 52(1): p. 1-35.

[12] Zhauniarovich, Y., Android Security (and Not) Internals. Trento: ASANI, 2014.

[13] Shukla, H., A Survey Paper on Android Operating System. Journal of the Gujarat Research Society, 2019. 21(5): p. 299-305.

[14] Oliver, E., A survey of platforms for mobile networks research. ACM SIGMOBILE Mobile Computing and Communications Review, 2009. 12(4): p. 56-63.

[15] Palmieri, M., I. Singh, and A. Cicchetti. Comparison of cross-platform mobile development tools. in 2012 16th International Conference on Intelligence in Next Generation Networks. 2012. IEEE.

[16] Sbîrlea, D., et al., Automatic detection of inter-application permission leaks in Android applications. IBM Journal of Research and Development, 2013. 57(6): p. 10: 1-10: 12.

[17] Felt, A.P., et al. Android permissions demystified. in Proceedings of the 18th ACM conference on Computer and communications security. 2011.

[18] Zarni Aung, W.Z., Permission-based android malware detection. International Journal of Scientific & Technology Research, 2013. 2(3): p. 228-234.

[19] Wang, G. Designing smule's iphone ocarina. in Proceedings of the International Conference on New Interfaces for Musical Expression. Pittsburgh. 2009.

[20] McMillan, D., et al. Further into the wild: Running worldwide trials of mobile systems. in International Conference on Pervasive Computing. 2010. Springer.

[21] Giese, D., et al., Security Analysis of Near-Field Communication (NFC) Payments. arXiv preprint arXiv:1904.10623, 2019.

[22] Philip, J. and M. Raju, An Overview About the Security Architecture of the Mobile Operating System iOS. Indian Journal of Computer Science, 2019. 4(1): p. 13-18.

[23] D'Orazio, C.J., et al., A Markov adversary model to detect vulnerable iOS devices and vulnerabilities in iOS apps. Applied Mathematics and Computation, 2017. 293: p. 523-544.

[24] Bhardwaj, A., K. Pandey, and R. Chopra, Android and iOS Security-An Analysis and Comparison Report. Int'l J. Info. Sec. & Cybercrime, 2016. 5: p. 30.

[25] Zlatolas, L.N., et al., Models of Privacy and Security Issues on Mobile Applications, in Mobile Platforms, Design, and Apps for Social Commerce. 2017, IGI Global. p. 84-105.

[26] Leavitt, N., Mobile security: finally a serious problem? Computer, 2011. 44(6): p. 11-14.

[27] Yang, W., et al. Vulnerability analysis of iPhone 6. in 2016 14th Annual Conference on Privacy, Security and Trust (PST). 2016. IEEE.

[28] Lauer, I. and T. Lauer, Undoing encryption: the argumentative function of metonyms. Argumentation and Advocacy, 2018. 54(1-2): p. 53-71.
[29] Saicheur, V. and K. Piromsopa. An implementation of AES-128 and AES-512 on Apple mobile processor. in 2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). 2017. IEEE.
[30] Gao, F., L. Leng, and J. Zeng. Palmprint recognition system with double-assistant-point on iOS mobile devices. in Proceedings of the 29th British Machine Vision Conference, BMVC. 2018.
[31] Goldsmith, M.A., et al., Mobile communications device providing heuristic security authentication features and related methods. 2019, Google Patents.
[32] Eisner, A., et al., BLACKBERRY LIMITED: IS THERE A PATH TO RECOVERY? Global Journal of Business Pedagogy Volume, 2018. 2(1).
[33] Philippe, E. and L. Bes, Method for manufacturing a multilayer data medium with security marking which can be marked by laser. 2019, Google Patents.
[34] Bender, C.L., D.J. Major, and J.R. Cardy, Data source based application sandboxing. 2019, Google Patents.
[35] Zaborski, A.A. and A.Y.W. Tam, Mobile phone shell. 2016, Google Patents.
[36] Novac, O.C., et al. Comparative study of Google Android, Apple iOS and Microsoft Windows phone mobile operating systems. in 2017 14th International Conference on Engineering of Modern Electric Systems (EMES). 2017. IEEE.
[37] Yesilyurt, M. and Y. Yalman, Security threats on mobile devices and their effects: estimations for the future. International Journal of Security and Its Applications, 2016. 10(2): p. 13-26.
[38] Joseph, J. and K. Shinto Kurian, Mobile OS–Comparative Study. Journal of Engineering Computers & Applied Sciences, 2013. 2(10): p. 10-19.
[39] Pawłowicz, B., A. Strzałka, and M. Tybura, Privacy and security of contact data on mobile phones with Windows Phone Operating System. Measurement Automation Monitoring, 2017. 63.
[40] Jaafar, A., et al. Dynamic home automation security (DyHAS) alert system with laser interfaces on webpages and windows mobile using rasberry PI. in 2016 7th IEEE Control and System Graduate Research Colloquium (ICSGRC). 2016. IEEE.
[41] Zaidi, S.F.A., et al., A survey on security for smartphone device. International journal of advanced computer science and applications, 2016. 7(4): p. 206-219.
[42] Waheed, A. and H.A.F. Khan. Artificial Intelligence in Operating System. in Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence. 2019.
[43] Tonogai, D., AI in Operating Systems: An Expert Scheduler. 1988: University of California.
[44] Thórisson, K.R.J.M. and Machines, Integrated AI systems. 2007. 17(1): p. 11-25.
[45] Malallah, H., et al., A comprehensive study of kernel (issues and concepts) in different operating systems. 2021: p. 16-31.
[46] Omelchenko, T., et al. Protection Software for Mobile Operating Systems. in 2018 International Conference on System Modeling & Advancement in Research Trends (SMART). 2018. IEEE.

AUTHORS

ABID ALI is pursuing his Ph.D. degree in Computer Science at the Department of Computer Science, The University of Engineering and Technology Taxila Pakistan. He did his MS (CS) from the same institution in 2018. He is currently serving as a Lecturer in Computer Science in Higher Education Department KP Pakistan. He has 8 years of teaching and 5 years of research experience. His Currently Research interests are IoT, Distributed Computing, Big Data, Task Scheduling, Data mining, Cloud and Mobile Cloud Computing ICN, SDN, and VANET.

Faisal Jamil has secured his Ph.D. in the Department of Computer Engineering from Jeju National University, the Republic of Korea. He received his MS in Computer Science from University of Engineering and Technology, Taxila Pakistan in 2018. He did his BS in Computer Science from the Capital University of Science. His research work mainly focused on Internet of Things application, Blockchain application, energy optimization and prediction intelligent service, and mobile computing. He is currently serving as a postdoctoral researcher with Gachon University at the department of Computer Engineering.

Shabir Ahmad has earned his PhD in Computer Engineering from Jeju National University, the Republic of Korea. He received his Master's degree in Computer Software Engineering from the National University of Science and Technology, Islamabad, Pakistan, in 2013. He did his Bachelor of Science in Computer Systems Engineering from the University of Engineering and Technology, Peshawar, Pakistan, and is now serving in the same university as a faculty member of the Software Engineering Department. His research work mainly focused on Internet of Things applications, cyber-physical systems and Intelligent systems.

AN AUTOMATED VIDEO CONTENT CUSTOMIZATION SYSTEM USING EYE TRACKING AND ARTIFICIAL INTELLIGENCE

Yuyang Lou[1] and Yu Sun[2]

[1]Charles Wright Academy, 7723 Chambers Creek Rd W, Tacoma, WA 98467
[2]California State Polytechnic University, Pomona,
CA, 91768, Irvine, CA 92620

ABSTRACT

In the past few years, the internet and online social networks developed drastically, promoting the development of online learning programs. These programs provided opportunities for a digital learning experience that allows students to explore beyond what's taught in school. However, having a clear understanding of what topic might interest the user and motivate the user to further explore that topic is hard for both the user and the learning program. This paper proposes to create one centralized method of predicting what the user would be interested in and provide them with educational content recommendations. Our design builds upon the eye-tracking techniques, which allows us to capture users' eye movements, and object recognition achieved by machine learning, which allows us to examine the specific object that the users are looking at and provide data for the users' interest analysis [3]. Our results show a success rate of 90% of analyzing what the user is truly looking at. We used our decision heuristic, etc.

KEYWORDS

Eye Tracking, Deep learning, computer vision.

1. INTRODUCTION

Moving into the next decade, individual interest and the inclusion of various communities became increasingly important when companies are designing products [1][2]. Content recommendation based on the analysis of these individual interests, as a result, grows more and more popular in many fields [3]. In the industry of entertainment, various companies use content recommendations to attract their users and keep them entertained [6]. In the industry of online shopping, content recommendations are utilized to help the customers discover their favorite item and increase the likelihood that an item is going to be purchased. In the advertisement industry, content recommendation has become one of the most important aspects to attract users to their potential interests. However, in the field of education, the potential of content recommendation has not yet been fully developed. While a person might not be fully aware of what they are interested in the most during their learning process, wandering on learning platforms aimlessly and browsing content from various topics is very time-consuming and not helpful in determining their path of interest. Analyzing their interest throughout their learning process by tracking their behaviors and recommending them on more topics based on the analysis, can direct the users to their interest and speed up the exploration process drastically. This project, therefore, attempts to build a content recommendation system based on analysis of users' interests, aiming to provide a

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 39-48, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120403

learning experience that is intriguing to the users and allows them to further study and exceed on topics they are truly passionate about.

Before the existence of the internet, advertisements, commercials on televisions and radios, and billboards were the only access companies had to market their products [7]. To improve its efficiency while bound by the technological limitations at the time, people started to deliver those advertisements by geographical interests, the first form of a content recommendation system based on users' interests. Several decades later, some of the most popular methods of creating such a system focus on the analysis of users' social networks and the construction of an evaluation system that categorizes different products for different needs [8]. These methods are heavily implemented in real life, making impacts daily. Entertainment platforms such as YouTube and Netflix rely heavily on their evaluation system that categorizes movies and videos into different genres, which in turn allows them to recommend to users similar products that they have been previously enjoying. On the other hand, social media such as Twitter and Instagram, while also implementing an evaluation system, also analyzes its users' social network information. By portraying a social network graph, social media can capture users' connections with each other, similar interests among a group of users. However, these techniques are never used alone. Companies such as amazon utilize a combination of the two, recommending users similar products they have previously bought based on their evaluation metrics of the products, and products favored by similar users through their analysis of the user's social network graph. These techniques are relatively cheaper and practical yet consist of disadvantages too. Programs utilizing these techniques, at best, can only give estimations of a user's potential interests, because the data it is processing are all secondary data that does not directly represent users' true intention, and sometimes waste time and effort marketing the wrong product to the wrong group [9].

In this paper, we follow the same line of research to construct an evaluation metric that can help us categorize our products and better deliver them to our users of similar interests. However, the process of determining the users' potential interest is different from the existing ways elaborated in the former paragraph. Here, we took a more direct approach, extracting information regarding the users' eye movement utilizing eye-tracking equipment, and a video that the users are watching [4]. We then process the video through a machine learning model, recognize the objects occurring in the video, and analyze the objects that the users' eyes are focusing on, in turn predicting the users' potential interest. As human eye movement is a way to express their emotion, this method is a more direct way of predicting users' potential interest and delivering them similar products. In the future, there are many ways in which this concept can be put into practical use and benefit our community. In the field of education, this technique can help children develop their interest and explore their topics of interest at a young age and help them learn about vocabularies by staring at different pictures corresponding to different words.

In two application scenarios, we demonstrate the practicality and accuracy of the model in terms of predicting the user's interest after watching a video. In the first experiment, we tested our model after adjusting the parametric to analyze its performance at recognizing objects in the video. We make amends to the number of batches the model is trained on, the number of epochs the model will iterate through, and train the model with a dataset with different sizes [10]. Its accuracy is shown by the graphs and data collected after the experiment. In this case, the model yielded a result of over 80% of accuracy and around 90% of precision. In the second experiment, we tested the practicability of the model when analyzing the users' interest through the data collected when they are watching the video. A questionnaire to the users is performed after the model makes its prediction, to reflect on the percentage of error between users' intended interests and the prediction. By linking the program to the Google search engine, the program reached a considerable amount of accuracy at predicting users' interest and recommending them further

content of interest. The model indeed exhibits a percentage of error that is not negligible, and future adjustment of the evaluation metric will likely solve this problem. Through the above two experiments, we can show the accuracy and practicality of the program, given its performance on object recognition and user interest analysis. We are also aware of the necessary future improvement to withstand every scenario.

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 concluding remarks, as well as points out the future work of this project.

2. CHALLENGES

To build the project, a few challenges have been identified as follows.

2.1. Implementing/combining the YoloV5 established model into the rest of our codebase can often lead to issues

Integrating other pre-developed programs into our code can be both very time-consuming and troublesome. The YoloV5 libraries that we are using have pre-built programs that perform object detection. To extract information for our use in the eye-tracking system, it is necessary to make adaptations to both systems, which requires a lot of work [5]. Generally, the solution is to write the program around the libraries that we intended to integrate into our system. In our case, however, the eye-tracking program that we built is a new system. Our solution to this problem then is to scan through the YoloV5 library, interpret the functions of its code, and make changes to its scripts to integrate it into our eye-tracking system.

2.2. Designing a content viewing tracking system with the mouse and human eyes is tough to do without it being slow/hard-to-use

Building a real-time content viewing tracking system is very hard because object detection consumes a lot of time and that leads to a sub-optimal user experience. Real-time object detection is often too slow for proper viewing experiences due to constraints in hardware. If a real-time tracking system is implemented, frames of the videos will drop significantly, leading to negative user experiences. Generally, there are several solutions to the problem. On one hand, we can add more computational power, such as multiple processors, to the system to support the calculation needed for object detection or find better algorithms that improve the efficiency of the program. On the other hand, we can pre-process the video to not cause the slow-down from real-time object detection. Due to our limitation in hardware resources and the intended purpose of this program to be publicized, we decided to pre-process the videos and implement a real-time eye-tracking system, so that the user is ensured fluency in their experience without requiring too much computational power.

2.3. Designing a decision heuristic network requires a lot of training and adaptations to accurately analyze the user's interests

Due to the inconsistency of Users' behavior, decision heuristic networks will need a lot of data to adapt to various conditions. In our real-time eye-tracking system, data regarding users' eye movement must be captured in real-time and analyzed. It imposed a challenge for us to grasp the

data and analyze it in a way that is not too time-consuming to affect user experiences. Generally, to solve the problem the data of users' eye movement are recorded and analyzed at the end of the video, so the analysis will not affect the user experience. In addition to the collection of the data and an overall analysis of the data at the end of the video, we built a user interface using Tkinter that allows us to create a display right next to the video, displaying the result when the program performs real-time interest analysis.

3. SOLUTION

The system is a user interest analysis system based on object recognition and eye-tracking techniques. The system integrates the yolov5 machine learning model to perform object detection on the videos and output the video's corresponding data. The model is trained on various topics of interest, to perform accurate object detection on videos regarding those topics. This object detection is then pre-run before creating the system to decrease real-time object recognition delays. When the users begin to interact with the video and give input with their eyes or the mouse, the system then retrieves the corresponding datasets of the video acquired during the pre-running process and monitors the users' eye or mouse movements. While the user is watching the video, the system will be responsive to the user's eye movement/mouse movements, the objects they are currently looking at by referring to the corresponding datasets and giving real-time recommendations to articles and web links based on the objects. In addition to the real-time recommendations, the system also records users' eye or mouse movements and performs overall analysis at the end of the video, to give more accurate suggestions based on a heuristic network. The system also provides a modern user interface and convenient function for users to manipulate the system and grant them a great user experience (See Figure 1 for the system's overall structure).

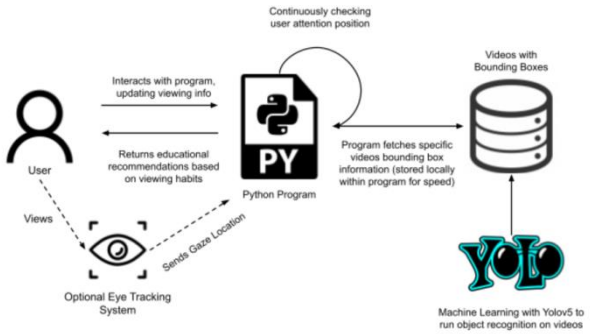


Figure 1. An outline of the Interest Analysis System

To train the model, we searched for videos of different topics of interest such as biology, astronomy, rocket science, and extracted frames out of those videos as our training set using OpenCV libraries. We also added additional distorted versions of the images to create a stronger dataset using a tool called Roboflow. We then hand-labeled the images, generated a yolov5 file, and trained the model with it. To output the video's data during the pre-running process, which contains the bounding box information of the objects, we made changes in the yolov5 file and extracted that information to a separate text file. To achieve real-time interaction between the user and the system, we built a python application using Tkinter and OpenCV, which provided convenient library features to track the user's eye or mouse movements and integrated the Tobii eye-tracking system into the application, which allows us to access data regarding the users' eye movements. We also made adaptations in the yolov5 files so that every time the system detects a change in the users' eye or mouse movements, it refers back to the text file generated during the pre-running process, access the bounding box information, and determine the objects the users

are currently looking at. To achieve real-time recommendation, we integrated google search into the function so that when the program receives data of the objects that the users are looking at, it looks for information regarding the objects on google and shows five web links, based on their popularity, in the application. The links are integrated into a clickable label and the user will be able to access the web link instantly as they are looking at the object. To make this process more efficient, we also built a dictionary that keeps track of searched items, and their corresponding search results, and make sure that no repeated work has been done to interrupt the users' experience. The program also stores the data to the end of the video and performs more sophisticated analysis on the users' eye movement, in terms of the percentage of the time the users are looking at a specific object, or how many consecutive seconds the users are looking at a single object. The users will be able to access such data through a stats button. This information is also utilized by the recommendation system to produce a more personalized recommendation regarding the users' major interests. Finally, to make the UI modern and convenient to use, we designed the layout of the application, so the widgets are aligned with each other to bring comfort to the users aesthetically. The media player occupies most of the screen space in the application, and the real-time recommendations and the data appear on the right column of the application. There is also a list of videos the users can play at the bottom of the application. These features add to the simplicity of the application and make it easier to interact.

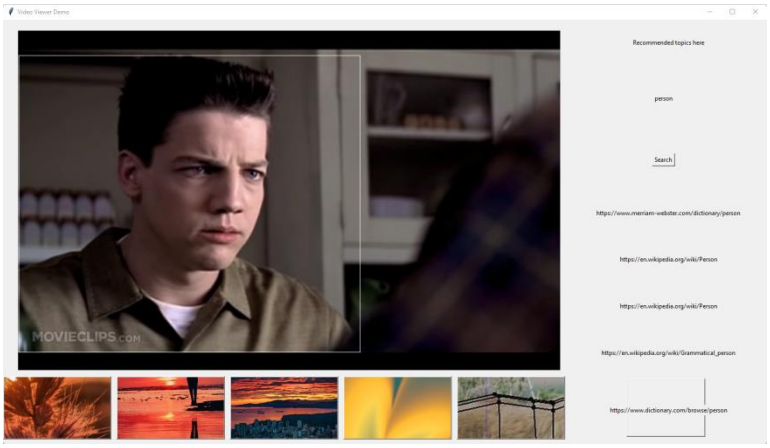


Figure 2. A screenshot of the application

```
def motion(self, event):

    x = event.x - 5
    y = event.y - 5
    for bbox in self.curr_bboxes:
        min_x = bbox[0]
        min_y = bbox[1]
        max_x = bbox[2]
        max_y = bbox[3]

        if x >= min_x and x <= max_x and y >= min_y and y <= max_y:
            print("Currently looking at {}".format(bbox[4]))
            if self.watch_timer.get(bbox[4]):
                self.watch_timer[bbox[4]] += 1
            else:
                self.watch_timer[bbox[4]] = 1
            if bbox[4] != self.mostrecentlylookedat:
                print(self.seen_words)
                if self.seen_words.get(bbox[4]):
                    self.search_label.configure(text=str(bbox[4]))
                    numlink = 5
                    links = []
                    eval_link = lambda x: (lambda p: webbrowser.open_new(x))
                    for i in range(numlink):
                        links.append(Label(self.root, text=self.seen_words[bbox[4]][i]))
                        links[i].grid(row=i + 3, column=5)
                        links[i].bind('<Button-1>', eval_link(self.seen_words[bbox[4]][i]))
                    print(links)

                else:
                    self.mostrecentlylookedat = bbox[4]
                    self.search_label.configure(text=str(bbox[4]))
                    data = str(self.mostrecentlylookedat)
                    results = search(data)
                    self.seen_words[bbox[4]] = results
                    numlink = 5
                    links = []
                    eval_link = lambda x: (lambda p: webbrowser.open_new(x))
                    for i in range(numlink):
                        links.append(Label(self.root, text=results[i]))
                        links[i].grid(row=i + 3, column=5)
                        links[i].bind('<Button-1>', eval_link(results[i]))
```

Figure 3. A segment of the code that allows the program to track users' eye and mouse movements

```
def videoLoop(self):
    try:
        timer = FPS().start()
        while not self.stopEvent.is_set() and not self.paused:


            self.grabed, self.frame = self.vs.read()

            if self.frame is None:
                break
            self.frame = imutils.resize(self.frame, width=1080)

            image = cv2.cvtColor(self.frame, cv2.COLOR_BGR2RGB)
            image = Image.fromarray(image)
            #Update the bounding boxes for said image
            self.update_curr_bboxes(self.framenumber)
            for bbox in self.curr_bboxes:
                min_x = bbox[0]
                min_y = bbox[1]
                max_x = bbox[2]
                max_y = bbox[3]
                draw = ImageDraw.Draw(image)
                draw.line((min_x, max_y, max_x, max_y))
                draw.line((min_x, min_y, max_x, min_y))
                draw.line((max_x, min_y, max_x, max_y))
                draw.line((min_x, min_y, min_x, max_y))

            image = ImageTk.PhotoImage(image)
            self.framenumber += 1
            if self.panel is None:
                self.panel = Label(image=image)
                self.panel.image = image
                self.panel.grid(column=0, row=0, columnspan=5, rowspan=7, padx=self.display_padx, pady=self.display_pady)

            else:
                self.panel.configure(image=image)
                self.panel.image = image
        timer.stop()
        print(timer.fps())
        print(timer.elapsed())
```

Figure 4. A segment of the code that streams the video

4. EXPERIMENT

4.1. Experiment 1

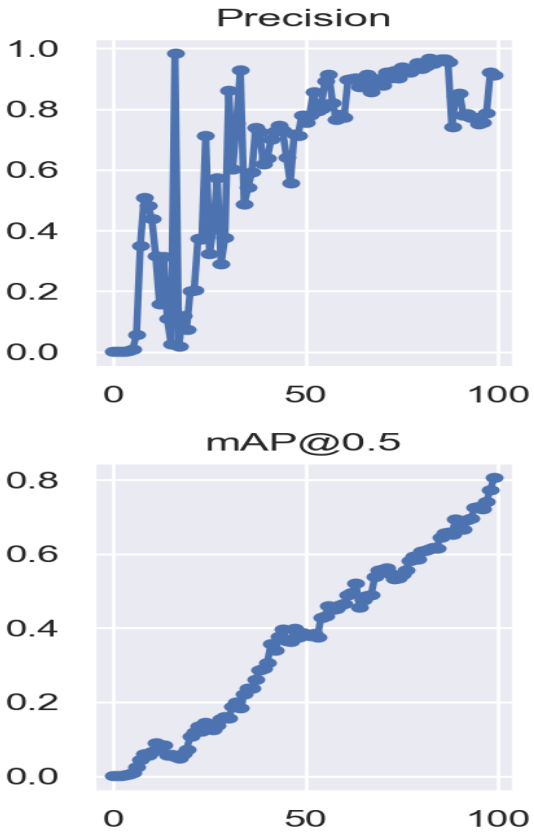


Figure 5. Result of the experiment

After adjusting the parameters multiple times, the results show that a relatively lower number of batches and a higher number of epochs will result in the most optimally trained model for our experiment. We initially started training the model on only 1 batch and 10 epochs, yielding an accuracy of at most 10%. When we tried to train the model with over 16 batches, the program crashed due to the limitations of our hardware. Eventually, we decided to increase the number of epochs and keep the number of batches around 16. As the number of epochs increased, accuracy and precision improved and when the number of epochs reached 100, the model yielded a result of over 80% of accuracy and around 90% of precision, which is a major improvement over our initial trial. The accuracy and precision of the model only showed slight improvement when the number of epochs increased over 100.

Finding training metrics that result in the most optimally trained model allows us to train our model with more unique objects with higher object recognition accuracy, eventually improving the overall accuracy at analyzing users' interests. We trained on the model with a specific dataset. We adjust the parametric regarding the number of epochs and batches we are training the model on and analyze the performance of the model when the training is complete, and tests are performed on the model. Tests are performed multiple times each time we adjust the parametric, to ensure that the outliner will have a lesser impact on the overall result.

4.2. Experiment 2

With higher accuracy at analyzing users' interests, the system will be able to produce more sophisticated and personalized recommendations, improving users' experience when interacting with the system and having a further understanding of their interests.

To measure the system's extent of success in analyzing the users' interests, we run the system and record how accurate the program is at tracking users' eye or mouse movements and recognizing the objects they are focused on. Analyzing the program's success at cultivating interest and the relevance between the system's recommendations and the users' real interests is done by user feedback.

When interacting with the system, the program shows major success at tracking the users' eye or mouse movement quickly and accurately, in turn allowing successful object recognition. The program's interpretation of the users' interests, however, wasn't as successful. At the end of each video stream, I usually found 40% of the suggestions that the program interpreted as my interests corresponding to my real interests, partly since the correlation between what the users are looking at and their real interests aren't always stable, and people's interests can be affected by measures other than their vision. The weblinks that the program provides also show low relevance between the links and the users' real interests, primarily because we are choosing the most popular web pages that have low correspondence to personal background.

Based on the two experiences we have done; we can prove that our solution has both stable performance and high accuracy.

5. RELATED WORK

In this article, the team utilizes SNS (social network service) data and eye-tracking data to create a preference metric, and yield user preferences for categories [11]. While this research leans more on the analysis of SNS data using eye-tracking techniques, my work focuses primarily on the eye-tracking data extracted directly from the media by detecting the objects the users are looking at using machine learning. Compared to their work, my research proposed a more direct approach toward user preference analysis but overlooks the effect of SNS on user preference analysis.

In this article, the team attempts to achieve accurate event detection, that is, the process of analyzing event streams to recognize the event types [12]. They implemented machine learning to train the classifiers and performed event detection based on the classifier. Compared to their work, my research approaches event detection by pre-training the media that the users are looking at, and yielding results based on their eye-movement data. My work is more efficient in terms of time, however, less flexible when performing live updates.

In this article, the team utilizes analysis on the shape and the color features of the objects to achieve object detection [13]. The team extracts the length=width ratio as the shape information of the object and extracts the color histogram to determine the color of the object. Eventually, a video surveillance system is implemented. While they focus on the color and shape of the objects to achieve object detection, my project focuses on the heaving training and the large dataset built into the system, which allows it to extract features itself and classify the object based on those features. My work is more accurate and more flexible, however, requires a lot of data training [14].

6. CONCLUSIONS

In this project, we proposed a way of analyzing users' interest while browning videos and articles, by constructing a program that can analyze users' eye movement on the screen and recognize the objects their eyes are focusing at through implementation of the Yolov5 machine learning model and the use of eye-tracking equipment. Through real-time recommendations and post-video analysis, we intend to provide the users with a comfortable and personal entertainment experience while exploring their field of interest. The program will also implement a modern user interface, so it is suitable for everyone. In this project, we designed two experiments to evaluate the success of our program. We tested its accuracy at recognizing objects in the video by training the model with different parametric, specifically the batches the model is running with and the number of epochs the model will be trained on. We also managed to measure the margin of error between the model's prediction of the user's interests and the user's real interests by performing a set of questionnaires to the users after the program delivered them recommendations. The result of these two experiments proved the effectiveness and the practicality of the program. The program was able to recognize objects at a precision of over 90% and an accuracy over 80%, it also predicts users' interests at a considerable accuracy, over 50% accuracy on average.

However, current limitations of the program exist and are not negligible in the future development of the program. Due to hardware limitations, the machine learning model is not able to train on many batches, which in turn undermines its accuracy at performing object recognition [15]. The program is also limited in its current prediction of user's interest, and with the percentage of error displayed from the experiments, The program will not exceed other measures to analyze users' interest. Currently, there still exists a gap between what the user's real intentions are when watching the video and what the program interprets the users' eye movement as an indicator of their preference.

To make sure that the application's hardware requirement is comfortable and affordable to the majority, we will tackle the technical limitations by controlling the quality of the training dataset and possibly replacing our current machine learning model to reach a higher accuracy when recognizing objects from the video. We will also begin to construct evaluation metrics using machine learning in the future to process users' eye movement data and give them more reasonable recommendations based on the processed data.

REFERENCES

[1] Mitchell, J. Clyde. "Social networks." Annual review of anthropology 3.1 (1974): 279-299.
[2] Twigg, Carol A. "Models for online learning." Educause review 38 (2003): 28-38.
[3] Tanenhaus, Michael K., and Michael J. Spivey-Knowlton. "Eye-tracking." Language and Cognitive processes 11.6 (1996): 583-588.
[4] Holmqvist, Kenneth, et al. Eye tracking: A comprehensive guide to methods and measures. OUP Oxford, 2011.
[5] Krafka, Kyle, et al. "Eye tracking for everyone." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[6] Vogel, Harold L. Entertainment industry economics: A guide for financial analysis. Cambridge University Press, 2020.
[7] Beaver, William H. "Market efficiency." Accounting Review (1981): 23-37.
[8] Arreola, Raoul Albert. Developing a comprehensive faculty evaluation system. Madison, WI: Magna Publications, 2004.
[9] Vartanian, Thomas P. Secondary data analysis. Oxford University Press, 2010.
[10] Epstein, Joshua M. "Why model?" Journal of artificial societies and social simulation 11.4 (2008): 12.
[11] Song, Hyejin, and Nammee Moon. "Eye-tracking and social behavior preference-based recommendation system." The Journal of Supercomputing 75.4 (2019): 1990-2006.

[12] Zemblys, Raimondas, et al. "Using machine learning to detect events in eye-tracking data." Behavior research methods 50.1 (2018): 160-181.
[13] Wu, Jun, and Zhitao Xiao. "Video surveillance object recognition based on shape and color features." 2010 3rd International Congress on Image and Signal Processing. Vol. 1. IEEE, 2010.
[14] Pazzani, Michael J., and Daniel Billsus. "Content-based recommendation systems." The adaptive web. Springer, Berlin, Heidelberg, 2007. 325-341.
[15] Zhang, Xian-Da. "A matrix algebra approach to artificial intelligence." (2020): 803.

AN INTELLIGENT SYSTEM TO ASSIST THE DRAWING COMPLETION AND COLORING SUGGESTION USING AI AND IMAGE PROCESSING TECHNIQUES

Jichuan Wang[1] and Yu Sun[2]

[1]Northwood High School, 4515 Portola PKWY, USA
[2]California State Polytechnic University,
Pomona, CA, 91768, Irvine, CA 92620, USA

ABSTRACT

Many times when we create any design or drawing on a digital platform, we face the problem that lines we created couldn't close up the shape correctly [1]. When we use the paint bucket tool, it fills the whole screen up with one click, and we need to re-look through the whole design again to find where it wasn't closed up correctly, not only taking up much of our time, but it's also very annoying. By determining a rule for whether or not a shape is completely closed or open, this application checks whether the shapes we created were closed by going through each pixel and detecting if there are holes around it [2]. The application could be provided for anyone creating 2d designs on a computer whether for hobby or job, to save the time looking through the whole design again and find the errors with much time are really tiny.

KEYWORDS

Shapes, checker, fixer, pixel extraction.

1. INTRODUCTION

As using any designing app that includes a tool to select or fill an area, the tool such as paint buckets often would take account of any tiny holes on the outline. This, in return, messes up the whole design by filling up or selecting not only the area wanted, but also the whole opening area connected to that tiny opening. When this occurs, you need to adjust it by hand, whether through erasing or finding and closing up the opening yourself, as it's more annoying to fill in the area using a pen tool which may draw on top or over the outlined shape [3]. And because of this, you often have to look through everything in the image to find the location of the problem, a process that could be even more difficult when the complexity of the design increases [4]. This is opposed when people have to look through the whole canvas and in the end find the opening at the smallest tip over, for example, the character's hair. Maybe over time people could get better at making designs to make no holes on the outline, but it's a common problem for most people who weren't trained for years, and they often thought they have completed their design, only to find that when filling in the desired color, the whole page is filled up with the color, and possibly covering other work they have already done. This can speed up the process of creating designs by letting people not put too much of their mind on finding problems in the closure of their design, and allow for more advanced novice design [5].

Some of the techniques and systems that have been proposed until now repeat and draw out existing shapes that allow the user to input their drawing into the program and draw the shape

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 49-58, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120404

out, providing a canvas that allows the user to edit [7]. However, this implementation was limited as the user could only see the shape they drew. Furthermore, various other software applications only assist the process further by giving the user the tools to manually fix the problems with the image. This doesn't actually help the user with the original problem they had of wanting to fix their shape, but instead just draws what the user gives and asks the user to fix and find the problems themselves [8]. Tools such as Onshape and Goodnotes provide these useful tools, but do not provide fixtures to incomplete shapes. They only support fully closed shapes.

In this paper, we follow the same idea that helps the user make their shape more complete for further steps in creating their art. In this case, we have decided to utilize pixel art, or art that is drawn using single pixels that connect to form a complete image, for the drawings that are checked in the program. Instead of providing tools for users to manually do the drawing themselves, we first check their image for pixels in the shape that do not connect to other pixels, forming an incomplete shape. We implement a steady algorithm to check eight different directions that adjacent pixels can be from any given pixel in the image. Second, we tell the user where the problems are located in their image. In other words, we describe which pixels are not connected to other pixels in the image that cause our shape to be incomplete. Third, we help them fix their problem by filling in any gaps that they missed in their pixel drawing. We implement a similar algorithm to the one that determines completeness or incompleteness of a shape; however, we use this algorithm to now determine a start point and an end point to which a line will connect the two incomplete pixels from.

We used the graphing calculator to plot out all the coordinates that our program went through that determined the shape's pixels in a given image. With this graph, we saw what pixels were analyzed and how they were interpreted, and how closely it was to our predicted result. In terms of how the pixels were interpreted, we were using the graph to see if the program could extract all the colored pixels that we were supposed to analyze (this could be easily proven to be correct with closed images, since any gaps would prove an incorrect analysis and/or extraction in the program). If the plots on the graphing calculator are the same shape from the original picture (i.e. the coordinates on the graphing calculator match given coordinates extracted from the image in the program), we have proven that we can determine whether the shapes were closed or opened.

The rest of the paper is organized as follows: Section 2 gives the details on the challenges that we met during the experiment and implementing solid tests that check for the overall correctness of our project; 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. Avoid Confusion

When implementing file-opening components in a program, we want to make sure that the file we open is the one we desire to check its shape for. To omit this confusion, we name our files specifically beforehand, naming them "complete"/open or "incomplete"/closed [6]. In real-world terms, the user will not know whether or not their shape is open or closed, and they are going to want the confidence of knowing that it is their shape that is being checked (not some other file); that is why they will use our application to check their shape. However, we use the os Python

library, which guarantees file safety measures to ensure 100% accuracy regarding opening the correct file.

2.2. Identify the Pixel

For the Pixel Fixer design we made to close the opened shapes, we wanted to make sure that there are no pixels in the shape that do not meet the adjacency requirements (2 pixels adjacent). To identify a pixel to be an open pixel, the program checks for an adjacent pixel in eight different directions; if the number of adjacent pixels is less than two, the pixel was identified as an incomplete (or open) pixel. To make sure the pixels that were connecting are the incomplete pixels, we initialize an empty list and append each pixel that is identified as incomplete pixels into this list. When the user clicks to complete the image, the turtle object is placed at the first pixel in the list within the GUI and the program will go to the closest pixel. Once the turtle pen finishes its job of drawing, the program indicates a complete drawing in the GUI (there are no gaps in coordinates, and this is where incomplete images are most prominently indicated).

2.3. Make Sure the Image is Complete

In the program, we must make sure that an image that was identified as complete is actually a complete shape. The program will scan and record the coordinates of each pixel into an empty list that has a color that isn't white (this is supposed to be the color of the canvas). The program will go through every pixel in the list and check for the eight adjacent directions and see if these pixels are in the list of colored pixels. The pixel could only be identified as a closed pixel if two or more pixels adjacent were colored, and a shape will only be identified as closed if all pixels were identified as closed [9]. And when the shape was identified as closed, the program will then go on and display that the shape is a closed shape.

3. SOLUTION

Pixel Checker is an application that utilizes python libraries including turtle and tkinter. The Pixel retrieving component includes implementation of the PIL library, which is an image-based library used for photo-related data retrieval. The program, after retrieving the file in jpg format, goes through each pixel in the image using a for loop, and if a pixel's color isn't equal to white, or 0,0,0,0, the program will append the coordinate x and y of the pixel into a list for other parts of the program to use (this means that every coordinate in the list is represented as a tuple). Once all pixels have been retrieved, the program goes through a pixel checking process, which goes through each pixel to see if there are two pixels that are adjacent to the current pixel in the list. If a pixel has less than two colored pixels adjacent to it, it will be identified as an open pixel, and will be handled accordingly to fix the shape. The program only indicates a shape to be closed if and only if all of it's pixels are closed pixels; otherwise, the program indicates that the shape is open. The shape drawing component implements the turtle library, where the turtle will draw out lines when it's commanded to put it's pen down, it is implemented so that it puts its pen down as the first pixel of the list of colored pixels, and goes through all the pixels that were colored. Each pixel that was drawn was appended into another separate list to ensure the turtle didn't go on to redraw any pixel that was drawn. The shape fixer is similar to the shape drawer, in the sense that it goes through a list of pixels to move the turtle to its indicated location. Only in this component, however, the program goes through the list of opened pixels and draws it to the closest colored pixel to close up the shape. While there is a small portion of information being displayed onto the Console after drawing the image, the visual parts of the program that displays the drawing were put together with the GUI using the tkinter library (from a turtle screen).The GUI utilizes the screen with buttons for the user to interact with. The user can upload and select an image from a drop down menu. When the user clicks on the check image, the check image and draw image

functions will be enacted to check the image for its closeness, and send an error message when there wasn't an image chosen. The program will draw out the image first, and if the shape is not closed, a pop up window will appear to tell the user that the shape isn't closed, and the user can click on the fix image button to fix and redraw a complete shape. Once the shape was identified as completed, a button will appear and show how the program goes through and check all the pixels of the image. After this, the program resets, the user can select another image to check.

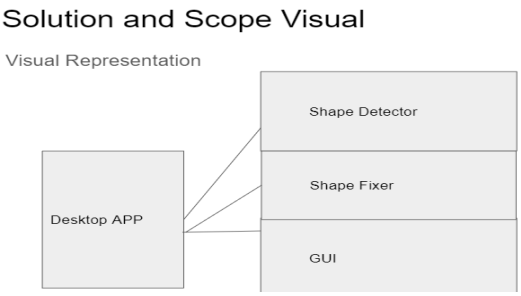


Figure 1. Overview of the system

```
class GUI:

  def __init__(self):
    self.homepage = turtle.Screen()

    self.homepage.bgcolor("light yellow")
    self.homepage.title("Pixel Processor")
    self.font = tkinter.font.Font(family="Times New
    Roman", size = 7, weight="normal")
    self.canvas=self.homepage.getcanvas()

    self.image_name = "Choose Your Image"
    turtle.clear()


  def reset_screen(self):
    self.homepage.clear()
    self.homepage.bgcolor("light yellow")
    self.homepage.title("Pixel Processor")


  def initialize_upload_check_buttons(self):
    upload_button= tkinter.Button(self.canvas.master,
    height=1, width=10, bg="white", command=
```

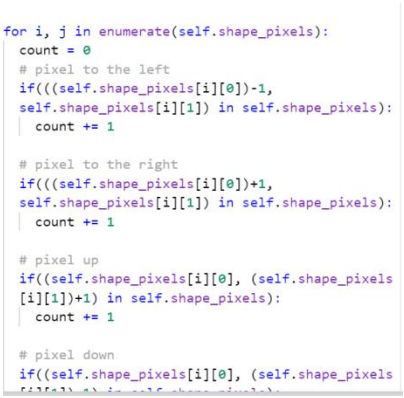
Figure 2. Code of class GUI

```
def draw_shape(self):
  turtle.clear()
  self.is_closing_shape()
  if 0 == 0:#self.closed_shape == True:
    list_of_pixels_already_drawn = []
    coordinate = self.shape_pixels[0] #coordinate
    = (x, y); coordinate[0] = x; coordinate[1] = y

    turtle.penup()
    turtle.goto((coordinate[0], coordinate[1]))
    turtle.pendown()

    for i, j in enumerate(self.shape_pixels):
      # pixel to the left
      if((((coordinate[0])-1, coordinate[1]) in
      self.shape_pixels) and (((coordinate[0])-1,
      coordinate[1]) not in
      list_of_pixels_already_drawn)):
        turtle.goto(((coordinate[0])-1, coordinate
        [1]))
        list_of_pixels_already_drawn.append((
        (coordinate[0])-1, coordinate[1]))
        coordinate = ((coordinate[0])-1, coordinate
        [1])
```

Figure 3. Code of draw shape



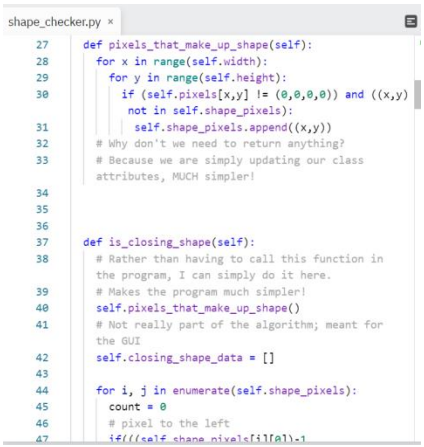
```
for i, j in enumerate(self.shape_pixels):
  count = 0
  # pixel to the left
  if(((self.shape_pixels[i][0])-1,
  self.shape_pixels[i][1]) in self.shape_pixels):
    count += 1

  # pixel to the right
  if(((self.shape_pixels[i][0])+1,
  self.shape_pixels[i][1]) in self.shape_pixels):
    count += 1

  # pixel up
  if((self.shape_pixels[i][0], (self.shape_pixels
  [i][1])+1) in self.shape_pixels):
    count += 1

  # pixel down
  if((self.shape_pixels[i][0], (self.shape_pixels
```

Figure 4. Code of enumerate



shape_checker.py ×

```
27  def pixels_that_make_up_shape(self):
28    for x in range(self.width):
29      for y in range(self.height):
30        if (self.pixels[x,y] != (0,0,0,0)) and ((x,y)
           not in self.shape_pixels):
31          self.shape_pixels.append((x,y))
32    # Why don't we need to return anything?
33    # Because we are simply updating our class
      attributes, MUCH simpler!
34
35
36
37  def is_closing_shape(self):
38    # Rather than having to call this function in
      the program, I can simply do it here.
39    # Makes the program much simpler!
40    self.pixels_that_make_up_shape()
41    # Not really part of the algorithm; meant for
      the GUI
42    self.closing_shape_data = []
43
44    for i, j in enumerate(self.shape_pixels):
45      count = 0
46      # pixel to the left
47      if(((self.shape_pixels[i][0])-1
```

Figure 5. Code of pixels that make up shape

4. EXPERIMENT

4.1. Experiment 1

We check for every pixel in the picture that was colored (a pixel with a different RGB color value than the background's RGB color value). The program will go through each one of the shape pixels, and for the current pixel that we are checking, we check for the adjacent pixels around it (8 different directions). If there are two adjacent pixels to the current pixel that are in the shape pixels list (and thus meet the RGB value specification), we continue to check the next pixel in the list of shape pixels. If there is a pixel in the list of shape pixels that does not have two adjacent pixels to the current pixel that are in the shape pixels list, we end the function and the shape is not complete. If all of the pixels in the list of shape pixels meet the specified requirements, we can indicate that the shape is complete.

Using a professional graphing website: Desmos, we can specify all of the coordinates from the shape pixel (after projecting the coordinates in the program) to prove that there are two adjacent pixels in the direction. In the graph, the distance from each adjacent pixel should be no greater than 1 coordinate point away from each other.

That is, for a pixel, pixel1 that is adjacent to the pixel, pixel2, we expect that the distance equation below should follow for all of the pixels in the list of shape pixels:

Distancei ≤ 1, or:
|pixel1(x) - pixel2(x)| ≤ 1 or |pixel1(y) - pixel2(y)| ≤ 1
Coordinates from square_complete.py that are in shape_pixels list:
[(20, 20), (20, 21), (20, 22), (20, 23), (20, 24), (20, 25), (20, 26), (20, 27), (20, 28), (20, 29), (20, 30), (20, 31), (20, 32), (20, 33), (20, 34), (20, 35), (20, 36), (20, 37), (20, 38), (20, 39), (20, 40), (21, 20), (21, 40), (22, 20), (22, 40), (23, 20), (23, 40), (24, 20), (24, 40), (25, 20), (25, 40), (26, 20), (26, 40), (27, 20), (27, 40), (28, 20), (28, 40), (29, 20), (29, 40), (30, 20), (30, 40), (31, 20), (31, 40), (32, 20), (32, 40), (33, 20), (33, 40), (34, 20), (34, 40), (35, 20), (35, 40), (36, 20), (36, 40), (37, 20), (37, 40), (38, 20), (38, 40), (39, 20), (39, 40), (40, 20), (40, 40), (41, 20), (41, 21), (41, 22), (41, 23), (41, 24), (41, 25), (41, 26), (41, 27), (41, 28), (41, 29), (41, 30), (41, 31), (41, 32), (41, 33), (41, 34), (41, 35), (41, 36), (41, 37), (41, 38), (41, 39), (41, 40)]

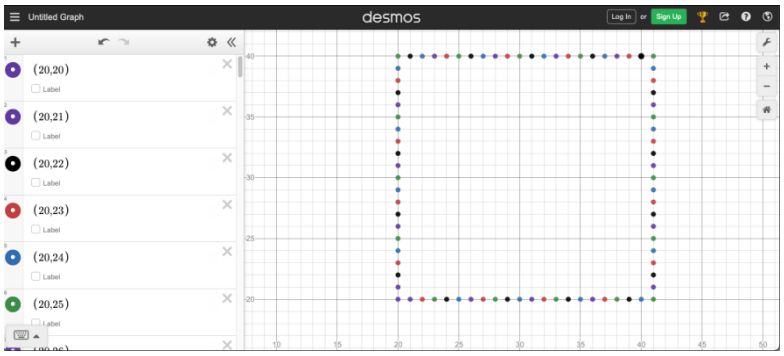


Figure 6. Square

For every pixel in the graph, there are two pixels in an adjacent direction with a distance (based on the formula above) of exactly 1; and thus, we have proven that the shape is complete [10].

4.2. Experiment 2

If the shape was identified as incomplete, (in other words, there is one pixel that did not have two adjacent pixels that were in the shape), a method in the shape checker known as draw_pixel is called. In this method, we start by storing all of the incomplete pixels to a separate list (this step is very similar to the is_complete_shape method; except now we are taking into consideration the incomplete pixels rather than the complete pixels). Using the GUI, the method will go to the specified coordinate of the first pixel in the list. Using a turtle, the turtle object goes to this coordinate. Next, the method moves onto the second pixel (the last pixel that is incomplete in the image), along with the turtle object moving forward to the specified coordinate in the GUI (it's important to note that the pen is down (turtle.pendown(), so there is a line being drawn from the first pixel in the list of incomplete pixels to the second pixel in the list of incomplete pixels). Once this line connects the two pixels together, we can then identify the shape as being complete. We know this to be true because for every x-y coordinate in the path of the line that was drawn, we assume there to be a coordinate in place so long as there are no gaps in the GUI within the shape's figure at this point.

Using a professional graphing website: Desmos, we can specify all of the coordinates from the shape pixel (after projecting the coordinates in the program) to prove that we have properly identified the two pixels that are open (incomplete). Open pixels are specified by the following requirement:
For every open pixel i in graph G:
∑(adjacent pixelsi)≤ 1,
(Distancei from any non adjacent pixel) > 1

[(34, 25), (34, 26), (34, 27), (34, 28), (34, 29), (34, 30), (34, 31), (34, 32), (34, 33), (34, 34), (35, 34), (36, 34), (37, 34), (38, 34), (39, 34), (40, 34), (41, 34), (42, 25), (42, 34), (43, 25), (43, 34), (44, 25), (44, 26), (44, 27), (44, 28), (44, 29), (44, 30), (44, 31), (44, 32), (44, 33), (44, 34)] ; coordinates for the line: (34, 25), (42, 25)

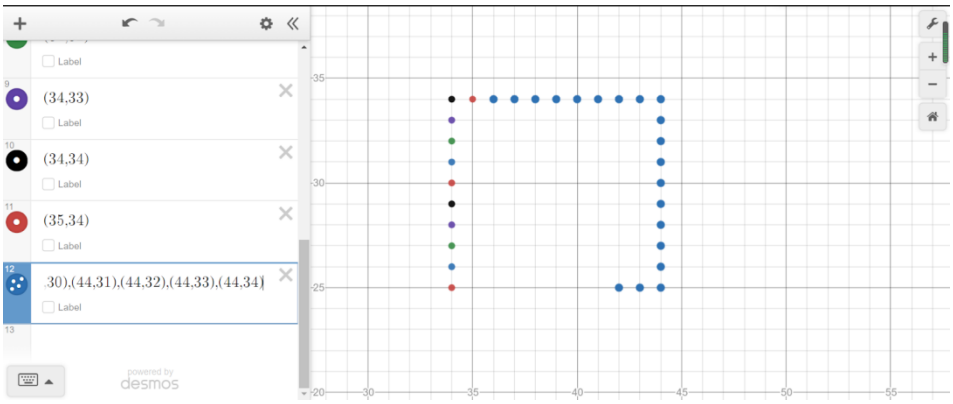


Figure 7. Desmos graph (screenshot after the line is drawn)

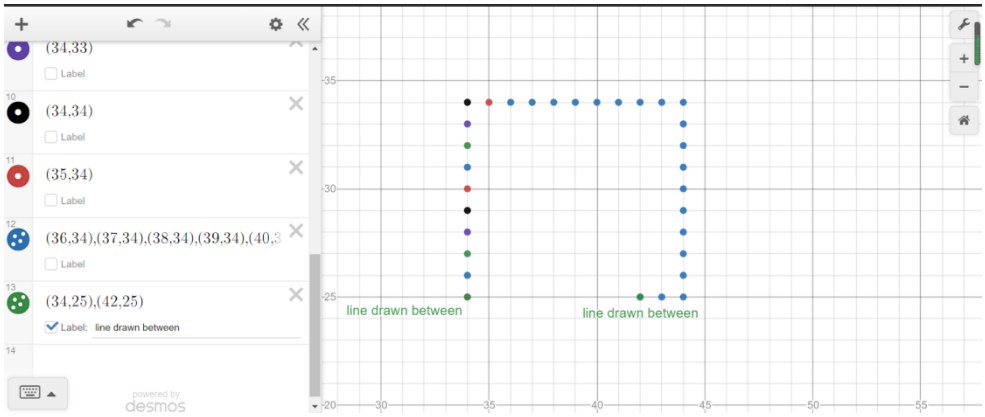


Figure 8. Desmos graph 2

We wanted to prove for the pixels that met the adjacency requirement, and based on the graphs for the image, that you can see these pixels have two other pixels adjacent with a distance on the graph next to each other equivalent to 1. For the pixels that did not meet the adjacency requirement, we can see that they have only one adjacent pixel with a distance of 1. The program identifies the two incomplete pixels, and it will draw a line between them (in the GUI we draw a connecting line between them). We have proven the summations of adjacent pixels for open and closed pixels, the distances for adjacent pixels for open and closed pixels (and how they determine a closed shape overall).

Through the experiment, we have proven that the program succeeds in determining whether the shape is closed or open, and how it could identify which pixels need to be connected with a line to close up the shape. The program will identify whether two of the pixels adjacent with a distance of 1 were colored to determine the pixel to be closed.

For example, the completed square could be identified correctly when every pixel in the shape has two adjacent pixels with a distance of 1; this requirement means that all of its pixels were complete.

In the incomplete square the program can identify the incomplete pixels (i.e. with only one pixel adjacent to it with distance of 1), and can draw a line from both pixels to connect them. The line drawn has no gaps (because a line is simply one solid shape that connects to the shape). And because a shape connects to another shape to close any gaps (and thus create another adjacent pixel with a distance of 1), our shape is complete after the line is drawn.

5. RELATED WORK

Utilizes previously written data to fill in image gaps [11]. As stated in the article, "Based on the assumption that the same-class neighboring pixels around the un-scanned pixels have similar spectral characteristics, and that these neighboring and un-scanned pixels exhibit similar patterns of spectral differences between dates, we developed a simple and effective method to interpolate the values of the pixels within the gaps". They are attempting to fill in image gaps by selecting certain pixels. Given that they are searching for a similar pattern to fill in image gaps (scanned and unscanned pixels), we are searching for a similar pattern to fill in image gaps (pixels that meet a specific adjacency requirement to be determined as closed or open pixels). They are using a different form of technology to retrieve the pixels (they use Landsat ETM+, while we used a method of pixel extraction). And the images that they are testing are quite different as well (they use SLC-off ETM+ images, while we use .jpeg images created from a individualized pixel-focus drawing platform: pixlart). We can assume that they possibly used thousands of pixels, while our implementations, while they can handle this quantity of pixels, only used a few dozen.

The research tests how open and closed shapes affect the trend of judgement regarding the processing of shapes [12]. This research determines open and closed shapes for parts of their experiment. Rather than judging on whether the shape is open or closed, this tests how the open or closed shape affects the running time of finding a target that is a particular open or closed shape.

A method in Java attempting to extract edge end pixels for further image analyzing to link two edges together [13]. The research extracts pixels from images through methods focused on direction sensitives, as well as linking edges together from 2-d images (the edge segmentation on theirs, the line drawn on ours). The research was based on Java, while ours is based in python. This research focuses on analyzing the directions that the two edges are facing from each other, while our research gathers the information regarding the two edges without further analysis (simply draws a connecting line between the two).

6. CONCLUSIONS

The pixel checker utilizes a method to extract all colored pixels in a shape, and then holds on to certain pixels (the ones that make up the shape we are checking for) based on their RGB value [14]. The program will append all pixels into a list, and check through each of them. If each pixel has two pixels adjacent to it in any of the eight different directions horizontally, vertically, or diagonally (up, down, left, right, up-left, upright, down-left, or downright). If all the pixels in the shape meet this requirement, it will be identified as a closed pixel; otherwise, the pixel will be identified as open. The program will tell the user whether the shape is closed or not (using the specified method), and close the shape when it isn't. The program will close an open shape by appending all the open pixels into a separate list (in our experiments, there is a total of 2 pixels in the simple pixel-art based images). Then, we use a turtle.pen object to draw through all open pixels in the shape by going from the first pixel to the second pixel, which results in a line being drawn. Once the line is drawn, we have no pixels without two adjacent pixels in our shape, and

thus we can conclude that we have closed an open image. Using images drawn on pixlart, we used the shape's pixels and plotted them on a graphing calculator. If the adjacency rules specified above in the experiments are met in the plotted points on the graph and in the coordinates in the program, we expected the results for both to determine closed shapes (our program would indicate this, while we would investigate every coordinate manually on the graphing calculator). For open shapes, we wanted to compare the two edge points (the two points that did not meet the adjacency requirement). If these two points did not meet the adjacency requirement manually by checking the graphing calculator and in the program, we can determine that our program correctly checks for both open and closed shapes.

For now, our program only draws lines through the pixels that are open, and cannot go back to check if the shape could be completed correctly. Our current program can only run simple pixel art images (our pixel checker's adjacency requirements allow us only to check for images with shapes that have two pixels adjacent to them without the possibility for more than this amount with complete accuracy). The current fixing method only closes up the shape by drawing a line that connects the two points together that are open. In terms of optimizing our program, we currently use brute-force to check for all of the given coordinates in our list of shape pixels. We have not implemented a dynamic-programming related concept nor a recursive algorithm to work our way through all of the pixels in the shape [15]. Doing so would drastically reduce the overall running time of the algorithm for the shape checker.

In the future, we could implement machine learning into the program to allow more advanced techniques to draw the pixels in our image and allow for more optimized solutions to fix open shapes. This will allow our program to handle more complex images, and to further our adjacency requirement for a higher accuracy rate on more complicated images (and hence we could possibly run this program on images that are not just single-pixels).

REFERENCES

[1] Geliskhanov, Islam Zelimkhanovich, and Tamara Nikolaevna Yudina. "DIGITAL PLATFORM: A NEW ECONOMIC INSTITUTION." Quality-Access to Success 19 (2018).

[2] Fisher, Peter. "The pixel: a snare and a delusion." International Journal of Remote Sensing 18.3 (1997): 679-685.

[3] Simader, Christian G. "Outline." On Dirichlet's Boundary Value Problem. Springer, Berlin, Heidelberg, 1972. 1-12.

[4] Gell-Mann, Murray. "What is complexity?." Complexity and industrial clusters. Physica-Verlag HD, 2002. 13-24.

[5] Hawthorne, John. "The case for closure." Contemporary debates in epistemology (2005): 26-43.

[6] D'MELLO, SIDNEY K., and Arthur C. Graesser. "Confusion." International handbook of emotions in education. Routledge, 2014. 299-320.

[7] Kendall, David George, et al. Shape and shape theory. Vol. 500. John Wiley & Sons, 2009.

[8] Osada, Robert, et al. "Shape distributions." ACM Transactions on Graphics (TOG) 21.4 (2002): 807-832.

[9] Sloan, Peter-Pike J., Charles F. Rose III, and Michael F. Cohen. "Shape by example." Proceedings of the 2001 symposium on Interactive 3D graphics. 2001.

[10] Hengl, Tomislav. "Finding the right pixel size." Computers & geosciences 32.9 (2006): 1283-1298.

[11] Chen, Jin, et al. "A simple and effective method for filling gaps in Landsat ETM+ SLC-off images." Remote sensing of environment 115.4 (2011): 1053-1064.

[12] Burlinson, David, Kalpathi Subramanian, and Paula Goolkasian. "Open vs. closed shapes: New perceptual categories?." IEEE transactions on visualization and computer graphics 24.1 (2017): 574-583.

[13] Pathegama, Mahinda, and Özdemir Göl. "Edge-end pixel extraction for edge-based image segmentation." ICSP, 2004.

[14] Süsstrunk, Sabine, Robert Buckley, and Steve Swen. "Standard RGB color spaces." Color and Imaging Conference. Vol. 1999. No. 1. Society for Imaging Science and Technology, 1999.
[15] Bellman, Richard. "Dynamic programming." Science 153.3731 (1966): 34-37.

ARTIFICIAL INTELLIGENCE SYSTEMS FOR TEACHING AND LEARNING IN CIVIL ENGINEERING: CONCEPTUAL FRAMEWORK

L O Toriola-Coker[1], N AYekini[1], H Alaka[2] and A O Ajimo[1]

[1]School of Engineering, Yaba College of Technology, Yaba Lagos, Nigeria
[2]University of Hertfordshire, Hatfield, Hertfordshire, UK

ABSTRACT

Artificial intelligence technology is based on design of machine or computer application that mimic human intelligent. Use of artificial intelligence in teaching and learning in civil engineering is a welcome development. This paper presents a conceptual framework of Artificial Intelligence Systems for Teaching, Learning, and administration of in Civil Engineering education. The proposed system is to be designed using the following tools: Extensible Markup Language (XML) to develop the GUI, Hypertext Pre-processor (PHP) for the web user interface (WUI), APACHE for middleware, MYSQL for database design, and UML will be used to visualize the design of the system. If the system is developed and implemented, it will go a long way to advance teaching and learning, and educational administration in civil engineering profession.

KEYWORDS

Artificial Intelligence, learning management system, Civil Engineering, MYSQL, Genetic algorithm.

1. INTRODUCTION

Artificial intelligence (AI) has been developed through research since 1956 and was firstly used at the meeting held in Dartmouth College United State of America [1]. This was developed as a comprehensive discipline based on the teamwork of various kinds of disciplines, such as computer science, cybernetics, information theory, psychology, linguistics, and neurophysiology and civil engineering. It can also be refereed as a branch of computer science involving research, design and application of intelligent computer [1], [2]. Technological development products and establishment of relevant theories where people can progress is the aim of (AI) while its objective is to explore how to imitate and execute some of the intelligent function of human brain and other relevant technologies [3] Artificial intelligence is playing a critical role in civil engineering allowing for substantial increases in automation, directing digitalization and intelligence, reliability and performance by establishing an active connection between physical and digital construction [4], [5]. AI rise and fall in the 1950s before making great progress with the development of the fifth-generation computer in the 1980s. AI research by a single intelligent agent began to turn to the study of distributed artificial intelligence based on network environment in the 1990s where new upsurge of the research of artificial intelligence: with the development of network technology especially at the establishment of international internet technology [1], [6] Civil engineering industry as a large economic sector can influence national progress and growth as well as term development growth [7]. A projected amount of 15% of global gross domestic product (GDP) was on the rise in 2020 compared to 13% of global GDP in 2017 which was gotten from the global construction industry surveyed by McKinsey Global

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 59-74, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120405

Institute survey during their conducted exercise in 2017 [8]. Considering construction sector as one of the major contributor to the world economy representing 13% of the global GDP with a promising 85% to $15.5 billion globally by the year 2030 with three leading countries – China, the United States and India – contributing 57% of its global demand [9]. Moreover world's technology leaders and governments around the world should continue to put in more effort into the implementation of AI in order to gain a competitive advantage [10]. The United Kingdom for example, recently signed an agreement by investing an additional €1.8 billion to make the country more creative and innovative putting the country at the forefront of the AI industry [11], [12]. Likewise, France proposes to spend €1.5 billion on AI science.

AI became more practical because society study not only the same goal-based distributed problem solving, but also the multiply intelligent agent's problem solving. Additionally, a thriving scene of artificial neural network research and application emerged and it had been deep into all areas of life including civil engineering learning program [1]. The experience and knowledge are unscientifically incomplete and inaccurate, and they cannot be handled by traditional procedures. However, artificial intelligence has its own superiority. It can solve complex problems to the levels of authorities by imitating the professionals. All in all, artificial intelligence has a broad application prospect in the practice of civil engineering learning.

The main theories and methods of artificial intelligence can be described as a distributed, statistical, symbolic, behaviorism and connectionism technique that focuses on simulating human functions before moving on to understanding impressionist acceptability in communication focusing on intellectual responsibilities before moving into the field of earlier approaches such as modelling and analytical representation [13]–[15]. In 1950s, AI had lots of anticipations and visions about it have been generated. Let us now elaborate the modern progress of artificial intelligence technology in all aspects of civil engineering learning and their relationship.

First, such technology provides advanced methods of learning with a competitive advantage at the tertiary level; however, they can have a positive effect on the globalization of education and researches [16], [17]. Application of this technology nowadays in many fields is fast approaching such as such as expert system, knowledge base system, intelligent database system, and intelligent robot system [1]. The decision-making and the knowledge management of the 21st century is the expert system, which is widely acceptable as the earliest, most extensive, most active and most fruitful area in learning process. In the field of civil engineering, many problems, especially in engineering design, program decision-making and construction management were influenced by many reservations which could be solved not only in need of physics, mechanics, and mathematics calculations but also depend on the experience of experts.

Progress report were presented in the field of adaptive civil-engineering structures by Adam and Smith in 2008 describing how Self-diagnosis, multi-objective shape control, and reinforcement-learning processes were implemented within a control framework on an active tensegrity structure [18]. The suitability for modelling complex systems with known input-output data sets were among the usefulness of artificial intelligence-based computational techniques and adaptive neuro-fuzzy inference systems. Such systems can be efficient in modelling nonlinear, complex, and ambiguous behavior of cement-based materials undergoing single, dual, or multiple damage factors of different forms in civil engineering.

Recently, the development of the genetic algorithm presented many new mathematical tools and engrossed civil engineering learning as the latest achievement of applications. Due to a lot of undefined and complicated influence factors in civil engineering, each project has its individual character and generality; function of expert system in the special links and cases is a notable effect [1], [19]. We can expect, along with the computer technology, the genetic algorithm in

civil engineering application will be more general and more effective. Over the past 20 years, in the civil engineering courses and profession, progress and application of the expert system have made a lot of accomplishments, mainly used in decision-making and prediction, building design and optimization, project evaluation, diagnosis, the construction technology and project management of roads, rails, bridges including health detection and some special field, and so on.

In recent time's artificial intelligence have been widely used in many relevant civil engineering courses such as geo-technical engineering, structural engineering, water resource engineering and so on. In the field of civil engineering, many problems, especially in engineering design, construction management and decision-making program were influenced by many doubts which could be solved not only in need of mathematics, physics and mechanic calculations but also depend on the experience of professionals. Civil engineering students need to learn how to deliver practical sustainable solutions for the engineering projects [19]. Applied assessment and award techniques was established by Thompson [20]saying it can be usefully used as teaching tools. Construction courses based on enhancing virtual computing technologies using agent-based techniques can be deploy for an e-learning environment conditions [21].

The aim of this study is to introduce AI as an auspicious administrative and learning tools that can enhance current learning efforts in the civil engineering courses including its application in construction management, geo-technical engineering, highway engineering, structural engineering, traffic engineering, water engineering and so on. The objective will be focusing at the contributions in the direction of AI in civil engineering learning in all tertiary institution. This will provide theoretical foundation or may play an important role in the development of artificial intelligence in civil engineering courses; would represent the echelons of current research of artificial intelligence in civil engineering and facilitate sustainable research efforts.

2. LITERATURE REVIEW

Artificial intelligence (AI) has a subfield known as Evolutionary computation (EC) which uses iterative process (often inspired by biological mechanisms of evolution) to evolve a population of solution to a desired end [1]. The application has been in existence for quite a long time within the arena of civil engineering with an effective usage for solving complex optimization problems.

The first journal article published on neural network application in civil/structural engineering was in 1989 by Adeli since then the article reviewing neural network articles published in archival research journals [22]. The great majority of civil engineering applications of neural networks are based on the simple backpropagation algorithm. AI was applied to stimulate interest within the civil engineering research community for developing the next generation results in learning.

Hayes-RothFrederick [23] in his earliest research described Genetic algorithms (GAs) as one of the renowned evolutionary algorithms which simulate the Darwinian principle of evolution and the survival of the fittest in optimization. It needs to be improved and further studied due to its extensive application in civil engineering field. A lot of improvement needed to be done in terms of technological usage of (GA) including the hybrid genetic algorithm, the dynamic adaptive technology, using nonstandard genetic operators, and the parallel genetic algorithm and so on.

Various types of Machine Learning (ML) such as (GA), Neural Network (NN), Linear Regression, Logistic Regression, Nearest-Neighbor Mapping, Decision Trees, K-Means Clustering, Random Forests, and Support Vector Machines exist for ML model implementation [9]. Several factors needed to be considered when choosing ML to be used in training time,

accuracy, and so on. AI and ML algorithms have been attempted for use in civil engineering fields by some researchers [24]

The application of (GA) in civil engineering will be more effective and efficient if multi-objective optimization model for the scheduling of linear construction projects are implemented [25]. This model permits construction planners to appraise and generate optimal/near-optimal construction scheduling plans that minimize both project time and cost. Recently, introduction of (GA) improve many new mathematical tools and absorbed civil engineering as the latest achievement of applications.

Adaptation of immune system of a living creature was stimulated by Artificial immune system (AIS) which unravel the various complexities in real-world of engineering optimization problems [26]. Blending of the genetic algorithm and the least-squares method was used to find feasible structures and the appropriate constants for those structures in this system.

The shortcomings of the traditional and artificial neural network-based methods were disabled by the new approach which was presented in the literature for the analysis of civil engineering systems. The algorithm design has many aspects for improvement since the immune system characteristics of the application exploration is still in its initial stage, and the immune system in civil engineering application, still needing further development with the realization of the algorithm, parameter selection, the theory discussion [1].

Electimize novel evolutionary optimization algorithm was developed by Khalafallah & Abdel-Raheem [27] for solving nonlinear construction optimization problems in engineering. Another population-based global optimization technique called Particle swarm optimization (PSO) which enables a number of individual solutions, called particles, to move through a hyper dimensional search space to search for the optimum. This PSO algorithm was used to calculate the final value of every joint point in civil engineering jobs There are many ill factors impact on the quality control that are environment interference, drive saturation, measurement error, rounding error and sampling delay, in the actual operating work of the concrete pump truck [28]. The group control was used to overcome disadvantage, prove mathematical model accurate and identify parameters in full. In order to improve the joint operation real-time and accuracy, using cubic polynomial interpolation method described trajectory; adaptive robust PD control for concrete pump truck boom was used, the advantage of control law are simple and easy to implement and can guarantee a good dynamic and static quality [28]. Through the simulation examples analysis, the conclusion is that the method of combination PSO and adaptive robust PD method is suitable for concrete pump truck boom control in civil engineering works Particle Swarm Optimization (PSO) application was adopted to design and optimize the parameters of the Tuned Mass Damper (TMD) for achieving the best results in the reduction of the building and bridges response under earthquake excitations [29]. The analysis results shows that the designed PSO based TMD controller had an outstanding performance in reduction of the seismically excited example building.

A method that combines the metaheuristics Particle Swarm Optimization (PSO) with the Rough Set Theory (RST) was proposed by Filiberto et al., (2010) in order to carry out the prediction of the resistant capacity of connectors (Q) in the branch of Civil Engineering. The *k*-NN method is used to calculate this value. Similarly, a mathematical model of schedule control based on the quantitative description of the relationship between impact factors with the schedule control of the project was analyzed by [31] developing the various impact factors of project progress. Authors used the particle swarm algorithm to solve the model in order to improve the speed and accuracy of solving civil engineering learning outcomes. The empirical research showed that the method is effective in the field of project schedule control in civil engineering works.

A novel approach based on multi-robot cooperation for inspection and repair of dome structures in civil engineering was introduced by [32]. The results of the simulations show there exists a stable path to fully sweep the surface of a dome. Small scale prototype that was experimented validated the results of findings. Finite difference method (FD) was used for the development of dynamic model of flexible beam structure in civil engineering [33]. The simulated model was validated by comparing the resonance modes with the theoretical values.

The vibration control of the beam was established through nature inspired intelligence method, the Particle Swarm Optimization (PSO), showing comparison of results with (GA) approach. The numerical simulation demonstrates that sufficient vibration suppression can be realized by means of these methods. An improved ant colony optimization (1ACO) was proposed by Kaveh & Talatahari [34] for constrained civil engineering design problems, and applied to optimal design of different engineering problems.

Pareto Ant Colony Optimization was introduced as an especially effective meta-heuristic for solving the portfolio selection problem and compares its performance to other heuristic approaches by means of computational experiments with random instances. An efficient approach to estimate the friction coefficient via an artificial neural network was introduced as a promising computational tool in civil engineering [35]. The estimated value of the friction coefficient was used in one of the civil engineering courses in order to carry out a comparison between the proposed neural networks-based approach and the conventional ones in Manning Equation to predict the open channel flows (water engineering and fluid mechanics).

Another adaptive neural network composed of Gaussian radial functions for mapping the behavior of civil structures controlled with magnetorheological dampers was proposed for effective usage in civil engineering learning [36]. Using three types of earthquakes the proposed controller is simulated.

An AI was used as a technique of back-propagation neural networks to assess the slope failure [37]. The effectiveness of artificial neural networks in the evaluation of slope potential failure demonstrated the numerical results in civil engineering (water engineering) on five major factors, such as the slope gradient angle, the slope height, the cumulative precipitation, daily rainfall, and strength of materials.

The model of cost estimation of highway engineering was set up based on the back- progression (BP) neural network [38]. To come true quick cost-estimating the BP neural network model is trained by a sample data obtained from some performed typical engineering. According to lots of examples, it is sure that the method is practical and the estimating results are reliable. Expression has shown that BP Neural Network in cost estimate of construction engineering has a promising perspective [38].

An application of Neural Network (NN) simulation in civil engineering science was presented by Alacali et al. [39]. Predicting the lateral confinement coefficient in reinforced concrete columns is a very important issue in structural engineering [40]. Therefore, prediction have been developed through several experimental formulas. The confinement coefficient has been predicted recently using soft computing tools for artificial neural networks.

Researchers have realized that solving complex problems and analysis especially in the nonlinear problem in engineering and information technology, neural network potentiality is an important tool. The neural network will be broadly used in the civil engineering field application prospect [1]. The neural network still belongs to the new cross science, itself not perfect. Improvement research of neural network structure and algorithm still in progress where its application studies still have some problems especially in the combination method of the neural network, fuzzy logic

genetic algorithm, and expert system. This will be a striking tool in continuous research field of civil engineering learning with the help of artificial intelligence networking

3. METHODOLOGY AND DESIGN

Our methodology and design will be built on the flaws from analysis of the existing system used during COVID-19 pandemic and subsequent lockdown.

3.1. Analysis of the Existing System

As at the time of venturing into this research work, there was no customize e-learning system for teaching and learning TVET. Majority of the institution that engaged their students virtually used social media platform like Facebook, WhatsApp, Zoom, Telegram etc. We conducted an interview for assessment of the adopted/improvised e-learning system during COVID-19 among the students and teachers in Yaba College of Technology and Lagos State Polytechnics Ikorodu. Out of two-hundred and fifteen (15) circulated to each category of the sampled population, only 199 were returned among the students while 201 were returned among the lecturers. Data collected were tabulated as in Table 1 and 2.

Table 1. Responses from Students.

QUESTIONS			RESPONSE				
		YES	**%**	**NO**	**%**	**UNDECIDED**	**%**
Did your school engaged Students virtually during covid-19 lockdown?	A	189	95.0	3	1.5	7	3.5
Did your school has its own eLearning platform	B	0	0.0	188	94.5	11	5.5
If YES to question 1, did the platform used facilitate the availability of course materials?	C	179	89.9	13	6.5	7	3.5
If YES to question 1, did the platform used promote autonomous learning?	D	191	96.0	6	3.0	2	1.0
If YES to question 1, did the platform enhance the mastering of the course?	E	39	19.6	149	74.9	11	5.5
eLearning platform fosters didactic interaction between the students and lecturers.	F	176	88.4	12	6.0	11	5.5
There is need to design a robust eLearning platform that will take into consideration the needs of the Polytechnic educational sector.	G	199	100.0	0	0.0	0	0.0
The overall rating/assessment of eLearning platform adopted by your institutions is satisfactory with curriculum requirement of your course/s.	H	12	6.0	181	91.0	6	3.0

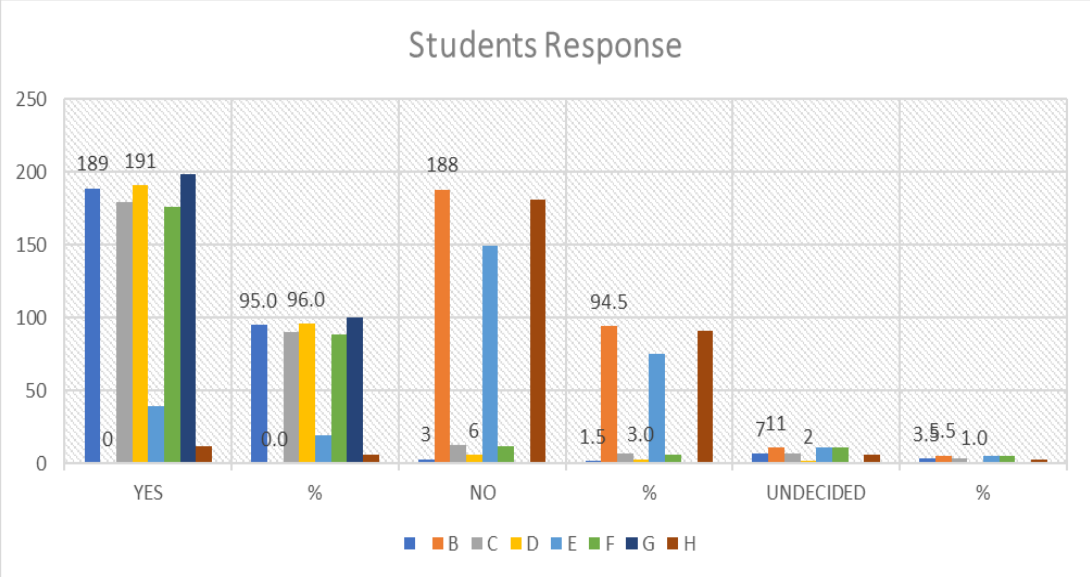


Figure 1. Student response

Table 2. Responses from the Lecturers

QUESTIONS		RESPONSE					
		YES	%	NO	%	UNDECIDED	%
Did your school engaged Students virtually during covid-19 lockdown?	A	201	100.0	0	0	0	0.0
Did your school has its own eLearning platform	B	0	0.0	201	100	0	0.0
If YES to question 1, did the platform used facilitate the availability of course materials?	C	63	31.3	37	18	101	50.2
If YES to question 1, did the platform used promote autonomous learning?	D	89	44.3	112	56	0	0.0
If YES to question 1, did the platform enhance the mastering of the course?	E	21	10.4	171	85	9	4.5
eLearning platform fosters didactic interaction between the students and lecturers.	F	181	90.0	20	10	0	0.0
There is need to design a robust eLearning platform that will take into consideration the needs of the Polytechnic educational sector.	G	201	100.0	0	0	0	0.0
The overall rating/assessment of eLearning platform adopted by your institutions is satisfactory with curriculum requirement of your course/s.	H	13	6.5	187	93	1	0.5

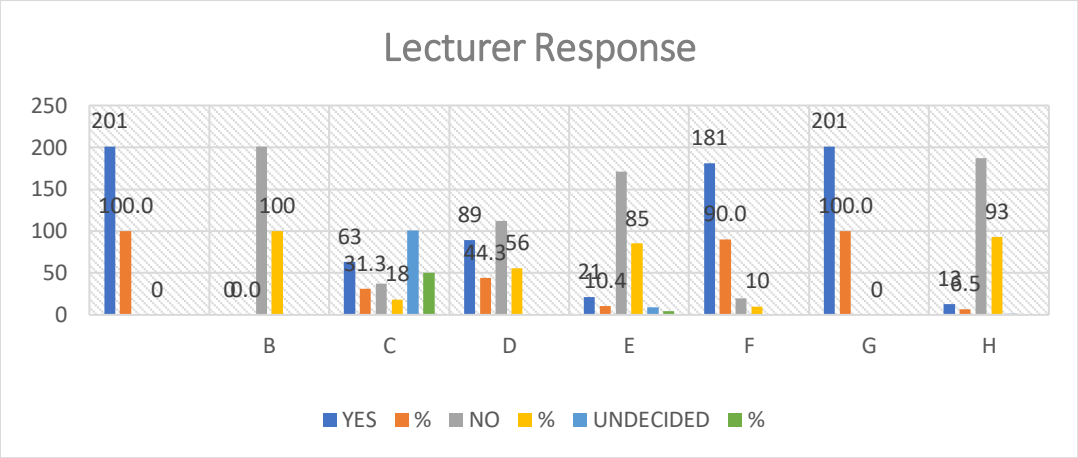


Figure 2. Lecturer response

Our findings from the analysis of data collected showed that majority of sampled population were not comfortable with e-learning platform adopted during COVID-19 pandemic in polytechnics sector. 100% of the sampled population were of opinion that there is need e-learning platform.

As mentioned in the well written and formatted introduction and review literature sections of this research work, artificial intelligence main objective is to encapsulate human intelligent in machine and it is applicable in all area of human endeavor where intelligent is needed to do one thing or the others, civil engineering profession inclusive.

In this section we present conceptual framework of cloud-based artificial intelligence learning management systems for the purpose of automation of civil engineering curriculum in Nigeria polytechnics sector.

3.2. Design Tool

The proposed system will have three categories of users, Administrator, Students, and Instructors (Lecturers and Technologists), and it will provide access to the students and other users through a secure Graphical User Interface (GUI). The proposed tools for the design are Extensible Markup Language (XML), Hypertext Preprocessor (PHP), APACHE, Unified Modelling Language (UML), and MYSQL. Area of the application of each tool in the design of the proposed tools are: Extensible Markup Language (XML) will be used to develop the GUI, Hypertext Preprocessor (PHP) to be used for the web user interface (WUI), APACHE will be used for middleware, and MYSQL to be used for database design, UML will be used to visualize the design of the system.

3.3. Data Collection

The main data required for the purposed of this research is the curriculum for teaching and learning of Civil Engineering, which will be obtain from NBTE.

3.4. Users and their roles

The roles of each user are as stated below. The basic activities of the proposed system will be centered on Course Registration, Audio Visual Learning Platform, and Tutorial (Attendance and grading).

- Administrator shall perform the following roles: manage other users (Students and Instructor), backup database, maintain database etc.
- Instructors (Lecturers and technologists): manage student's account, setup virtual classes, upload curriculum, deploy learning materials, upload student's attendance, anchor virtual classes, attend to student's complaint/question, and grading of the students.
- Learners (Students): view timetable, download courseware, download result, join virtual class, make complaint, attend examination/test, view account status, and playback the previous lecture.

Access to the system will be through internet enabled mobile phones, desktop computer systems, Laptops and other handheld devices, for security of the system and data, all users will have to login before accessing the system. Figure 1, 2, 3, and 4 represents the flowchart of user's activities, and architecture of the proposed system.

4. WORKING OF THE PROPOSED SYSTEM

The GUI will welcome the users to the proposed system, the users ought to have been enrolled and approved by the administrator before getting access to the system. Each user can then proceed to use the system after authentication. If authentication fail the users will not be allow to proceed in using the system.

It worth mentioning that the system will used supervised machine learning to perform some of it functions, consequently may have to submit query that needs to be match with the knowledge base term in the database and the provide solution or responses.

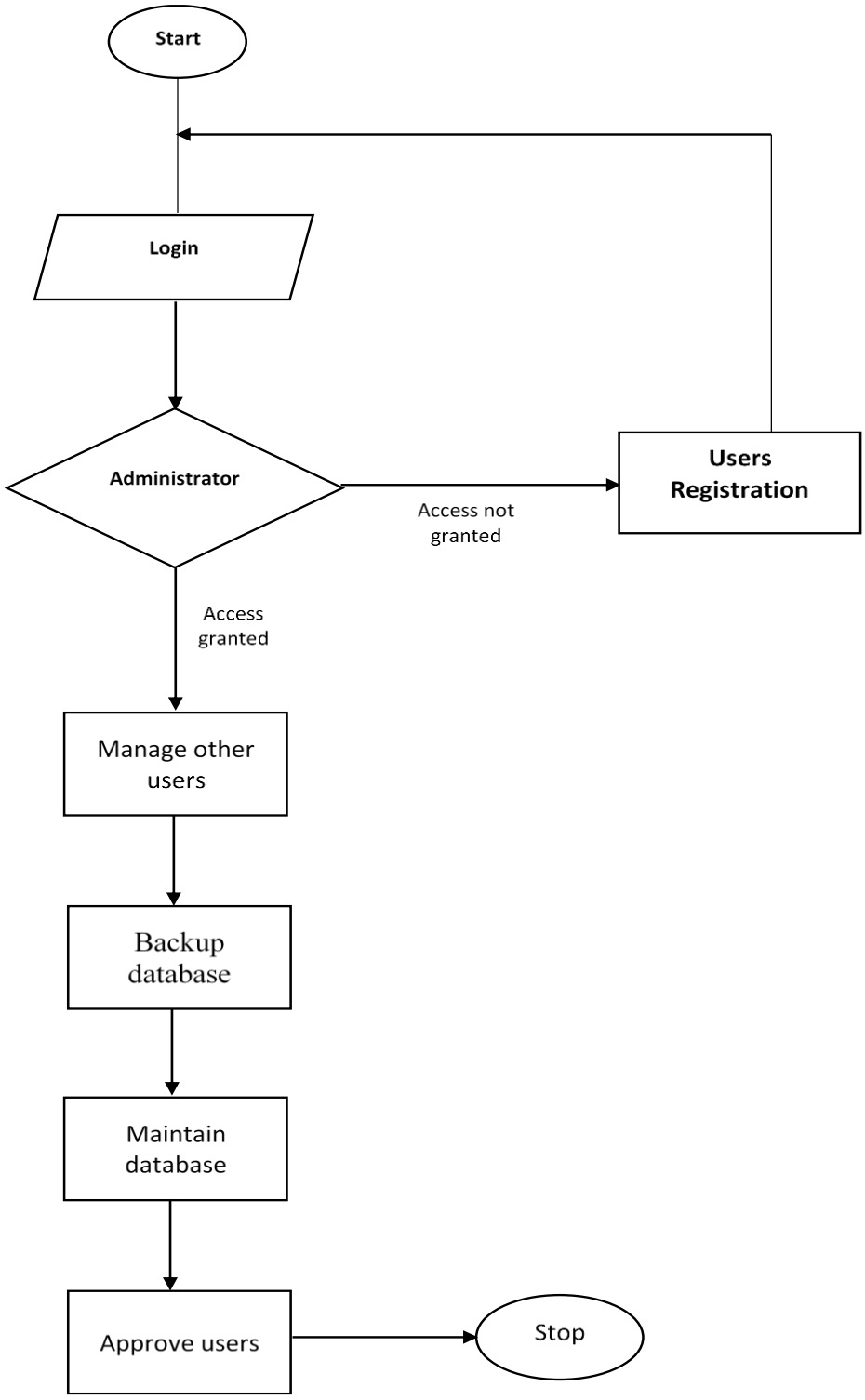


Figure 3. Administrator roles for the proposed system

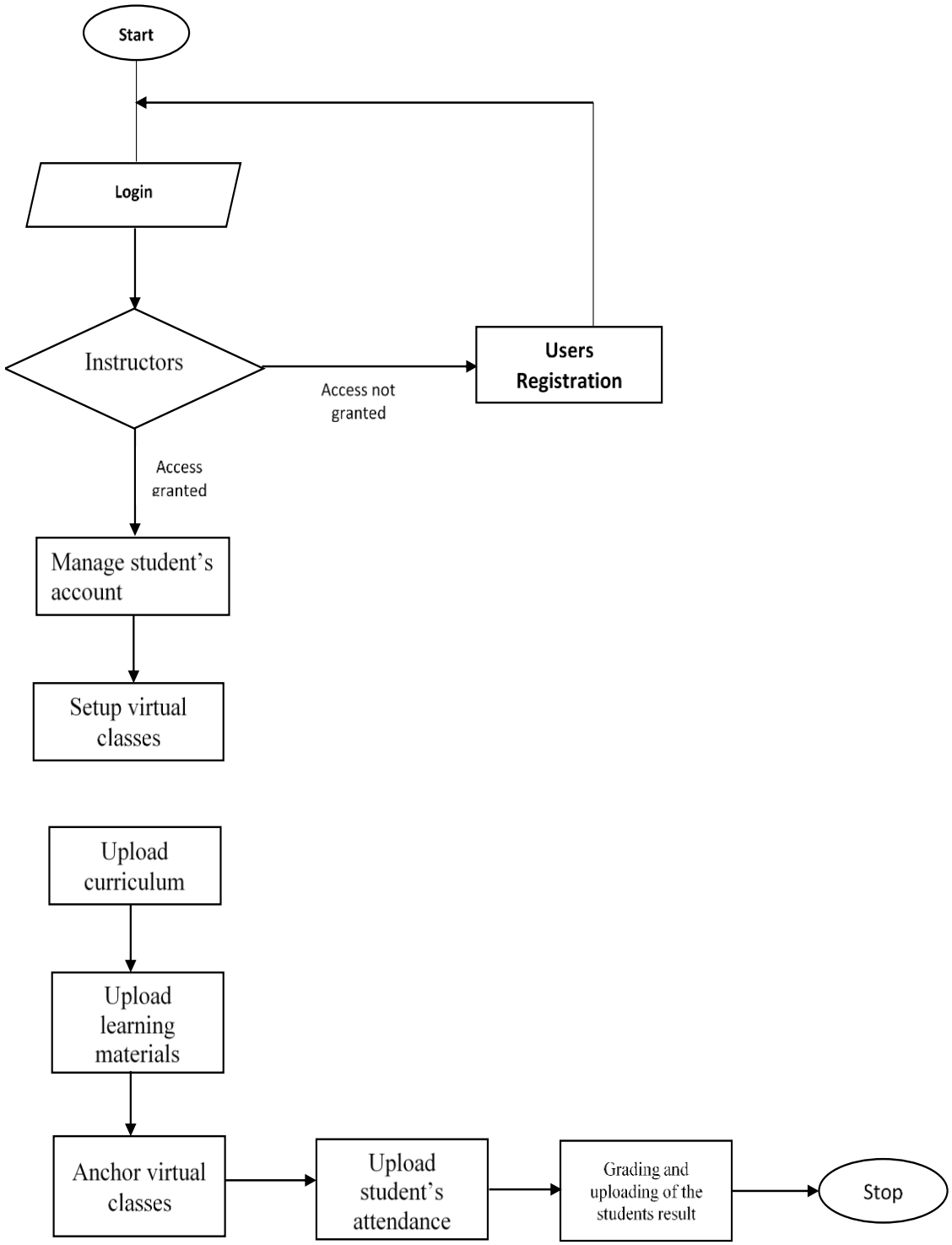


Figure 4. Instructors' roles for the proposed system

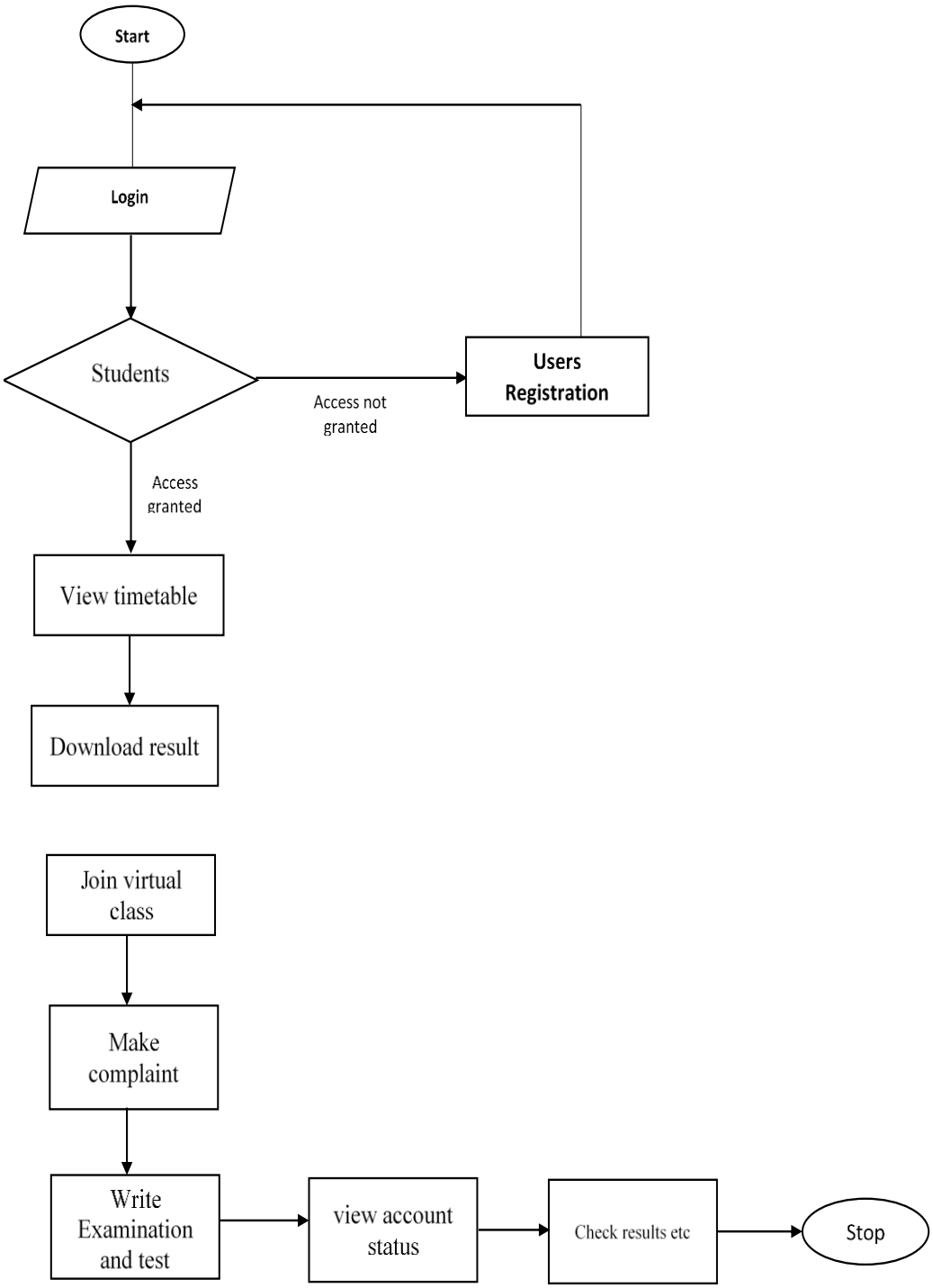


Figure 4. Student's roles of the proposed system

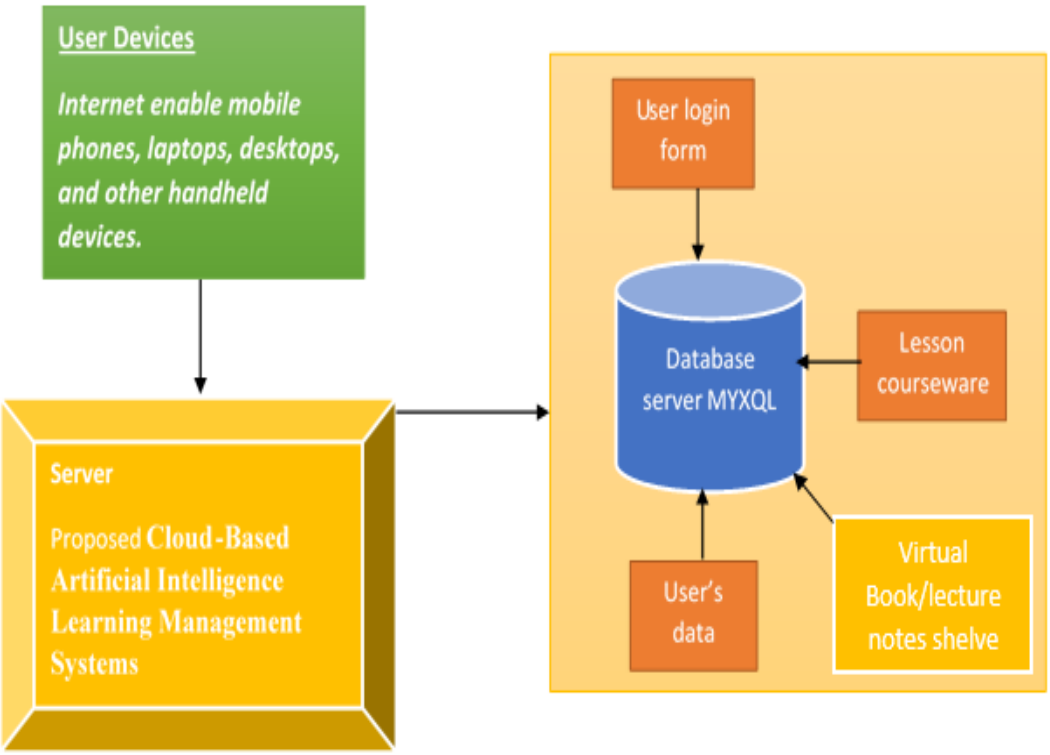


Figure 5. Architecture of the proposed System

5. CONCLUSIONS

This paper recapitulates and presents the relevance of artificial intelligence as an administrative and learning tool in civil engineering education. This research work presents a conceptual framework for effective online learning environment using knowledge based and supervised machine learning approach. The implementation and adoption of the proposed system will allow users access lessons, presenting lectures, grading student's exam/tests. It will also prevent niches where training is carried out completely through physical face-to-face method. Hence improve teaching and learning, and educational administration in civil engineering profession. It can be concluded from the study that artificial intelligence has a great potential to revolutionize the delivery experience of civil engineering programs at higher institutions. Future studies should look to implement the provided framework.

ACKNOWLEDGEMENTS

The authors would like to thank everyone, just everyone!

REFERENCES

[1] P. Lu, S. Chen, and Y. Zheng, "Artificial intelligence in civil engineering," *Math. Probl. Eng.*, vol. 2012, 2012, doi: 10.1155/2012/145974.

[2] A. U. Hosna, A. K. M. F. Ahmad, and J. U. Sun, "A Two-Step Local Search Enhancement with the Novel Solution Representation for Solving Capacitated Vehicle Routing Problems," *Adv. Sci. Lett.*, vol. 25, no. 1, pp. 95–99, Jan. 2019, doi: 10.1166/ASL.2019.13195.

[3] S. Chen, Y. Zheng, C. Cattani, and W. Wang, "Modeling of biological intelligence for SCM system optimization," *Comput. Math. Methods Med.*, vol. 2012, 2012, doi: 10.1155/2012/769702.

[4] D. Luckey, H. Fritz, D. Legatiuk, K. Dragos, and K. Smarsly, "Artificial Intelligence Techniques for Smart City Applications," in *Computing in Civil and Building Engineering*, 2021, vol. 98, pp. 3–15, doi: 10.1007/978-3-030-51295-8_1.

[5] A. Tavana Amlashi, P. Alidoust, M. Pazhouhi, K. Pourrostami Niavol, S. Khabiri, and A. Reza Ghanizadeh, "AI-based formulation for mechanical and workability properties of eco-friendly concrete made by waste foundry sand," *ascelibrary.org*, 2020, doi: 10.1061/(ASCE)MT.1943-5533.0003645.

[6] C. M. Pintea, C. P. Sitar, M. Hajdu-Macelaru, and P. Petrica, "A Hybrid Classical Approach to a Fixed-Charged Transportation Problem," in *International Conference on Hybrid Artificial Intelligence Systems*, 2012, vol. 7208 LNAI, no. PART 1, pp. 557–566, doi: 10.1007/978-3-642-28942-2_50.

[7] D. T. H. Giang and L. Sui Pheng, "Role of construction in economic development: Review of key concepts in the past 40 years," *Habitat Int.*, vol. 35, no. 1, pp. 118–125, Jan. 2011, doi: 10.1016/J.HABITATINT.2010.06.003.

[8] McKinsey&Company, "REINVENTING CONSTRUCTION: A ROUTE TO HIGHER PRODUCTIVITY," Feb. 2017.

[9] C. N. Egwim, H. Alaka, L. O. Toriola-Coker, H. Balogun, and F. Sunmola, "Applied artificial intelligence for predicting construction projects delay," *Mach. Learn. with Appl.*, vol. 6, p. 100166, Dec. 2021, doi: 10.1016/J.MLWA.2021.100166.

[10] P. Kumar Donepudi, A. Al Ayub Ahmed, C. Author, and S. Saha--EMERGING MARKET, "Emerging Market Economy (EME) and Artificial Intelligence (AI): Consequences for the Future of Jobs," *archives.palarch.nl*, vol. 17, no. 6, pp. 5562–5574, 2020.

[11] S. Shirowzhan, W. Tan, and S. M. E. Sepasgozar, "Digital twin and CyberGIS for improving connectivity and measuring the impact of infrastructure construction planning in smart cities," *Int. J. Geo-Information*, pp. 1–11, Apr. 2020, doi: 10.3390/ijgi9040240.

[12] M. S. Stojči´c, E. Kazimieras Zavadskas, D. Pamučar, Ž. S. Stevi´c, and A. Mardani, "Application of MCDM methods in sustainability engineering: A literature review 2008–2018," *mdpi.com*, pp. 1–24, Mar. 2019, doi: 10.3390/sym11030350.

[13] A. K. Feroz, H. Zo, and A. Chiravuri, "Digital Transformation and Environmental Sustainability: A Review and Research Agenda," *Sustain. 2021, Vol. 13, Page 1530*, vol. 13, no. 3, p. 1530, Feb. 2021, doi: 10.3390/SU13031530.

[14] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification Supplementary Online Material," *Comput. Vis. Pattern Recognit. (CVPR), 2012 IEEE Conf.*, no. February, pp. 3642–3649, 2012.

[15] M. A. Goralski and T. K. Tan, "Artificial intelligence and sustainable development," *Int. J. Manag. Educ.*, vol. 18, no. 1, p. 100330, Mar. 2020, doi: 10.1016/J.IJME.2019.100330.

[16] D. Pamučar, D. Božanić, … V. L.-F. universitatis, and U. 2018, "Normalized weighted geometric bonferroni mean operator of interval rough numbers–application in interval rough dematel-copras model," *casopisi.junis.ni.ac.rs*, vol. 16, no. 2, pp. 171–191, Jun. 2018, doi: 10.22190/FUME180503018P.

[17] A. Taeihagh, "Governance of artificial intelligence," *https://doi.org/10.1080/14494035.2021.1928377*, vol. 40, no. 2, pp. 137–157, 2021, doi: 10.1080/14494035.2021.1928377.

[18] B. Adam, I. S.-C. & Structures, and U. 2008, "Active tensegrity: A control framework for an adaptive civil-engineering structure," *Elsevier*, vol. 86, pp. 2215–2223, 2008.

[19] V. Singh, S. Bano, | Anand, K. Yadav, S. Ahmad, and A. K. Yadav, "Feasibility of Artificial Neural Network in Civil Engineering International Journal of Trend in Scientific Research and Development (IJTSRD) Feasibility of Artificial Neural Network in Civil Engineering the Creative Commons Attribution License (CC BY 4.0)," *Int. J. Trend Sci. Res. Dev.*, vol. 3, no. 3, pp. 724–728, Apr. 2019.

[20] P. Thompson, "Teaching sustainability in civil engineering using Ceequal," in *http://dx.doi.org/10.1680/ensu.2010.163.4.209*, May 2015, vol. 163, no. 4, pp. 209–217, doi: 10.1680/ENSU.2010.163.4.209.

[21] E. A. Obonyo, "An agent-based intelligent virtual learning environment for construction management," *Constr. Innov.*, vol. 11, no. 2, pp. 142–160, 2011, doi: 10.1108/14714171111124130/FULL/XML.

[22] H. Adeli, "Neural Networks in Civil Engineering: 1989–2000," *Comput. Civ. Infrastruct. Eng.*, vol. 16, no. 2, pp. 126–142, Mar. 2001, doi: 10.1111/0885-9507.00219.

[23] Hayes-RothFrederick, "Review of 'Adaptation in Natural and Artificial Systems by John H. Holland', The U. of Michigan Press, 1975," *ACM SIGART Bull.*, no. 53, pp. 15–15, Aug. 1975, doi: 10.1145/1216504.1216510.

[24] C. Q. X. Poh, C. U. Ubeynarayana, and Y. M. Goh, "Safety leading indicators for construction sites: A machine learning approach," *Autom. Constr.*, vol. 93, pp. 375–386, Sep. 2018, doi: 10.1016/J.AUTCON.2018.03.022.

[25] A. Senouci and H. R. Al-Derham, "Genetic algorithm-based multi-objective model for scheduling of linear construction projects," *Adv. Eng. Softw.*, vol. 39, no. 12, pp. 1023–1028, Dec. 2008, doi: 10.1016/J.ADVENGSOFT.2007.08.002.

[26] J. D. Farmer, N. H. Packard, and A. S. Perelson, "THE IMMUNE SYSTEM, ADAPTATION, AND MACHINE LEARNING," pp. 187–204, 1986.

[27] A. Khalafallah and M. Abdel-Raheem, "Electimize: New Evolutionary Algorithm for Optimization with Application in Construction Engineering," *J. Comput. Civ. Eng.*, vol. 25, no. 3, pp. 192–201, Jul. 2010, doi: 10.1061/(ASCE)CP.1943-5487.0000080.

[28] T. Wang, G. Wang, and K. Liu, "Simulation control of concrete pump truck boom based on PSO and adaptive robust PD," in *2009 Chinese Control and Decision Conference, CCDC 2009*, Jun. 2009, pp. 960–963, doi: 10.1109/CCDC.2009.5192810.

[29] K. Ghaedi, Z. Ibrahim, H. Adeli, and A. Javanmardi, "Invited Review: Recent developments in vibration control of building and bridge structures," *J. Vibroengineering*, vol. 19, no. 5, pp. 3564–3580, Aug. 2017, doi: 10.21595/JVE.2017.18900.

[30] Y. Filiberto, R. Bello, Y. Caballero, and R. Larrua, "Using PSO and RST to Predict the Resistant Capacity of Connections in Composite Structures," *Stud. Comput. Intell.*, vol. 284, pp. 359–370, 2010, doi: 10.1007/978-3-642-12538-6_30.

[31] X. Zheng and Z. Liu, "The schedule control of engineering project based on particle swarm algorithm," *2010 2nd Int. Conf. Commun. Syst. Networks Appl. ICCSNA 2010*, vol. 1, pp. 184–187, 2010, doi: 10.1109/ICCSNA.2010.5588681.

[32] A. Nejadfard, H. Moradi, and M. N. Ahmadabadi, "A multi-robot system for dome inspection and maintenance: Concept and stability analysis," in *International Conference on Robotics and Biomimetics, ROBIO 2011*, Dec. 2011, pp. 853–858, doi: 10.1109/ROBIO.2011.6181394.

[33] V. Raju, D. Maheswari, and S. K. Patnaik, "Active vibration control of piezo actuated cantilever beam using PSO," in *Students' Conference on Electrical, Electronics and Computer Science: Innovation for Humanity, SCEECS 2012*, Mar. 2012, p. 289, doi: 10.1109/SCEECS.2012.6184834.

[34] A. Kaveh and S. Talatahari, "An improved ant colony optimization for constrained engineering design problems," *Eng. Comput. (Swansea, Wales)*, vol. 27, no. 1, pp. 155–182, Jan. 2010, doi: 10.1108/02644401011008577/FULL/XML.

[35] A. Bilgil and H. Altun, "Investigation of flow resistance in smooth open channels using artificial neural networks," *Flow Meas. Instrum.*, vol. 19, no. 6, pp. 404–408, Dec. 2008, doi: 10.1016/J.FLOWMEASINST.2008.07.001.

[36] J. Simon Laflamme, J. J. Connor, and S. Laflamme, "Application of self-tuning Gaussian networks for control of civil structures equipped with magnetorheological dampers," *https://doi.org/10.1117/12.815540*, vol. 7288, no. 6, pp. 271–282, Apr. 2009, doi: 10.1117/12.815540.

[37] T. L. Lee, H. M. Lin, and Y. P. Lu, "Assessment of highway slope failure using neural networks," *J. Zhejiang Univ. A 2009 101*, vol. 10, no. 1, pp. 101–108, Jan. 2009, doi: 10.1631/JZUS.A0820265.

[38] C. Wang *et al.*, "Application of neural network in the cost estimation of highway engineering," *J. Comput.*, vol. 5, no. 11, pp. 1762–1766, Nov. 2010, doi: 10.4304/jcp.5.11.1621-1628.

[39] S. N. Alacali, B. Akba, and B. Doran, "Prediction of lateral confinement coefficient in reinforced concrete columns using neural network simulation," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2645–2655, Mar. 2011, doi: 10.1016/J.ASOC.2010.10.013.

[40] M. Naeej, M. Bali, M. R. Naeej, and J. V. Amiri, "Prediction of lateral confinement coefficient in reinforced concrete columns using M5′ machine learning method," *KSCE J. Civ. Eng. 2013 177*, vol. 17, no. 7, pp. 1714–1719, Oct. 2013, doi: 10.1007/S12205-013-0214-3.

AUTHOR

Dr Engr. TORIOLA-COKER holds higher national diploma HND (Civil Engineering), Master in Engineering MEng (Civil Engineering), Doctor of Philosophy PhD (Construction and Project Management) and professional certificates of Registered Engineer of Council for the Regulation of Engineering in Nigeria (COREN), Member of Nigeria Society of Engineer (NSE), Member of Environment Division of Nigeria Society of Engineer (NSE), Member of Construction Skills Certification Scheme (CSCS) UK, Track Certificate Holder on Rail Network UK and Sentinel Card Holder on Rail Network UK. He lectures at Yaba College of Technology and served as Class Adviser and Coordinator in the department of Civil Engineering. Dr Engr. TORIOLA-COKER is actively involved in academic and professional aspects of Transportation, Highway, Railway and Construction Engineering. Apart from these academic and professional engagements he has been involved in current projects in waste management under the auspices of Innovative UK. Dr Engr. TORIOLA-COKER has supervised student projects at undergraduate and graduate level in the core aspect of Civil Engineering among others and has contributed to articles in academic journals and conferences at local and international levels. He has also published & contributed to academic books.

AN INTELLIGENT MOBILE APPLICATION FOR CAR LICENSE PLATE DETECTION AND ANALYSIS USING MACHINE LEARNING ALGORITHM

Mengcheng Han[1] and Yu Sun[2]

[1]Santa Margarita Catholic High School, 22062 Antonio Pkwy,
Rancho Santa Margarita, CA 92688
[2]California State Polytechnic University,
Pomona, CA, 91768, Irvine, CA 92620

ABSTRACT

How did image recognition and object analysis function to bring convenience to people's lives? Within this question I bear in my mind, I started to explore and build this Automatic Car License Plate Detect and Analyze Project. Since cameras are being widely used for recording and analyzing vehicle information, it has been a great cost to buy such intelligent devices. Guided by recent research on machine learning approaches , we solve this financial problem by designing and implementing a mobile phone application that automatically utilizes the camera installed on the phone to analyze the information of the car license plate. Our design is built to provide users with an accessible way to analyze license plates in complex environments.

KEYWORDS

Machine learning, LPR, OCR libraries.

1. INTRODUCTION

The License Plate Recognition (LPR) System is an important study of public transportation in modern society [6][7]. The LPR System is used in many fields such as traffic law enforcement, toll road payment, community security, in the long decades of study and experiments. It benefits society by substituting a learning-based system to do the work used to be done by the human labor force. For example, a security guard would need to be present at a parking lot to record guest vehicles' information and time parked to tax. However, the LPR System automatically does those vehicle identifications that used to be done by humans, which have greatly increased the efficiency and accuracy of this job [8]. Although a few techniques have already been developed to fulfill these various situations, a unique camera is necessary to be installed in a fixed position to function properly. So we discovered that it is possible to implement a new LPR system that could adapt into mobile phone cameras based on an object-recognition model. This challenge is revolutionary in that we use another way to accomplish the same task of the traditional LPR System but without any expenditure (as long as a mobile phone with a camera is acquired to run the program).

As said above, an LPR camera is used in most situations for detection, which is a specialized camera that is designed to capture characters on the license of a car in motion [9]. A common feature of them is that they are bought as a unique device, and their price could be $100 to $800

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 75-82, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120406

or more. Compared to a free phone application, these equipment are much more expensive for the task they are to complete - car license detection. On the other hand, the LPR System, though is used nationwide in many different forms, has the same basic technique - license detection, image fragmentation, and character recognition. However, the accuracy can be interrupted under extreme environments because the algorithm produces results based on the similarity of pixels. For example, the LPR camera might be blurred or covered in weather like rain or snow, and the results of detection are not desirable anymore for either government use or business. In fact, the first LPR System was implemented in the 20th century which means the same technology has been used for more than 40 years. Thus, another practical problem of the current LPR System contains a much more complex algorithm (probably the reason for using a specialized camera) than a trained machine learning model, which also solves the task efficiently with a simpler approach [3].

In this paper, we follow the same line of research by the LPR System using machine learning. Our approach is based on an automatic object-detection model YOLO-V3 to handle the tasks of detecting car licenses in pictures or videos. After training the machine learning model with datasets, it recognizes licenses that have different plat formats in nonuniform outdoor illumination. While the other two parts of the new LPR System in this project are also implemented using machine learning, we built a web server for the LPR System adapted into a phone application. Our goal is to abridge the process of car license recognition, and this step provides the user a simpler way by using a phone application rather than buying an LPR camera, which could be lavish. Mobility is an important feature of this project that the analysis can be done at any location with an internet connection, while the same result could only be resolved by taking pictures at a fixed camera position. Therefore, we believe that this proposed application can be a reformation to the original LPR System and bring benefits to the users.

In the two application scenarios in part 4, we demonstrate how the above solution is carried out and the accuracy of the algorithm we implemented. First, we design an experiment to test different machine learning models for their accuracy in detecting the license plate in the image [4]. In fact, Yolo-V3 is the model that provides a better result of detecting US license plates within 300 test images. while comparing to Yolo-V4 which has only 50% accuracy. Second, we did another experiment regarding different libraries that contain specific functions to recognize text in the image. The results indicate that Easy-OCR is the one that has the highest accuracy of 88.26% with 50 test images. Although the percentage isn't perfect among these experiments, it demonstrates our project could provide usefulness to the app users with its mobility and compatibility. Further modification to the algorithm and libraries is possible to be implemented to enhance the accuracy of the analysis.

To conclude, the rest of the paper is organized as follows: Section 2 gives concrete details on challenges within this project such as picking a specific OCR library, implementing camera function inside phone application, and fixing code compatibility. Section 3 focuses on the details of our methodology to solve the challenges mentioned in Section 2. The solution walks through the entire process of code implementation and compiled together into one LPR System; Section 4 presents the experimental data that was produced during the implementation, followed by the related document on this technology from different scholars; Section 6 gives the conclusion and possible future work that is yet to be done for this project.

2. CHALLENGES

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

2.1. Picking A Specific Image Recognition and OCR Library is Difficult

Choosing a properly working and high accuracy OCR library is the core part of this type of project. For instance, there are specific issues within the Pytesseract Library that we could not implement the trained model on Google Colab. On another hand, we also tried KerasOCR Library which has low accuracy in recognizing the characters on the license plates. Having a high accuracy model is mandatory for this type of user experience. Thus, I picked EasyOCR Library as the final solution to this challenge. It is perfectly compatible with Google Colab and has 95% accuracy on overall detections. Moreover, the algorithm being used within this library is designed better than the previous two so much so that it took less processing time to do more analysis.

2.2. Installing and Implementing Camera Functionality for our App can Potentially Pose A Problem

Enabling the camera function is necessary for the mobile phone application to take pictures of car license plates. But getting permission to use the camera installed on the phone physically and testing this function on a virtual phone open in a computer is complicated. We specifically need to allow not only photos to be taken, but also access to the camera roll on most phones. As a result, the camera and mobile app are implemented with Flutter using Dart as the coding language as they have pre-built libraries for this solution. Also, setting up the camera in the virtual testing machine. requires some modification on the CPU of the computer through BIOS since it is tested on a virtual phone model instead of an actual phone. So I switched the setting of the CPU on my laptop and finally got the camera to work.

2.3. Due to the Multi-Faceted Nature of the Project, Integrating Multiple Libraries in Python can Lead to General Problems

It is difficult to adapt different libraries into this image-recognition project, especially using python and dart as main languages because I am more confident coding with java [11]. For instance, libraries that contain machine learning algorithms such as Pytesseract update their syntax periodically. So it is hard for programmers like me to implement functions with those syntaxes changing over time. By looking up relative resources, a variety of syntax errors are solved through debugging. It takes time and patience to go through each line of code to find out those “off by one” errors. Moreover, libraries' syntax changing over time also slows the process development of this project. With 6 months of effort in debugging, I was able to run through the whole project without any syntax error.

3. SOLUTION

This project is built based on YOLO V3 (You Only Look Once), which is a specific machine learning algorithm that is very efficient at analyzing pictures in real-time [10]. We used this advantage to train its function of recognizing license plates based on its function of detecting cars in the pictures. On the other hand, the phone application functions to bring the entire project together as it allows for ease of access to the research and program we have created. “What’s My License”, the APP written for this project, is implemented via Android Studio using the programming language Dart.

Once the object is detected from the picture taken from the mobile application, the program will send the image data through a web-based Python flask server to develop a machine-learning model [12]. Then, by utilizing the cropping function to separate the character-contained license plate from the rest of the image, we used tools implemented in Easy-OCR(Object Character

Recognition) library to analyze the picture. This library looks into the similarity of characters that need to be recognized in the picture and returns texts that are successfully detected and the confidence. Finally, the program would send the license plate information back to the user's phone through the Python Flask Server and display it on a separate application page.

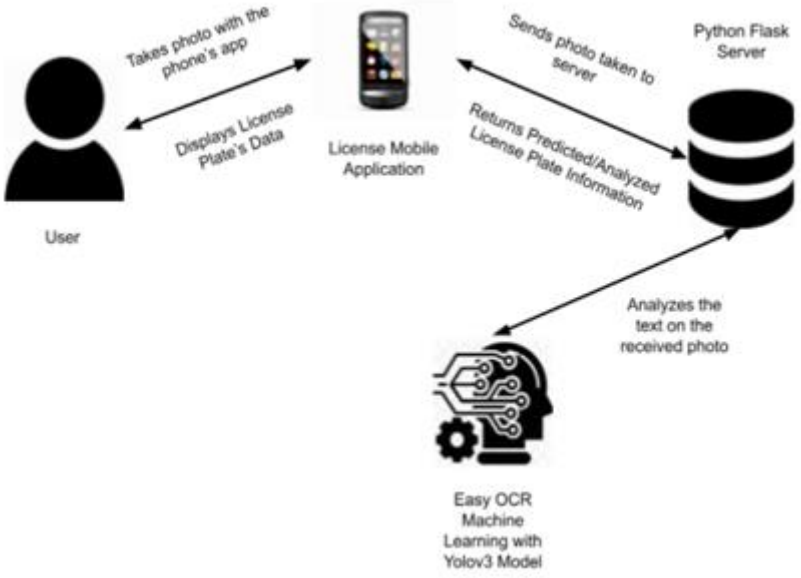


Figure 1. Overview of the APP

The APP that users download and install from google PlayStation / Apple App store is a Flatter application written in the programming language DART. By using dart, we can customize the theme colors, UI pages design, and App logos to produce a more efficient phone APP. The app "What's my license" uses the "Camera" tool to access the camera function on the phone to take desired images. It is adapted to mobile platforms to have easier access and experience of using this project in some real cases.

The user can take a picture of the license plate by clicking on the button, and the image data would be sent back to this project's local storage through a Web-Request of Python Flask Server. The Python Flask Server is a library that allows functions to turn locally running code into a server application that is accessible via HTTP request [13]. Thus, the image taken by the application is sent to the server through our web API, then the picture can be analyzed by the machine learning model by just calling the HTTP request on the server [5]. Once the analysis is done, the Python Flask Server would send a string response that contains the license number being detected back to the mobile application. After the whole process is done, the app would display the information on a separate screen.

As for how the analysis is actually done, that is done in the Python back-end code that is called through the Flask HTTP request. To begin, the original model is trained using 100s of photos of license plates. This data set is trained on the YOLO-V3 machine learning library/model for object recognition. We chose this specific library because its feature set is useful for license plate detection as it has some partial training with vehicle image detection already. We also utilized Tensorflow while training the model. Both of these in tandem allowed us to fully train the model and isolate license plates within any image. We then saved those weights to be utilized later on. (Something like ... now that this code has isolated the license plate, we move on to extract the text from it). Furthermore, Easy-OCR is another library we use for character/text recognition

after the actual license plate is cropped out of the rest of the image. The function, which is implemented for this project, matches similarities of possible characters in the picture and returns the results in string variable: the model can complete this task by taking in a specific training data set that contains pictures already analyzed and determine new data results by the machine learning algorithm.

To conclude, this LP-Detecting project was achieved by a mobile phone application that can take image data. By implementing a Python Flask Server, this application transfers images to be analyzed by a license-recognition HTTP request and displays the detected information back to the users.

4. EXPERIMENTS

4.1. Experiment 1

Data (Accuracy) Yolov3/Yolov4
Training on 100 images (US) - 61.22%
Training on 300 images (US) - 81.54%
Training on one type of license plate but detecting multiple - 54.70%
Training on multiple state license plates and European license plates - 72.31%

Experiment 1 is designed to test different machine learning models' efficiency in detecting different types of license plates. By implementing a generator using Python, the model can process a large number of test images. The accuracy of the analysis out of this experiment is more accurate than doing tests with single or multiple images manually. On the other hand, Yolov3 provides us the best result of an 81.54% accuracy in the training on 300 images that contain US license plates. In fact, images that failed to be analyzed by Yolov3 mostly have low quality because the learning algorithm cannot function properly while the picture is too blurry.

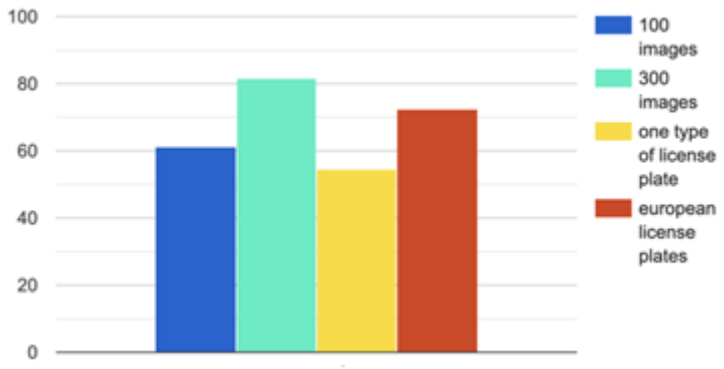


Figure 2. Result of experiment 1

4.2. Experiment 2

Experiment 2: Which machine Learning Model is most efficient for text recognition?
Data
TensorFlow Keras - 82.33%
Pytesseract - 79.01%
EasyOCR - 88.26%

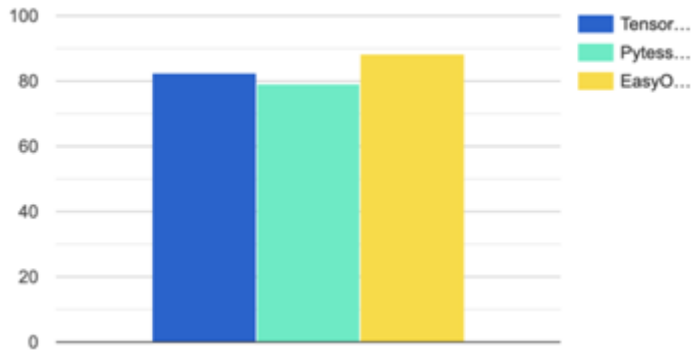


Figure 3. Result of experiment 2

Experiment 2 is to test several machine learning models running text-recognition algorithms on images that are already detected to have a license plate. These results do not account for highlighting more than required, but simply the accuracy of the text read on the screen. Out of the three libraries being tested, EasyOCR has the highest percentage among the other two, which is also the reason we choose to use this library in our project. Although it recognizes text with 88.26% accuracy, special signs and other language characters are not included.

Based on both experiments we have done above, we believe our solution has both stable performance and high accuracy.

5. RELATED WORK

This research paper implements a similar project to ours as it is focused on License Plate Detection. Its functionality is adopted to detect characters and license plates from different angles and different regions around the world. The researchers implemented modifications to the YOLOv2 networks in the paper, which is similar to the YOLO V3 model being trained in our project. However, theirs is used for locating the license plate in various scenarios. The testing data being used in the paper is focused on extreme viewing angles and regional designs, while our project emphasizes the mobility of algorithms utilized through a mobile application. Most users take the picture by phone camera that would most likely have a front-facing view of the license plate. Overall, both of the models use some sort of YOLO machine learning implementation but are focused on different goals.

In this research paper, the author concentrated on improving the processing speed and detection accuracy of their license plate recognition model [1]. They used an image downscaling method to achieve their goal as well as a method to efficiently limit what parts of the image had to be processed. Their modifications to this as well as the classifier resulted in a significantly faster and more accurate model than most standard implementations. One of the approaches of this research paper is similar to one of the python libraries in my project - the Pytesseract library. They both degrade the quality of the picture to a level that could accelerate the detection without sacrificing accuracy. This downgrade of the photo also is utilized to decrease any noise that may unnecessarily confuse the model. Ultimately, the focus of this research paper does not directly correlate to the goal of my project in that our work is focused more so on implementing machine learning via a mobile application, not simply trying to find the fastest prediction possible.

The article discusses an automatic LP-detect system using a Convolution Neural Network to train the model [2]. The system provides results with a great percentage of accuracy under complex environments and distorts perspectives. There are similarities between the Deep Learning System

implemented by the author and mine in that both of our goals are to analyze the license plates image into useful information. In addition, “What’s My License” is also tested for performance under extreme prerequisites such as low pixel, customized license plate. Perhaps all of these projects did a great job on data-collecting and analyzing, but our project is adapted to a mobile application to enhance its usefulness. Installing the APP grants people an easier way for LP detecting purposes in real-life scenarios instead of running source code on PC.

6. CONCLUSION

In this project, we designed a mobile application incorporating multiple services through Google-Colab to achieve the very same goal of the LPR System, which is already implied in many fields [14]. The application we proposed can be split into three parts: analysis, server, and application. First, detection and recognition are done in the analysis part; The trained machine learning model Yolov3 scan through the image that contains license plate and crops the image into a rectangle to separate the license plate from complex pixels; The function implemented by tools in the EasyOCR library is then used to recognize the text on the license plate and return the information as a string variable. Furthermore, we set up a Python-Flask server to allow the users to upload their own images and get results from the mobile application through HTTP requests. Last but not least, we use android studio and the programming language DART to implement a phone app that is more customizable than other applications, so the other two parts of this project can be fused into one without any compatibility error [15]. Throughout the entire development process, we also did a lot of experiments to ensure our approach is efficient; For example, we test different machine learning models to see which of them have a higher accuracy of detection within hundreds of sample pictures. The results indicate the effectiveness of our design that provides not only high accuracy analysis but also higher mobility than the original LPR System.

There are possible limitations in this phone application that is yet to be solved with more advanced technology, such as accuracy of the analysis could flat for a low-quality picture; there must be a stable internet connection for the transportation of information between the host and server; the function we currently developed can only analyze photos, so optimization on file compatibility like videos would provide more convenience to users.

To enhance the accuracy of analysis, we can try to implement other OCR libraries since different algorithms within each of them can produce different results. Moreover, by optimizing the camera function in the phone application to upload video files, users can have a better experience while using our application.

REFERENCES

[1] Z. Selmi, M. Ben Halima and A. M. Alimi, "Deep Learning System for Automatic License Plate Detection and Recognition," 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 2017, pp. 1132-1138, doi: 10.1109/ICDAR.2017.187.

[2] Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu and N. Komodakis, "A Robust and Efficient Approach to License Plate Detection," in IEEE Transactions on Image Processing, vol. 26, no. 3, pp. 1102-1114, March 2017, doi: 10.1109/TIP.2016.2631901.

[3] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. "Machine learning basics." Deep learning 1.7 (2016): 98-164.

[4] Jordan, Michael I., and Tom M. Mitchell. "Machine learning: Trends, perspectives, and prospects." Science 349.6245 (2015): 255-260.

[5] Zhang, Xian-Da. "Machine learning." A Matrix Algebra Approach to Artificial Intelligence. Springer, Singapore, 2020. 223-440.

[6] Vuchic, Vukan R. "Urban public transportation systems." University of Pennsylvania, Philadelphia, PA, USA 5 (2002): 2532-2558.

[7] Pucher, John. "Public transportation." (2004).
[8] Ozbay, Serkan, and Ergun Ercelebi. "Automatic vehicle identification by plate recognition." World Academy of Science, Engineering and Technology 9.41 (2005): 222-225.
[9] Niknam, Mehdi, and Parimala Thulasiraman. "LPR: A bio-inspired intelligent learning path recommendation system based on meaningful learning theory." Education and Information Technologies 25.5 (2020): 3797-3819.
[10] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[11] Van Rossum, Guido, and Fred L. Drake Jr. Python tutorial. Vol. 620. Amsterdam: Centrum voor Wiskundeen Informatica, 1995.
[12] De Smedt, Tom, and Walter Daelemans. "Pattern for python." The Journal of Machine Learning Research 13.1 (2012): 2063-2067.
[13] Lutz, Mark. Programming python. " O'Reilly Media, Inc.", 2001.
[14] Niknam, Mehdi, and Parimala Thulasiraman. "LPR: A bio-inspired intelligent learning path recommendation system based on meaningful learning theory." Education and Information Technologies 25.5 (2020): 3797-3819.
[15] Godefroid, Patrice, Nils Klarlund, and Koushik Sen. "DART: Directed automated random testing." Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation. 2005.

AN SIMULATION AND SURVIVAL BASED MOBILE GAME FOR PLAYERS DEVELOPING A SENSE OF ENVIRONMENTAL PRESERVATION USING MODELING AND MANUAL MANIPULATION PROGRAMMING

Yiwen Liu[1] and Yu Sun[2]

[1]The high school Affiliated to Renmin University, Beijing,
Haidian, Zhong guan cun street, 37
[2]California State Polytechnic University,
Pomona, CA, 91768, Irvine, CA 92620

ABSTRACT

The citizens nowadays are often born and raised in well developed urban areas and have rarely, even never experienced the difficulties that wildlife are suffering from due to human actions. They are likely to possess sympathy, but never do these individuals are aware of how their deeds may affect the living of other species on planet Earth [1]. However, if someone reveals the bloody truth to the people, they are likely to change for the greater good.

In this paper, we mainly used unity modeling and java programming skills to develop an animal simulation game in order to show the damage done by mankind and resonate the feeling of empathy so the players may alter their actions to preserve the environment [2][3]. The player starts the game as an animal figure in a randomly generated map. The player will control the animal to move around to consume water and food for survival. Meanwhile, the animal must avoid the invasion of human poison lands closing in on the habitats for this figure. Eventually, the player will starve or be poisoned and fail to survive. By setting this result, we hope to arouse the sympathy in hearts and lead to some alteration to a person's habits.

KEYWORDS

Simulator, self-manipulation, protection, ecosystem.

1. INTRODUCTION

This simulation game is developed mainly to raise the alertness for environmental protection of individuals, also leading to changes in lifestyle to benefit nature. People living in the wealthy areas of the world has never been to the developing areas where humans have to invade the territory of wild beasts, driving them to further lands or maybe blind alleys. However, this damages the ecological balance significantly because the distinction rate of endangered animal species is higher than ever [4]. This topic has been widely discussed ever since humans entered the era of technology, people are struggling to find the balance of sustainable development in life to live in peace with other species and maintain an appropriate biodiversity [5][6][7]. This game has a significant benefit when compared with other products or slogans in society nowadays. This game can provide a first person perspective, an immersed experience for individuals that have

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 83-92, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120407

little chance to actually feel the negative effects we have done to the nature kingdom. The commercials, slogans and products in the market currently cannot let the people directly soak themselves in the situation of wild lives. Consequently, while playing the game, the character that that player represent will slowly fall into the situation of lacking food, water and shelter, as the player helplessly watch the hp figure decreases to 0, they will have a similar emotion with animals that actually strives to live under this state. By receiving this sense of substitution, the individuals are more likely to make actual adjustments to life.

Some of Unity techniques and systems that have been proposed to model the creature figures and the map used in our simulation game, which allows the user to operate the character freely by the keyboard. Also, Java scripts are written to allow the health bar and poison bar to elevate and drop, the camera to always follow the player and be able to rotate according to the movement of the mouse, the resource area to randomly generate and etc. However, these proposals assume deliberate methods of modeling in Unity, which is rarely the case in practice. Their implementations are also limited in functions, with samples given for the inability for C# code itself to assort a tatted map that meets the demands of containing all kind of landscapes and has precise height demands [8]. Other techniques, such as animal building, may not be accurately done in Unity, in the “create object” options, there are mostly basic shapes like sphere and cube to choose, but we wish for the player to recognize the animal at first sight for better game experience and a deeper substitute into reality. Consequently, a new application is put into practice for character establishment. Because Maya has a high degree of finesse, when modeling, we are able to access the angle and shape freely and fabricate an ideal figure. Therefore the method can grow more sophisticated and results in hypothetical results. A new issue comes up as Maya is actually applied, the flips and twists of models are easily misunderstood and we often fail to solve the math involved in this process. A second practical problem is that some users find it hard to understand the rules and goal of the game, some players may accuse the game for not owing the method to survive and win, questioning the play-ability of the simulator.

In this paper, in order to make reasonable to answer the previously mentioned queries, we developed some methods and completed researches. Our goal is to establish a mature, playable simulation game that has a high degree of liberty. Our method is inspired by the asp codes and the WWF game of environmental protection. There are some good features of ASP code. First, with ASP, we are able to create more interactive and data-driven web applications. ASP consists a variety of control like text boxes, buttons, and labels for assembling, configuring, and manipulating the code to create certain pages. Second, since we only employed ASP in the limitations and arrangements of the map, which only require the basic lines of ASP, this kind of code is straight forward written and is therefore more easier to understand. Third, ASP allows the map to grow to a significant size in a short amount of time, if we only used the loop in C#, it may take too long to operate the code. The ASP code is flexible and swift when coding. Therefore, we believe that the usage of this method can help us avoid most of the coding issues. Moreover, our game is inspired initially by the WWF game, we learned from it’s strong points to offset our weakness. By learning to add some introduction to the end of the game and adding more interesting to our game, we can effectively avert the misunderstanding and dissatisfaction towards our simulator.

In two application scenarios, we demonstrate how the above combination of techniques increases the play-ability of the simulator. First, we show the usefulness of our approach by a comprehensive trial of the game. We play the role of a user and start from the initial page of the game. Beginning with a site that shows the options of "play", "instructions", and "menu". As the player chooses "play", the scene will shift to a randomly organized map containing all the landscapes, like water, food sand and so on. After entering, we will test the movement and the functions of each scale, making sure that food number rises when the animal stays in the resource

area and hp drops when the growth of the poison area outruns the player's speed. After playing throughout the whole procedure, we have an initial plan of aspects that needs to vary and other functions that can stay the same. Second, we analyze the evolution of our java code and debug every error that shows up, both in C# and in Unity. Sometimes we ignore a bracket or a semicolon that is required in the language of coding Java scripts. Other times we may forget to add a sphere collider on to the player or the resource area. According to the reminder panel in Visual studio and the error window in Unity, we are able to locate the line number of the error and the kind of mistake we have made, after finding it, we have to check the line itself and lines around it abortively to seek the solve. After rigid identification of the exact word or symbol, the debugging process is quiet simple. After making sure no mistakes occur while running, we can ensure the reliability of the code.

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, both in the coding process and in the functioning of Unity; Section 3 focuses on the details of our solutions corresponding to the challenges that we mentioned in Section 2, referring to the ASP code mapping technique and the additional supplement page in the end of game; Section 4 presents the relevant details about the trials we did, by putting the simulator on probation with various individuals playing a trial and giving feedback on their experiences and opinions, 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, which is closely related to spreading the game and more importantly the idea of preservation for the wilderness.

2. CHALLENGES

While building the game, I encountered several challenges. But with effort to solve the issues and flexibility to find other ways, I managed to polish Apocalypse the simulator into a mature game successfully delivering my concept of wildlife protection.

2.1. Game Ending

The fundamental one is how the game ends each level. I would like to make one round of the game short to cater to the rapid lifestyle of modern people, while strongly delivering my idea of environmental protection in reality [10]. The ideas I came up with were just not perfect, I thought of letting the animal character defeating the human invaders, but that is just very unrealistic and would not arouse much empathy in the players, I though about calculating the times reaching resources areas in a limited time for highest score, but it is weak in telling the idea. One night watching a movie, I cried as the leading roles die, giving the movie a tragic ending, and something hit me. Tragedy and death is often the most powerful weapon for emotions, if I created my game with a never winning mode, letting the animal character never surviving, much controversy would be raised and attention will be set on to the problem of ecosystem reservation. In the end, Apocalypse turned into a game of which the player will eventually have no way to live, longest surviving time would be kept as high score and two warning sheets will pop up after the player loses. This can achieve both my goals of short game and direct communication of thoughts.

2.2. Map Building

Another major problem that has been encountered is the is the map building, if the original C# codes were used in the script, the map would take minutes to respond even if it is already built when we were developing the game. This means that the player would be stuck in the loading

interface for a long time, leading to a misunderstanding of bugs appearing in the build and therefore losing interest in Apocalypse the simulator, even spreading a bad reputation for it. I solved this problem by obtaining a different method of coding, ASP codes. ASP or Active Server Page, requires a client-use of Java script, it is especially using when a user is setting some limitations or conditions to his or her page, it can be easily managed and understood by the computer. ASP codes I used in my code appears in paragraph-like contents, which allows me to consider the restriction input separately. Also, ASP codes reduces times greatly, showing this advantage greatly in large projects like Apocalypse. With the help of this technology, I solved the issue and the maps now can be generated within seconds on the player's end.

2.3. Character Choosing

The third issue I met is selecting the representative animal that is used in the game. I wish to use an animal that is well known around the world and is not a pet creature, also adapting the grassland and forest junction environment I generated in the map. At first it was hard to decide because so many animals own a high reputation and is favored by most people, like the pandas or the penguins, but selecting an animal meeting all demands are hard. After discussion with my school teacher, we worked out using the deer as the symbol and character of our game because it is not only known and loved, but also living in a unique habitat between the boundaries of the grassland and forest.

Meeting problems on the way of processing a project is inevitable, but using different methods or researching for extra information can always let one solve an issue in no time.

3. SOLUTION

Some of Unity techniques and systems that have been proposed to model the creature figures and the map used in our simulation game, which allows the user to operate the character freely by the keyboard. Unity is a 2D/3D engine and framework that gives you a system for designing game or app scenes for 2D, 2.5D and 3D. Unity allows us to interact with them via not only code, but also visual components, and export them to every major mobile platform and a whole lot more for free. Also, Java scripts are written to allow the health bar and poison bar to elevate and drop, the camera to always follow the player and be able to rotate according to the movement of the mouse, the resource area to randomly generate and etc. Unity is a native C++-based game engine. We can write code in C#, JavaScript (UnityScript) or, less frequently, Boo. Much of the power of Unity is in its rich scripting language, C#. We can use it to handle user input, manipulate objects in the scene, detect collisions, spawn new GameObjects and cast directional rays around the scene to help with our game logic. However, these proposals assume deliberate methods of modeling in Unity, which is rarely the case in practice. Their implementations are also limited in functions, with samples given for the inability for C# code itself to assort a tatted map that meets the demands of containing all kind of landscapes and has precise height demands. Therefore, we have obtained ASP codes to generate maps fast and accurate according to the area and height demands.

This simulation game is developed mainly to raise the alertness for environmental protection of individuals, also leading to changes in lifestyle to benefit nature. People living in the wealthy areas of the world has never been to the developing areas where humans have to invade the territory of wild beasts, driving them to further lands or maybe blind alleys. However, this damages the ecological balance significantly because the distinction rate of endangered animal species is higher than ever. This topic has been widely discussed ever since humans entered the era of technology, people are struggling to find the balance of sustainable development in life to

live in peace with other species and maintain an appropriate biodiversity. This game has a significant benefit when compared with other products or slogans in society nowadays. This game can provide a first person perspective, an immersed experience for individuals that have little chance to actually feel the negative effects we have done to the nature kingdom. The commercials, slogans and products in the market currently cannot let the people directly soak themselves in the situation of wild lives. Consequently, while playing the game, the character that that player represent will slowly fall into the situation of lacking food, water and shelter, as the player helplessly watch the hp figure decreases to 0, they will have a similar emotion with animals that actually strives to live under this state. By receiving this sense of substitution, the individuals are more likely to make actual adjustments to life.

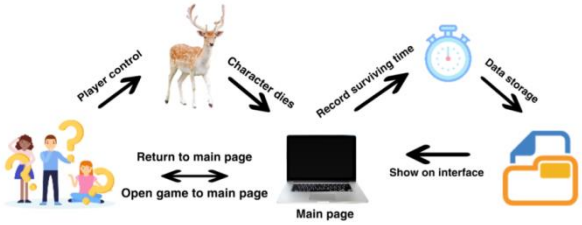


Fig.1 The overview of the solution

To achieve these goals, Apocalypse consists of three main components:

- The database containing all the scores and controls of the user in Unity.
- The controls of the player mode by C# and ASP in Visual studio.
- A interface for visual presentation of the game.

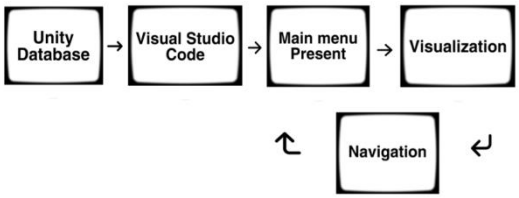


Fig.2 Overview of the components

Some of Unity techniques and systems that have been proposed to model the creature figures and the map used in our simulation game, which allows the user to operate the character freely by the keyboard. Unity is a 2D/3D engine and framework that gives you a system for designing game or app scenes for 2D, 2.5D and 3D. Unity allows us to interact with them via not only code, but also visual components, and export them to every major mobile platform and a whole lot more for free. Also, Java scripts are written to allow the health bar and poison bar to elevate and drop, the camera to always follow the player and be able to rotate according to the movement of the mouse, the resource area to randomly generate and etc.

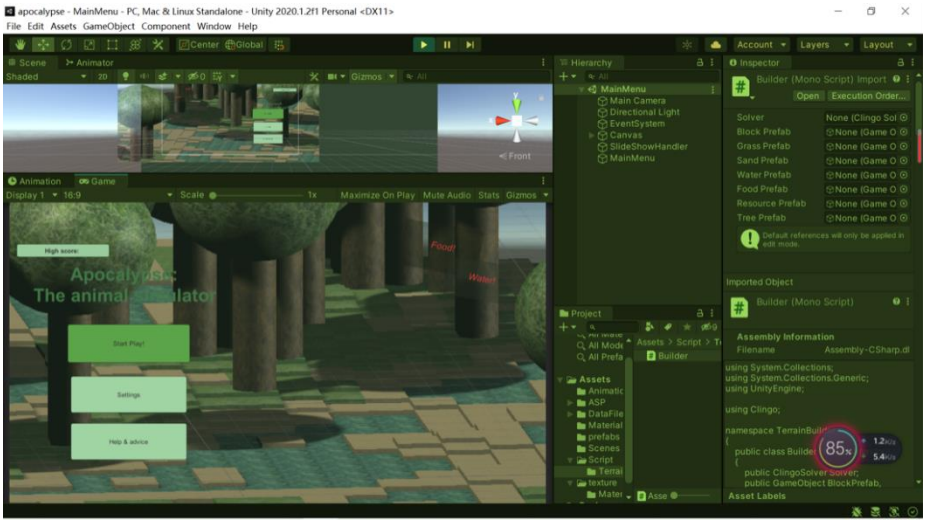


Fig.3 Unity UI

```
    //Shuffle buildQueue to produce more random food placement
    for (int i = 0; i < buildQueue.Count; i += 1)
    {
        int index = Random.Range(0, buildQueue.Count);
        int temp = buildQueue[i];
        buildQueue[i] = buildQueue[index];
        buildQueue[index] = temp;
    }
}

- 引用
void BuildArea(int maxWidth, int minWidth, int maxHeight, int minHeight, int maxDepth, int minDepth)
{
    string boarderCode = GetBoarder(maxWidth, minWidth, maxDepth, minDepth);
    Debug.Log(GetFoodCode(new Vector3(maxWidth, 0, maxDepth), new Vector3(minWidth, 0, minDepth)));
    string foodCode = GetFoodCode(new Vector3(maxWidth, 0, maxDepth), new Vector3(minWidth, 0, minDepth));
    ClingoSolve(aspCode + boarderCode + foodCode, $" -c max_width={maxWidth} -c max_height={maxHeight} -c max
}
- 引用
string GetFoodCode(Vector3 max, Vector3 min)
{
    float minDistance = 30;
    List<GameObject> foodBlocks = FoodCount();
    Vector3 minMax = new Vector3(min.x, 0, max.z);
    Vector3 maxMin = new Vector3(max.x, 0, min.z);
    bool outOfRange = true;
    Debug.Log("FoodCount: " + foodBlocks.Count);

    foreach (GameObject blocks in foodBlocks)
    {
```

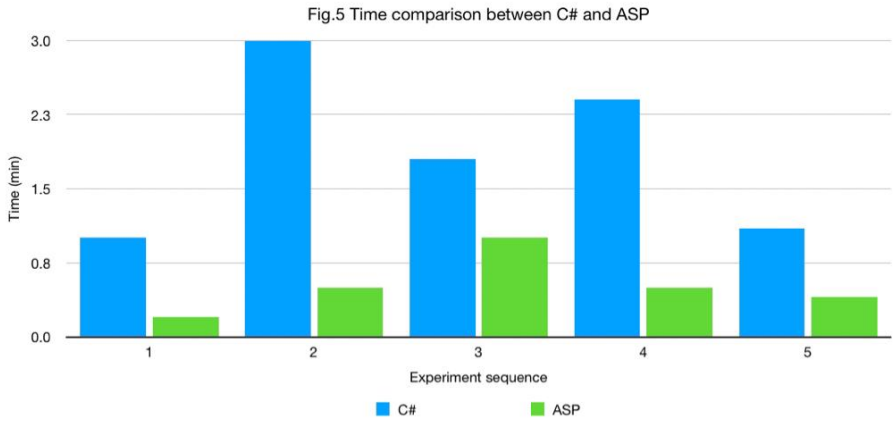
Fig. 4 Visual Studio coding

Unity is a native C# based game engine. We can write code in C#, JavaScript and Boo. Much of the benefit of Unity is in its abundant coding language, C#. We can use it to control player input, alter materials in the scene, spawn some imaginative GameObjects and obtain use of various shapes and color to build characters for the visual presence of the game. However, these proposals request complicated methods of modeling in Unity, which is rarely performed within the control of high school students like me. Their implementations are also restricted in abilities, with samples given for the unable of C# code itself to assort a tatted map that meets the demands of containing all kind of landscapes and has precise height demands. Therefore, we have obtained ASP codes to generate maps fast and accurate according to the area and height demands. Apocalypse obtains the technology of C#, ASP coding and Unity. Unity not only allows us to interact with scripts, but also sight components, and transfers them to every main mobile interface. Also, C#scripts are written to allow the health bar and poison bar to elevate and drop, the camera to always follow the player and be able to rotate according to the movement of the mouse, the resource area to randomly generate and etc.

ASP consists a variety of control like text boxes, buttons, and labels for assembling, configuring, and manipulating the code to create certain pages. Second, since we only employed ASP in the limitations and arrangements of the map, which only require the basic lines of ASP, this kind of code is straight forward written and is therefore more easier to understand. Third, ASP allows the map to grow to a significant size in a short amount of time, if we only used the loop in C#, it may take too long to operate the code. The ASP code is flexible and swift when coding.

4. EXPERIMENT

4.1. Experiment 1



Times	C#	ASP
1	1.0	0.2
2	3.0	0.5
3	1.8	1.0
4	2.4	0.5
5	1.1	0.4

Tab.1 Data of time used

Figure 5. Result of the experiment

Experiment one I designed to come up with a result solving the challenge of map building is that I measured the time used to generate the map using the original C# code and the time used to generate the map using the ASP code and made a comparison, repeated for 4 times. The results are shown above in Tab.1 and Fig.5, it can be concluded that building the map using C# code takes about 1.5-3.0 minutes, while using the ASP code only takes about 0.2-1.0 minutes, a significant decrease in time. Proving that obtaining ASP codes would be the best choice available.

4.2. Experiment 2

To reach the answer of the most suitable animal character used in the game, I have made a questionnaire and dropped it into my friend and family WeChat groups to collect some raw data. The question and conditions are shown below, in Fig.6 .

Fig.6 Questionnaire spread out

After gathering the data, the questionnaire program helped me to sort out the orders by most votes to least, and the best animal to use representing this game is the deer in most people's minds. It received 13 votes out of 58 votes. Deer votes take up approximately 22.4% out of total, tiger votes take up 17.2%, lion votes take up 13.8%, rabbit votes take up 12.1%, panda votes take up 10.3%, leopard votes take up 8.6%, horse votes take up 6.9%, wild buffalo votes take up 5.2%, and zebra votes take up 3.4%. The Results are present in Fig.7 below.

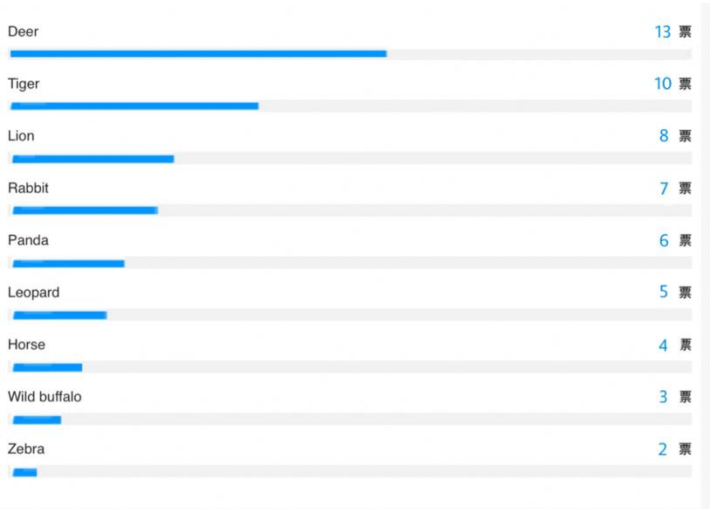


Fig.7 Experiment result

Both experiments have helped me obtain new ways or consider other people's opinions, which provided significant help since the game is based on environmental protection and hopes to arouse the empathy for wild lives, players will have to receive the perfected version of Apocalypse to become a persistent player and spread good reputation.

5. RELATED WORK

There are many animal simulators in the market currently and we chose three products to compare with Apocalypse. These games are: Roblox animal simulator, Wildcraft and Animalia [11]. First, Roblox is a game developed by @ragnar9878. This is a game where the player can choose from a variety of animals to play in the game. This is a bit similar to Apocalypse, however, after the player enters the map, things are very different from our game. The player in Roblox enters a big map with other players and attack with fireballs. Apocalypse surpasses this game by having a better idea of environmental protection because it is a thought of actual improvement for the world, also, Apocalypse has a better feeling of substitution since animals in real life do not fight each other with fireballs. Secondly, WildCraft is a simulator developed

published by Turbo Rocket Games. This game has single-player and multiplayer modes. Players can hunt in different worlds with real weather as different animals and create families. This way of playing is very similar to Apocalypse, even more deliberate. But Apocalypse has the uniqueness of never winning and cannot survive very long, which takes less energy and time meanwhile arousing more controversy and gives people a deeper impression. Lastly, the Animalia simulator developed by High Brazil Studio [12]. This game is released on steam and costs 70 rmb. The builds and fluency of the game is outstanding, but despite the great outlooks and precise plays, the game to me lacks an inner spirit, a theme. As I have mentioned, Apocalypse is unique because it devotes itself letting the players focus on the fact that wild-lives are in great danger and humans must set importance to the problem therefore making a change. The similar, competitive products in the market are as follows: Roblox animal simulator, Animalia survival and WildCraft simulation [13]. To begin with, Roblox simulator uses Roblox Lua coding for the game to run. Roblox Lua scripting is an embedded coding language to add features to your game. It is easy to learn and is a loose interpretation of modern day game programming. Although Roblox Lua is very similar to C#, the C# codes compiles directly to a machine's native code, allowing it to be one of the fastest languages in the world, if optimized; Lua: Powerful, fast, lightweight, embeddable scripting language, but not as efficient as what it developed from. Moving on is the Animalia survival, the builds and code used in this game is varied and exquisite, the only technological or reality issue seems to be the degree of difficulty and the fee spent on it. Finally, comparing the WildCraft simulation to Apocalypse. The wildcraft simulator comes from Turbo Rocket games and uses c#. But the builds run more fluent than Apocalypse. The real life problem is similar to above, finance aid needs are larger to our game.

6. CONCLUSIONS

Environmental protection is the keystone of the project.

The Earth is under great pressure, millions of creatures has lived on this beautiful planet in harmony for centers, creating a saga of life forms [14]. However, as the invasion of human oppresses nature constantly, this balance is on the edge of tipping over. Many people today only have an idea of eco-protection, but makes no change to their life because they do not feel the actual crisis and survival pressure the wild beasts are facing. However, if we do not set importance starting now, we will be facing climate change, soil erosion, poor air quality, and undrinkable water in no time.

This simulation game is developed mainly to raise the alertness for environmental protection of individuals, also leading to changes in lifestyle to benefit nature. People living in the wealthy areas of the world has never been to the developing areas where humans have to invade the territory of wild beasts, driving them to further lands or maybe blind alleys. However, this damages the ecological balance significantly because the distinction rate of endangered animal species is higher than ever. This topic has been widely discussed ever since humans entered the era of technology, people are struggling to find the balance of sustainable development in life to live in peace with other species and maintain an appropriate biodiversity. This game has a significant benefit when compared with other products or slogans in society nowadays. This game can provide a first person perspective, an immersed experience for individuals that have little chance to actually feel the negative effects we have done to the nature kingdom. The commercials, slogans and products in the market currently cannot let the people directly soak themselves in the situation of wild lives. Consequently, while playing the game, the character that that player represent will slowly fall into the situation of lacking food, water and shelter, as the player helplessly watch the hp figure decreases to 0, they will have a similar emotion with animals that actually strives to live under this state. By receiving this sense of substitution, the individuals are more likely to make actual adjustments to life.

Some of Unity techniques and systems that have been proposed to model the creature figures and the map used in our simulation game, which allows the user to operate the character freely by the keyboard. Also, Java scripts are written to allow the health bar and poison bar to elevate and drop, the camera to always follow the player and be able to rotate according to the movement of the mouse, the resource area to randomly generate and etc. For future development of Apocalypse [15]. We have already released it on itch.io and is planning to make a mobile version for apple. After the game is launched on every planned page. We will publish a research paper concluding our goals and everything that has been done. When all is ready, we will enter some global computer science event like GameGala and Technovation to advertise Apocalypse and transmit our idea of environment protection. We will also broaden our horizons in these events and try to improve the simulator.

Currently, our team is meeting capital issues due to the lack of advertising and persuasive proposal to sponsors. But our game can arouse the responsibilities of people to protect the environment, very worth investing for a better tomorrow.

REFERENCES

[1] Scheler, Max, and Graham McAleer. The nature of sympathy. Routledge, 2017.
[2] Little, Margaret. "On basic unity." International Journal of Psycho-Analysis 41 (1960): 377-384.
[3] Duarte, Nancy. Resonate: Present visual stories that transform audiences. John Wiley & Sons, 2013.
[4] Southwood, Thomas Richard Edmund, and Peter A. Henderson. Ecological methods. John Wiley & Sons, 2009.
[5] Rao, P. K. Sustainable development. Vol. 1. Blackwell Publishers, 1999.
[6] Elliott, Jennifer. An introduction to sustainable development. Routledge, 2012.
[7] Gaston, Kevin J., and John I. Spicer. Biodiversity: an introduction. John Wiley & Sons, 2013.
[8] Kohler, Robert E. Landscapes and labscapes. University of Chicago Press, 2010.
[9] Vartanian, Thomas P. Secondary data analysis. Oxford University Press, 2010.
[10] Veal, Anthony J. "The concept of lifestyle: a review." Leisure Studies 12.4 (1993): 233-252.
[11] Merika, Heidi. Wildcraft: The science & spirit of wild plants as food & medicine. Heidi Merika, 2019.
[12] Base, Graeme. Animalia. Viking Kestrel, 1986.
[13] Long, Roma U. Roblox and Effect on Education. Diss. Master's Thesis, Drury University, Springfield, MO, USA, 2019.
[14] Schoenberg, Arnold. Theory of harmony. Univ of California Press, 1978.
[15] Coppola, Francis Ford. Apocalypse now. University of Chicago Press, 2017.

TRUSTING MACHINE LEARNING ALGORITHMS IN PREDICTING MALICIOUS NODES ATTACKS

Basim Mahbooba, Mohan Timilsina and Martin Serrano

Insight Centre for Data Analytics, NUI Galway, Galway City, Ireland

ABSTRACT

Identifying network attacks is a very crucial task for Internet of things (IoT) security. The increasing amount of IoT devices is creating a massive amount of data and opening new security vulnerabilities that malicious users can exploit to gain access. Recently, the research community in IoT Security has been using a data- driven approach to detect anomaly, intrusion, and cyber-attacks. However, getting accurate IoT attack data is time-consuming and expensive. On the other hand, evaluating complex security systems requires costly and sophisticated modeling practices with expert security professionals. Thus, we have used simulated datasets to create different possible scenarios for IoT data labeled with malicious and non-malicious nodes. For each scenario, we tested off a shelf machine learning algorithm for malicious node detection. Experiments on the scenarios demonstrate the benefits of the simulated datasets to assess the performance of the ML algorithms.

KEYWORDS

IoT Simulation, Data Labels, Malicious Nodes, Attacks, Trust, Prediction.

1. INTRODUCTION

Network security has become one of the important agendas of most organizations due to the increasing risks brought by network attack threats. As a result, companies are spending a tremendous amount of money on information technology and cybersecurity solutions. For example, in 2019, one of the leading research and advisory companies known as Gartner[1] spent over 124 billion dollars on information security. Thus, network security is one of the serious and vulnerable issues worldwide.

A **network attack** is any derogatory action that targets information systems, infrastructures, Internet of Things (IoT) devices, computers using various methods to steal, alter or destroy data or information systems. There are different variants of these attacks known by other names, such as Denial of Service (DoS), Man in the Middle (MitM), Phishing, Password attack, SQL injection attack, Birthday, Malware attack, and many more. Understanding these variants of attacks requires extensive background knowledge of the mechanism of attacks. Unfortunately, it is tough to do this because each of the attacks is different.

[1] 1https://www.gartner.com/en/newsroom/pressreleases/2018-08-15-gartner-forecasts-worldwideinformation-security-spending-to-exceed-124-billionin-2019

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 93-103, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120408

In Figure 1, demonstrates the example of IoT security attack. Alex's smart coffee machine is connected to the Internet via a special app. The hackers can target that app and steal Alex's bank card details. The smart coffee machine allows Alex to control it remotely by his phone. As coffee machines are not designed for security, the hackers can easily access Alex's bank information navigating via coffee machine apps. It is just one example. However, there are swarms of physical objects on the Internet at an unprecedented scale due to the Internet of Things (IoT) [7]. Recently, the boom of IoT devices has opened the fear of mass vulnerabilities [3]. There are billions of IoT devices in the world, which all collect loads of data in real-time. These physical objects include, but are not limited to, temperature sensors, smartphones, air conditioning, medical equipment, light bulbs, smart grid,

thermostats, and TVs [8]. These devices are connected to consumers, enterprises, and healthcare organizations. Their internal vulnerabilities have created a security blind spot where cybercriminals can launch various attacks to compromise devices like webcams, smart TV, routers, printers, and even a smart home.

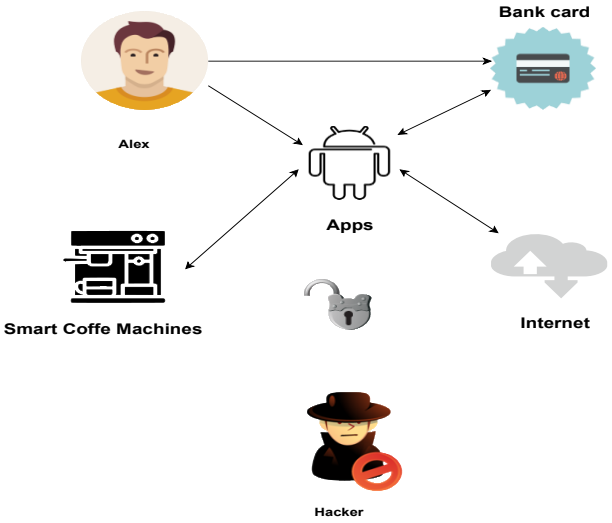


Figure 1. Example of IoT security Attack

There are many attacks due to the IoT devices when connected to the Internet. It is due to the limitations of these devices. Those limitations comprise small memory, minimal processing power, and shortbattery life. Nowadays, the attackers are focusing on IoT devices malicious intends [4]. The possible attack can be from hackers, hacktivists, and cybercriminals. The security in IoT is defined as the proper protection against acts such as data theft, unauthorized access, physical tampering, data manipulation, and network attacks [8]. Thus, cybersecurity is crucial for the proper operation of the IoT industry [22].

IoT raises many challenges, one of which is the massive amount of data generated by sensor devices. However, it was observed that, with the increase in sensor density, data generated by IoT devices tend to be highly redundant [23]. Thus, the redundant data consumes extra network re-sources and reduces network efficiency, making the IoT attacks easier for hackers. Also, another problemis inaccurate sensor data, which can mislead the net-work resources with erroneous information and lead to possible false reactions at the network center andmakes it vulnerable to attack. Thirdly, in the beginning, if the attack happens in a real-world scenario, there are few malicious nodes and large non-malicious nodes. Thus, early identifying such malicious nodes will reduce the massive loss of network resources.

In this paper, we address the problems as mentioned earlier using simple simulated data sets. As new technology is being accustomed to data- driven infrastructure, this leads the research more on to Machine Learning (ML) based applications along- side IoT. Thus, we are using an off-the-shelf machine learning suite to predict malicious nodes in a simulated setting. Such settings will allow us to test various ML algorithms to find which one works in what cases. The ML system provides high-performance indicators such as showing high accuracy score, which could enhance people's trust and acceptance of the system [30]. Thus, in this work, we explore different ML models to assess the prediction performance in a simulated environment. We believe that people trust an ML model if it can give high accuracy in unseen data.

Contributions: Our contributions are summarized as follows:

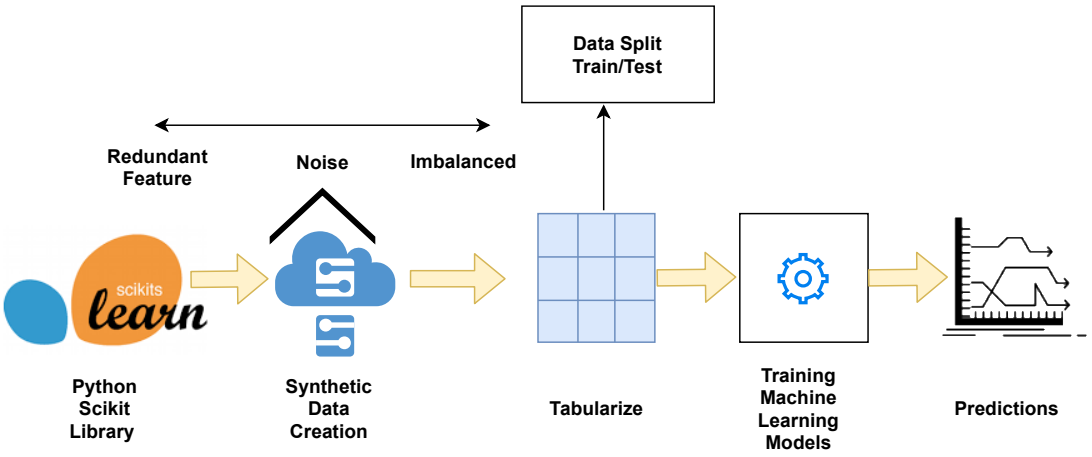
- Simulated data: We showed how we can take ad- vantage of generating simple simulated data creating different scenarios for malicious node detection.
- Evaluation: We applied off-the-shelf ML algorithms in the simulated datasets and evaluated the accuracy of those algorithms in those scenarios.

The rest of the paper is organized as standard: related work, method description, and experimental analysis, discussion and conclusion.

2. RELATED WORK

Trust is based on the evaluation of quantifying decision-making process [10]. For a sensor network, trust assessment based on malicious node detection can support to ensure the security of the network [32]. Similarly, in a social network, trust evaluation helps users build social relationships, reduce the risk of social activities and improve the quality of social networking [19][28]. Trust evaluation is also used to ensure secure interactions between automated agents and to promote the success of automation in a multi-agent system [26][15]. In Peer-to-Peer (P2P) networking, trust evaluation helps to identify interactive objects, ensuring resource sharing with friendly peers, and identifying malicious peers [14]. The majority of these trust algorithms determine trust by aggregating trust factors through weighting and other relevant calculations. Thus, to calculate such weight is not trivial and is hard to ensure evaluation accuracy. Therefore, many researchers in this field suggested using the ML approach for trust [29].

The problem of the identification of malicious nodes is regarded as a complex problem. It requires an in-depth analysis of nodes behaviors, and this motivates to use of ML-based models [1]. The study by Moore and Zuev [17] used the Na¨ıve Bayes estimator to classify internet traffic into security monitoring applications. They defined trust as how much the classification algorithms can be trusted based on their ac- curacy in unseen data. In this case, trust towards a dependable ML tool can be initiated as the accuracy with which the decision-maker follows the model's prediction [20][25]. One of the aspects of trust- worthy ML is based on the idea of maintaining high levels of performance and accuracy [27]. Every ML application requires a certain measure of overall trust in the model. It is assumed that the greater the accuracy values or the performance of the model, the higher the credibility of the ML models. The user forms a trust in ML algorithms based on the repeated interaction and accuracy of predictions produced by the machine learning models [13]. The study by [30] showed that how model accuracy can affect user trust. Similarly, findings from [31] demonstrated that the accuracy of the model did have a significant effect on the extent to which people trust the model.



To train the ML models for malicious node identification, a dataset of samples of intrusion signs that would reflect the abnormal behavior in the network under different scenarios is required. Thus, to create such real datasets is expensive, complicated, and time-consuming. The study by Belanko [5] showed the usefulness of synthetic data generation for intrusion detection in VANET (vehicular ad-hoc network). Many cybersecurity problems for malicious node detection are studied in simulated datasets [18][16]. IoT data is considered as noisy, heterogeneous, incomplete, high-dimensional, and nonlinear [9]. Thus, simulating such data and detecting threats using off-the-shelf ML algorithms is the main objective of this work. To the best of our knowledge, there is no such approach to demonstrate which ML algorithms work better in which scenario and which model is helpful to the user for identifying malicious nodes.

3. METHODS

This section presents the synthetic data preparation and ML models and evaluation criteria that we used in the study.

3.1. Datasets

Three different synthetic datasets are created using the Python Scikit-learn library[2] . Scikitlearn (sklearn) is an open-source machine learning library built on top of the Python programming language. This library contains a lot of efficient tools for machine learning and statistical modelling. The properties of the created datasets are shown in Table 1.

3.2. Evaluation Criteria

We evaluated the ML algorithms using 5-fold cross-validation. It is because cross-validation helps us to estimate the skill of a machine learning model on unseen data in different folds. For every fold, we receive the performance metrics of the ML algorithms, which enable us to draw important conclusions about the model.

3.3. Evaluation Metric

We have used the Area Under Precision-Recall Curve (AUPR) as the evaluation metric. One of the reasons for this is that we would like to know the skill of ML algorithms to predict positive

[2]https://scikit-learn.org/stable/

Table 1: Synthetic Datasets. ✓: Presence, ×: Absence, N: Number of data points

Question Addressed	N	Features	Label Percentage	Redundant Features	Noise
Q1	1000	10	50% malicious, 50% Non-Malicious	✓	×
Q2	1000	10	50% malicious, 50% Non-Malicious	×	✓
Q3	1000	10	1% malicious, 99% Non-Malicious 5% malicious, 95% Non-Malicious 10% malicious,90% Non-Malicious 20% malicious, 80 Non-Malicious	××	

class, which in our context is the identification of malicious nodes. On the other hand, AUPR provides an accurate prediction of future classification performance since they evaluate the fraction of true positives among positive predictions [24] and is considered robust performance metrics for imbalanced datasets. • Machine Learning Classification Models. We applied a set of six off-the-shelf classification models: Logistic Regression, Decision Tree Classifier, Support Vector Classification (SVC), Gradient Boosting Classifier, Multilayer perceptron (MLP) Classifier, Random Forests Classifier. In the next, we provide a short explanation of each of these models:

⋆ **Logistic Regression** is an appealing classification model because it is fast and straightforward to execute. The model is very applicable for linear data; however, it is not highly accurate for predicting complex non-linear data.
⋆ **Decision Tree Classifier** organizes rules in a tree to classify data. The rules produced by the decision tree are very intuitive and easy to understand. However, if a tree is designed to perfectly fit all training data set, it can easily be over-fitted and leads to a poor generalization in the test.
⋆ **SVC's** main advantage is that it can account for complex, non-linear relationships between features and survival. SVC finds a line that acts as a boundary to separate the data. This makes SVC extremely adaptable and applicable to a wide range of data.
⋆ **Gradient Boosting Classifier** is similar to a Random Forests Classifier because it depends on numerous base learners to produce an overall prediction, but varies in how those are aggregated. While a Random Forests Classifier fits a set of classification Trees separately and then averages their predictions, a Gradient Boosting Classifier is constructed back-to-back in a greedy stagewise manner.
⋆ **MLP classifiers** simulates a biological interconnected neuron. MLP is considered to have high accuracy and strong parallel distributed processing ability. The model works with a large number of parameters and provides quality prediction performance; however, the output results are difficult to explain and are termed, black-box models.
⋆ **Random Forests Classifier** is an ensemble of tree-based learners. It ensures that individual trees are de-correlated by 1) constructing each classification tree on a different bootstrap sample of the initial training data and 2) at each node, only estimate the split criterion for a randomly selected subset of features and thresholds. Predictions are formed by aggregating predictions of individual trees in the ensemble. • Model training pipeline. We trained the above-described ML models on the synthetic data using the procedure demonstrated in Figure 2. The ML models are trained using five-fold cross-validation. The model parameters are identified using grid search in the training sets.

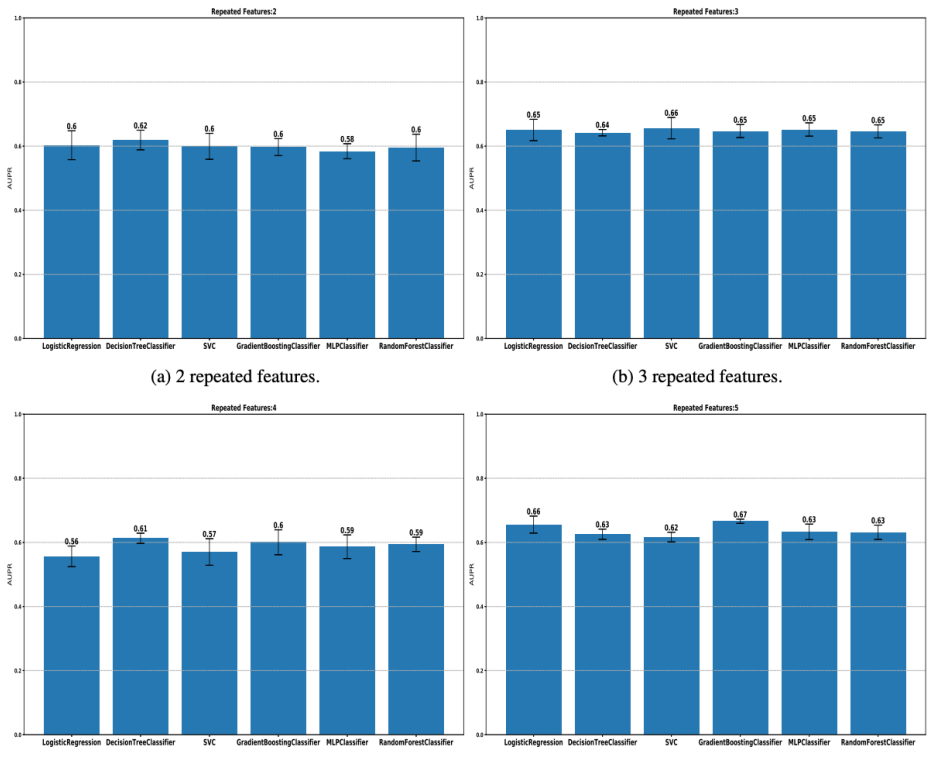
4. EXPERIMENTS

We investigated three questions. The questions are as follows:

- Q1-Redundant Features: Can we identify the malicious nodes in a setting where we have repeated features?
- Q2-Noise: How accurate are the ML models to predict malicious nodes in the noisy setting?
- Q3-Imbalanced: Are ML models still be able to learn and predict the malicious nodes in an imbalanced label distribution?

4.1. Q1- Redundant Features

IoT data can exhibit redundancy [21]. This behavior might impact degradation of the overall performance of the IoT sensor networks [12]. We varied the redundant features from 2,3,4, and 5 in our synthetic datasets. The performance of different algorithms in different redundant feature settings is shown in Figure 3. We ran all the ML algorithms in this setting. We found that in the highest number of repeated features, which is five, the Gradient Boosting classifier performed best compared to other algorithms. Similarly, the Gradient Boosting classifier has competitive performance with its competitors Decision Tree, Random Forest, and MLP Classifier for four repeated features. For two and three repeated features, we can see from the bar-plot (Figure 3) that all the algorithms have similar performance. It implies that in the setting where there are many repeated features, score. ensemble-based models like Gradient Boosting Classifier or Random Forest Classifier would be applicable.



(a) 2 repeated features. (b) 3 repeated features.

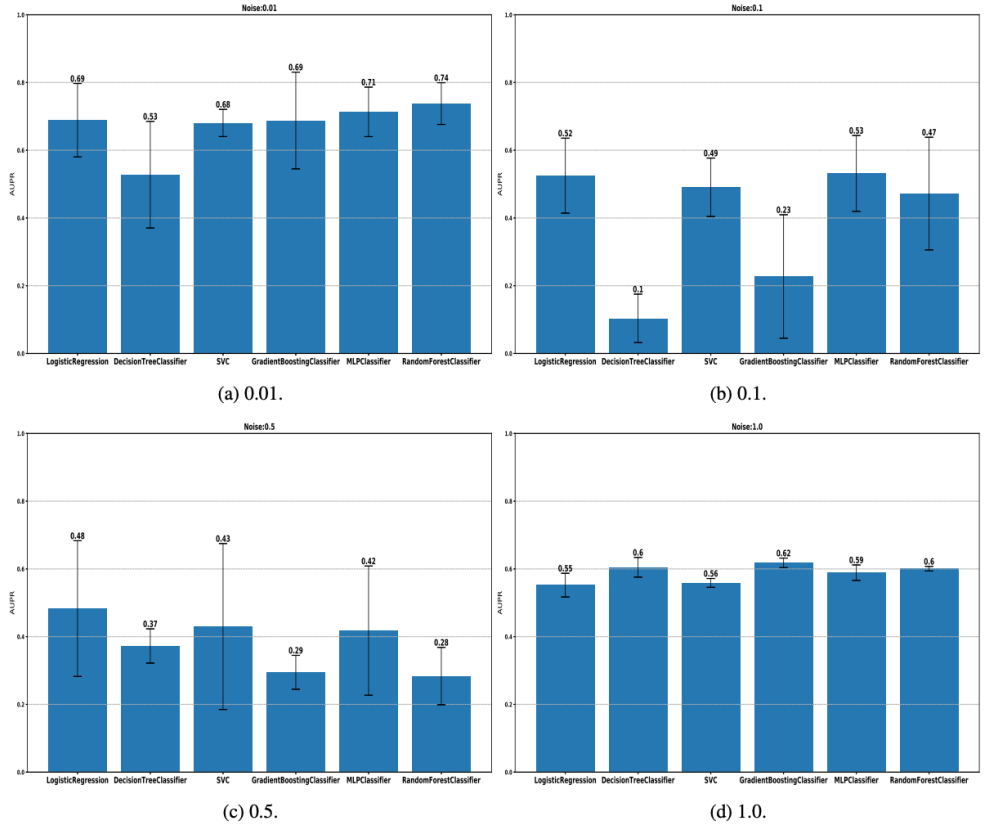
(c) 4 repeated features. (d) 5 repeated features.

Figure 3: The bar chart shows the Mean AUC-PR score of 5-Fold cross validation to predict malicious nodes. The error bar is the standard deviation obtained from the 5-Fold cross validation of AUC-PR score.

4.2. Q2- Noise

In IoT devices, it might suffer from irregularities in the correct readings of the sensor data, which might make them vulnerable for attack. Often data collected by IoT devices are noisy and corrupted [20]. To simulate that situation, we created synthetic datasets with different noise proportions from 0.01 to 1.0. 0.01 is a tiny noise distribution in the data, whereas 1.0 is high

noise in the data. We trained the ML models in this setting, and the result is presented in Figure 4. We observe that as the noise increases, the AUPR score of the model decreases. From Figure 4(b), we can see there is a huge drop in AUPR score when the noise is increased from 0.01 to 0.1. One of the reasons for that might be adding noise may under-fit the data, and a valuable signal is lost, so the ML model has performance degradation. However, in Figure 4(d), there is quite improved performance of the ML models with noise 1.0. For noise at 1.0, there is a positive influence in ML models that worked as regularizes to reduce over-fitting and has improved performance.



(a) 0.01. (b) 0.1.

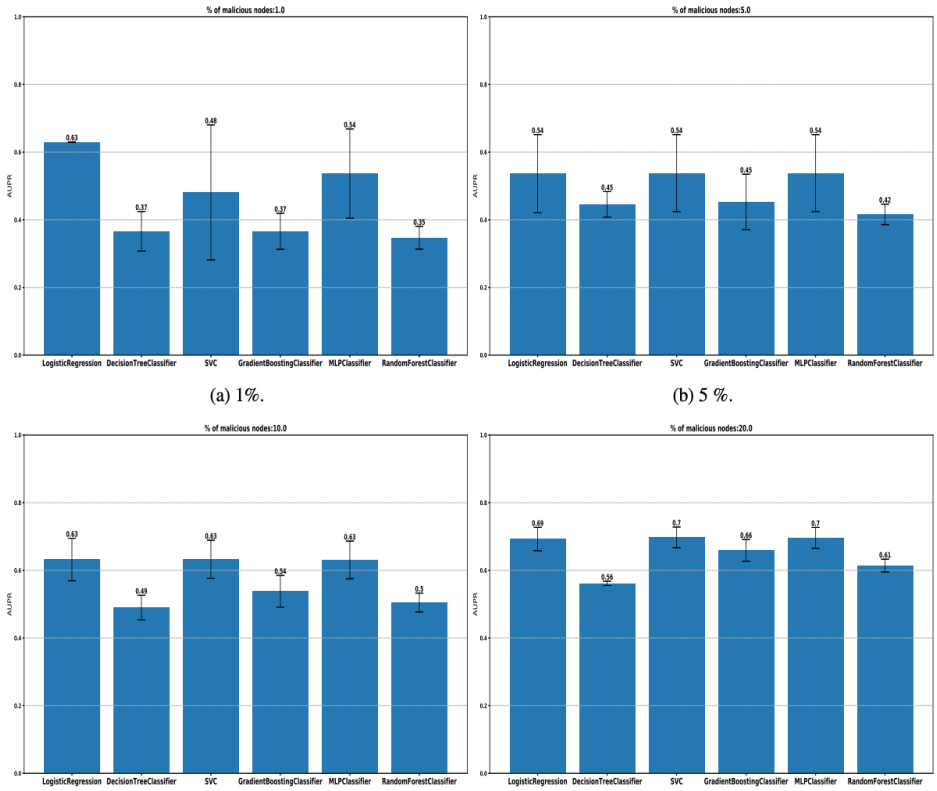
(c) 0.5. (d) 1.0.

Figure 4: The bar chart shows the Mean AUC-PR score of 5 Fold cross validation to predict malicious nodes. The error bar is the standard deviation obtained from the 5 Fold cross validation of AUC-PR score.

Q3- Imbalanced Data

When the attack happens, there is a chance that there are few malicious nodes and large non-malicious nodes at first. In such an event identifying the malicious node in advance will save the huge loss. Thus, we created a synthetic dataset with 1%, 5%, 10%, and 20% malicious nodes for this setting and applied the ML models. The result is shown in Figure 5.

For 1% malicious node, we observed that Logistic Regression had performed better than other models. It is because logistic regression performs better when the dataset is linearly separable and less prone to overfitting. However, when more malicious nodes are introduced, other models start to perform better. For example, in the case of 20% malicious nodes, SVC and MLP classifiers outperform all the other methods. Similarly, in 5% and 10%, Logistic regression, MLP, and SVC have similar performances.



(a) 1%. (b) 5 %.

(c) 10%. (d) 20%.

Figure 5: The bar chart shows the Mean AUC-PR score of 5-Fold cross validation to predict malicious nodes. The error bar is the standard deviation obtained from the 5-Fold cross validation of AUC-PR score.

5. DISCUSSION

From the experiment, we came up with the following summary shown in Table 2. We have demonstrated the summary for extreme cases, namely the maximum redundant features, noise, and imbalanced dataset with only 1% labeled malicious and 99% Non-malicious due to space limitation. We can see at the Gradient Boosting Classifier performs the best for a high repeated number of features and maximum noise. Gradient Boosting trains many models in a gradual, additive, and sequential manner to enhance the predictive performance and has better performance than others. Similarly, we observe that the Logistic Regression performed the best in predicting few malicious nodes and large non-malicious ones. If there is a linear association between features and the outcome variable, then logistic regression is able to capture that information and gives better performance in comparison to other methods. We tested off-the-shelf ML classification algorithms for our synthetic data to identify the best performing algorithm. However, picking up a suitable algorithm is always not enough. We must also choose the correct configuration of the algorithm for a dataset by tuning the parameters to get high predictions. These high predictions are evaluated by metric.

An ML algorithm's performance metric can represent effectiveness. For many years, machine learning researchers measured the trustworthiness of the models through evaluation metrics. The evaluation metric like AUPR which we used in our studies describe the number of correct predictions made by the model. It answers an important question: if a model is making a random guess or has learned to classify the data correctly. The higher the AUPR score, the better is the prediction and the more trust we can do in the machine learning algorithms. However,

transparency, explainability, computational complexity are also the primary step for enhancing trust in an ML system for malicious node detection and choosing suitable ML algorithms. We have not explored this scope, which is the limitation of this work, but we would like to explore this further in our future work.

The subjectivity is a basic characteristic of trust. In this work, we have not looked into the subjective aspect of the user for the trust in algorithms for predicting malicious nodes. The user studies in the trust is an important issue. As this is the initial stage of our work, we consider user studies part of our further research.

Table 2. Qualitative comparison of the performance of the ML classification models in a synthetic data.✓: higher performance, ×: poor performance

	Machine Learning Classification Models					
	Logistic Regression	Decision Tree Classifier	SVC	Gradient Boosting Classifier	MLP Classifier	Random Forest Classifier
Scenarios						
Maximum Redundant Features (5)	×	×	×	✓	×	×
Maximum Noise (1.0)	×	×	×	✓	×	×
Imbalance Dataset (1% Malicious, 99% Non Malicious)	✓	×	×	×	×	×

6. CONCLUSIONS

We presented the malicious node classification using off-the-shelf ML algorithms in a synthetic dataset. We showed the ability of ML algorithms in the scenarios mimicking for IoT devices: (a) redundant features, (b) noisy, and (c)imbalanced datasets and reported the AUPR score of the algorithms. Our work explores looking at the synthetic data and applying ML models to see which one performed the best in what setting.

We believe the evaluation metric provides the first step towards trusting the ML for developing malicious node detection. However, this idea will give a simple, cheap, and effective way to generate the data and perform ML experiments for secure systems.

ACKNOWLEDGEMENTS

We would like to acknowledge Science Foundation Ireland (SFI/12/RC/2289_P2) for funding this research.

REFERENCES

[1] Abdelghani, W., Zayani, C. A., Amous, I., and Se`des, F. (2018). Trust evaluation model for attack detection in social internet of things. In International Conference on Risks and Security of Internet and Systems, pages 48–64. Springer.

[2] Abdul-Ghani, H. A., Konstantas, D., and Mahyoub, M. (2018). A comprehensive iot attacks survey based on a building-blocked reference model. International Jour- nal of Advanced Computer Science and Applications, 9(3):355–373.

[3] Anand, P., Singh, Y., Selwal, A., Alazab, M., Tanwar, S., and Kumar, N. (2020). Iot vulnerability assess- ment for sustainable computing: threats, current solu- tions, and open challenges. IEEE Access, 8:168825– 168853.

[4] Arıs, A., Oktug, S. F., and Voigt, T. (2018). Security of internet of things for a reliable internet of services.

[5] Belenko, V., Krundyshev, V., and Kalinin, M. (2018). Syn- thetic datasets generation for intrusion detection in vanet. In Proceedings of the 11th international confer- ence on security of information and networks, pages 1–6.

[6] Cao, N., Nasir, S. B., Sen, S., and Raychowdhury, A. (2017). Self-optimizing iot wireless video sensor node with in-situ data analytics and context-driven energy- aware real-time adaptation. IEEE Transactions on Circuits and Systems I: Regular Papers, 64(9):2470– 2480.

[7] DaCosta, F. and Henderson, B. (2013). Rethinking the In- ternet of Things: a scalable approach to connecting everything. Springer Nature.

[8] Deogirikar, J. and Vidhate, A. (2017). Security attacks in iot: A survey. In 2017 International Conference on I- SMAC (IoT in Social, Mobile, Analytics and Cloud)(I- SMAC), pages 32–37. IEEE.

[9] El-Zeheiry, H., Elmogy, M., Elaraby, N., and Barakat, S.(2018). Fuzzy c-mean and density-based spatial clus- tering for internet of things data processing. In Med- ical Big Data and Internet of Medical Things, pages 161–187. CRC Press.

[10] Jiang, W., Wu, J., Li, F., Wang, G., and Zheng, H. (2016). Trust evaluation in online social networks using gen- eralized network flow. IEEE Transactions on Comput- ers, 65(3):952–963.

[11] Kang, M.-J. and Kang, J.-W. (2016). Intrusion detection system using deep neural network for in-vehicle net- work security. PloS one, 11(6):e0155781.

[12] Krishnamurthi, R., Kumar, A., Gopinathan, D., Nayyar, A., and Qureshi, B. (2020). An overview of iot sensor data processing, fusion, and analysis techniques. Sensors, 20(21):6076.

[13] Lee, E. (2021). How do we build trust in machine learning models? Available at SSRN 3822437.

[14] Li, X., Zhou, F., and Yang, X. (2011). A multi-dimensional trust evaluation model for large-scale p2p comput- ing. Journal of Parallel and Distributed Computing, 71(6):837–847.

[15] Mahbooba, B., Timilsina, M., Sahal, R., and Serrano, M. (2021). Explainable artificial intelligence (xai) to en- hance trust management in intrusion detection sys- tems using decision tree model. Complexity, 2021.

[16] Milenkovic, M. (2020). Internet of Things: Concepts and System Design. Springer.

[17] Moore, A. W. and Zuev, D. (2005). Internet traffic classi- fication using bayesian analysis techniques. In Pro- ceedings of the 2005 ACM SIGMETRICS interna- tional conference on Measurement and modeling of computer systems, pages 50–60.

[18] Navarro-Lara, J., Deruyver, A., and Parrend, P. (2016). Morwilog: an aco-based system for outlining multi- step attacks. In 2016 IEEE Symposium Series on Com- putational Intelligence (SSCI), pages 1–8.

[19] Niu, D., Rui, L., Huang, H., and Qiu, X. (2017). A service recovery method based on trust evaluation in mobile social network. Multimedia Tools and Applications, 76(3):3255–3277.

[20] Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Wortman Vaughan, J. W., and Wallach, H. (2021). Manipulating and measuring model interpretability. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, pages 1–52.

[21] Qin, Y., Sheng, Q. Z., Falkner, N. J., Dustdar, S., Wang, H., and Vasilakos, A. V. (2016). When things matter: A survey on data-centric internet of things. Journal of Network and Computer Applications, 64:137–153.

[22] Rajendran, G., Nivash, R. R., Parthy, P. P.,and Balamurugan, S. (2019). Modern security threats in the internet of things (iot): Attacks and countermeasures. In 2019 International Carnahan Conference on Security Technology (ICCST), pages 1–6. IEEE.

[23] Razafimandimby, C., Loscri, V., Vegni, A. M., and Neri, A. (2017). A bayesian and smart gateway based communication for noisy iot scenario. In 2017 International Conference on Computing, Networking and Communications (ICNC), pages 481–485. IEEE.

[24] Saito, T. and Rehmsmeier, M. (2015). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3):e0118432.

[25] Schmidt, P. and Biessmann, F. (2019). Quantifying interpretability and trust in machine learning systems. arXiv preprint arXiv:1901.08558.

[26] Schmidt, S., Steele, R., Dillon, T. S., and Chang, E. (2007). Fuzzy trust evaluation and credibility development in multi-agent systems. Applied Soft Computing, 7(2):492–505.

[27] Thiebes, S., Lins, S., and Sunyaev, A. (2020). Trustworthy artificial intelligence. Electronic Markets, pages 1–18.

[28] Timilsina, M., Davis, B., Taylor, M., and Hayes, C. (2017). Predicting citations from mainstream news, weblogs and discussion forums. In Proceedings of the International Conference on Web Intelligence, pages 237– 244.
[29] Wang, J., Jing, X., Yan, Z., Fu, Y., Pedrycz, W., and Yang, L. T. (2020). A survey on trust evaluation based on machine learning. ACM Computing Surveys (CSUR), 53(5):1–36.
[30] Yin, M., Vaughan, J. W., and Wallach, H. (2018). Does stated accuracy affect trust in machine learning algorithms. In Proceedings of ICML2018 Workshop on Human Interpretability in. Machine Learning (WHI 2018), volume 7.
[31] Yin, M., Wortman Vaughan, J., and Wallach, H. (2019). Understanding the effect of accuracy on trust in machine learning models. In Proceedings of the 2019 chi conference on human factors in computing systems, pages 1–12.
[32] Zhang, T., Yan, L., and Yang, Y. (2018). Trust evaluation method for clustered wireless sensor networks based on cloud model. Wireless Networks, 24(3):777–797.

AUTHORS

Basim Mahbooba
I am a Ph.D. student in the Unit of Internet of Thing and Network security of INSIGHT at the National University of Ireland, Galway. Before joining Insight, I was a researcher at Kufa University, Iraq.

Mohan Timilsina
Working as a postdoc in the Unit of Information Mining and Retrieval of INSIGHT at the National University of Ireland, Galway.

Martin Serrano
Dr. Martin Serrano (PhD. MSc. Eng. B.Eng) Data Scientist IoT-M2M Semantic Interoperability Expert National University of Ireland Galway (NUI Galway)

IOLLVM: ENHANCED VERSION OF OLLVM

Chengyang Li Tianbo Huang, Xiarun Chen,
Chenglin Xie and Weiping Wen

School of Software and Microelectronics, Peking University, Beijing, China

ABSTRACT

Code obfuscation increases the difficulty of understanding programs, improves software security, and, in particular, OLLVM offers the possibility of cross-platform code obfuscation. For OLLVM, we provide enhanced solutions for control flow obfuscation and identifier obfuscation. First, we propose the nested switch obfuscation scheme and the in-degree obfuscation for bogus blocks in the control flow obfuscation. Secondly, the identifier obfuscation scheme is presented in the LLVM layer to fill the gap of OLLVM at this level. Finally, we experimentally verify the enhancement effect of the control flow method and the identifier obfuscation effect and prove that the program's security can be further improved with less overhead, providing higher software security.

KEYWORDS

Software Protection, Code Obfuscation, Control Flow Obfuscation, Identifier Obfuscation, LLVM.

1. INTRODUCTION

Software protection [1-3] has become more attention-grabbing with the increased awareness of privacy and copyright, especially when the physical and virtual economies are becoming more and more integrated. According to the medium of intervention, existing software protection can be divided into hardware-level [4-6] and software-level [7-9] protection. The former can theoretically provide more robust protection because it can provide a relatively controlled environment. The latter has advantages in cost and applicability compared to the former. Especially with the development of IoT, the security level needed in different scenarios, and the availability of limited computing resources, software-level protection is unavoidable. Considering the broad applicability and objective security obtained for low cost, we focus on software-level protection. Software-level protection can be divided into protection before being damaged and forensics after being damaged, such as protecting core technologies through code obfuscation [10, 11], encryption and decryption [12, 13], software watermarking [7, 14], or software birthmark [15, 16] for effective tracking of software copyright.

Moreover, from the attacker's perspective, it can be divided into anti-static debugging and anti-dynamic debugging. Code obfuscation is effective against static analysis, and virtual machine protection [17, 18] is effective against dynamic analysis. However, virtual machine protection's huge overhead in time and space does not allow for a wide range of landing in the actual production environment. Furthermore, the low overhead of code obfuscation provides considerable security, making code obfuscation continue to occupy an essential role among the many software protection techniques available today. As for code obfuscation, it can be further divided from the hierarchy of roles into source code level [19], intermediate code [20], and binary file [21] obfuscation. Because the intermediate representation is easier to adapt to new front and

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 105-117, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120409

back ends and easier to carry out optimization work, we propose the code obfuscation scheme on LLVM intermediate representation (LLVM IR).

OLLVM (Obfuscator-LLVM) [20] has implemented a mature and effective obfuscation system on LLVM IR, but currently, OLLVM has some problems as follows. First, OLLVM is based on LLVM 4.0, and as of January 2022, LLVM has released LLVM13. Coupled with the large API gap between different versions of LLVM, OLLVM cannot use some new API features. The adaptation to the newly emerging front-end and back-end may also be problematic. Although different open-source communities [22, 23] have implemented OLLVM migration to new LLVM versions, it is inescapable that existing OLLVM obfuscation techniques are developed based on previous LLVM versions. Furthermore, OLLVM, as a mature obfuscation framework, has had different anti-obfuscation methods implemented for it, and the security that OLLVM itself can provide is questionable. Finally, the existing OLLVM provides obfuscation features including, bogus control flow and flattening at the control flow level and instruction substitution at the instruction level, but does not provide identifier [24] related obfuscation methods. Therefore, this paper implements the iOLLVM system to provide new obfuscation methods at the control flow and identifier levels to address the issues mentioned here.

Following are the main contributions of this paper:
1. The paper proposes enhanced flattening processing and adds in-degree obfuscation for OLLVM control flow obfuscation to resistance to existing scripting attacks;
2. The paper also proposes four algorithms in identifier obfuscation to compensate for the lack of OLLVM.

The rest of the paper is organized as follows: Section 2 introduces the overview of the iOLLVM. Section 3 describes the details of identifier algorithms, and the experiment results are presented in section 4. Finally, in section 5, we end up with a conclusion and a description of future work.

2. THE COMPONENTS OF IOLLVM

Considering the stability of the new version of LLVM API, the prototype iOLLVM obfuscation system was implemented on LLVM 10, giving effect enhancements at the control flow obfuscation level and the identifier obfuscation level, respectively. The general framework diagram is shown in Figure 1.

Based on the existing OLLVM, iOLLVM proposes an enhanced obfuscation implementation at the control flow level and complements the obfuscation method at the identifier level. The overall flow of the obfuscation system is as follows: first, the source code is transformed into LLVM IR code by the front-end tools provided by LLVM, such as clang, flang, denoted as xclang in Figure 1. Then, the LLVM IR code is obfuscated by the new modules provided by iOLLVM: the control flow module and the identifier module. They can be called and used separately or nested. Finally, the obfuscated IR files generate platform or environment-ready executables by specific back-end programs.

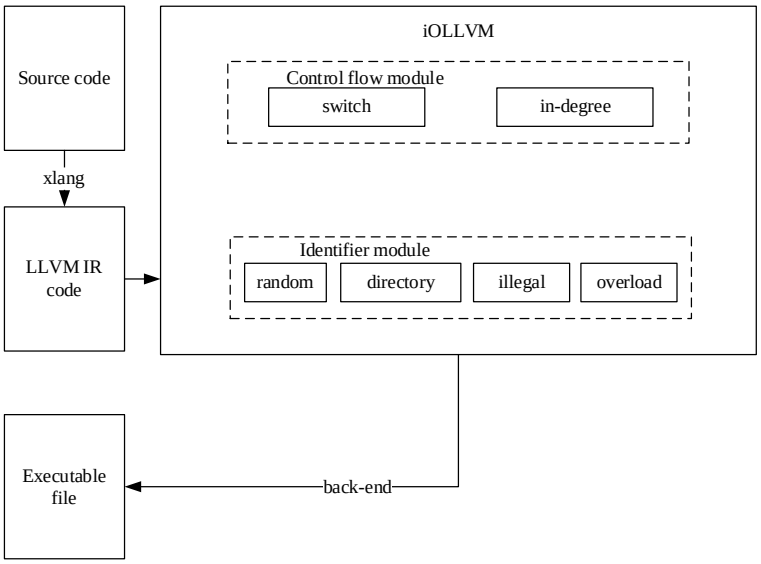


Figure 1. iOLLVM general framework diagram

2.1. Control Flow Obfuscation

According to Collberg's classification of code obfuscation [10], control flow obfuscation intends to effectively hide the program's control flow and make the attacker's path analysis more difficult. OLLVM provides control flow obfuscation, including bogus control flow and flattening processing [20]. However, using some IDA Python script files [25] to target the features of OLLVM obfuscation processing can effectively remove the added flattened obfuscated code. In the meantime, if a cracker can infer that control flow obfuscation has been performed based on the control flow graph processed by the automated tool, this itself is also valuable information for further analysis of the program. Moreover, at the same time, the bogus control flow design currently provided does not take into account the attack of the degree of entry analysis. Therefore, this paper proposes a counter-common sense nested switch flattening process and an in-degree obfuscation to prevent incidence analysis.

2.1.1. Nested Switch

The original purpose of flattening was to disrupt the program's execution flow. In LLVM IR, the basic blocks that initially had an upper and lower hierarchy are transformed into the same hierarchical relationship. Considered the convenience and the cross-platform nature of the program, the general implementation uses the dispatcher architecture of the switch, and the processed program has a prominent feature while bringing security, namely the presence of the loop structure. The switch structure distributes the basic blocks. From the point of view of hiding the flattened processing and resisting script analysis, the switch structure is created again in the flattened generated switch structure, and a new switch is created in each basic block corresponding to the case.

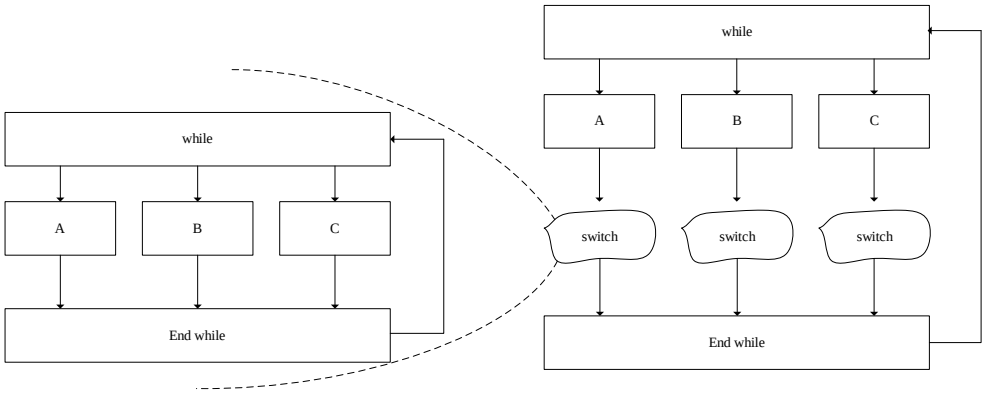


Figure 2. Nested switch structure

As shown in Figure 2, the part in the imaginary coil is noted as procedure F: the general flattening operation. The new switch structure is created inside the basic block, noted as procedure N. The procedure for adding boolean or numerical operations against variable v is noted as T. The steps of the nested switch are as follows.

Step 1: Perform procedure F on input program (or LLVM IR file) P0 to realize the flattening operation; the obfuscated program is noted as P1.
Step 2: Iterate through the basic blocks in P1 at the same level, and record them as the setBlocks, and for each basic block inside, process N to achieve the nesting of switch structure, and the obfuscated program is recorded as P2.
Step 3: For the switch generated at the P1 level, the jumping variable is recorded as the case-outer; for the switch generated at the P2 level, the jumping variable is recorded as the case-inner. The procedure T is added to the basic block corresponding to the case-inner, but only one block is left unprocessed to ensure normal program execution. The obfuscated program is denoted as P3.

P3 is the program after nested switch processing. The process of creating the nested switch structure requires attention to the creation of the basic block (or called the bogus block) and the setting of the jumping variables (case-outer, case-inner) used by the internal and external switches for distribution.

1. Because it involves the construction of a new switch structure, corresponding to the creation of new basic blocks, to enhance the similarity with the native program, drawing on the idea of bogus control flow, based on the program's native basic blocks to transform. Further, the number of bogus blocks is created because different space overhead scenarios are expected to be different. The default value is the number of cases of the outer switch while providing external parameters specified at the time of use.
2. Because the internal and external switch structure is involved, to strengthen the confusion of the internal switch structure and ensure the semantics of the program, the external case values are processed in the internal switch structure. However, there will always be a basic block in the internal randomly generated case that is not processed for the external case. All other basic blocks corresponding to the same level of cases add random boolean or arithmetic operations. This results in a combination of obfuscation processing on data dependencies.

2.1.2. In-Degree obfuscation

For the native bogus control flow, it is possible to protect the program by copying and transforming the native basic blocks to construct new bogus blocks embedded in the program through opaque predicates [26]. Generally, in practice, to reduce the space overhead, the obfuscation algorithm only protects against the critical code in the program. So if the attacker can recognize that the bogus control flow protects the program, there is a risk of identifying the bogus blocks. Plus, to hide the difference between the real block and the bogus block, a path to the real block can generally be constructed in the bogus block, so there will be: in general, the real block in-degree will be greater than that of the bogus block. Moreover, this can play a more significant role in excluding bogus blocks. Therefore, to resist the attacker's degree of entry analysis, the degree of entry of the bogus block is obfuscated.

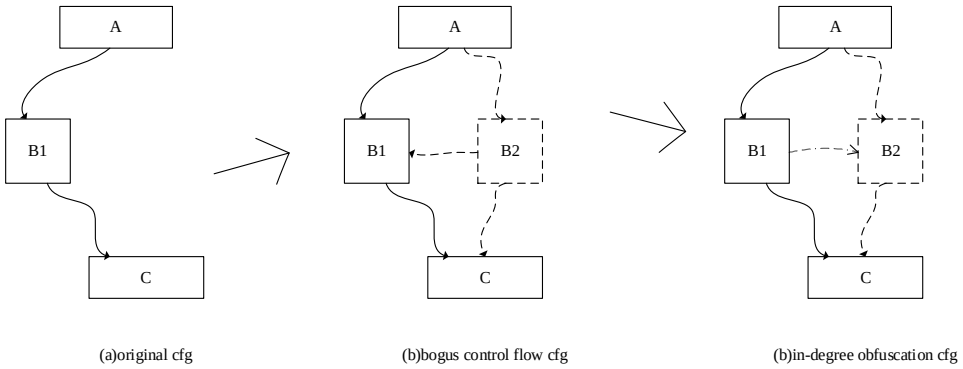


Figure 3. In-degree obfuscation

Figure 3(a) shows the sample program with a direct jumping relationship between the basic blocks A, B1, and C. Figure 3(b) shows the schematic diagram after adding the bogus control flow processing. Figure 3(c) shows the schematic diagram after adding the in-degree obfuscation processing, noting that B1 points to the edge of B2.

The fundamental point of entry analysis is to make bogus blocks have more entries than the real block by opaque predicates, i.e., adding opaque predicates to the real block to realize the addition of false jump relations. However, suppose there are already jump relations in the original real block without splitting the real block. In that case, the logic can be combined with the idea of flattening, transforming different branches into switch-case jump forms, and adding opaque predicates to the internal cases.

2.2. Identifier Obfuscation

During the reverse analysis of the program, the strings and identifiers exposed by the program will play a significant role in guiding the attacker to the specific location in the program based on the strings. They may even expose some core algorithm design and other information about the program. Therefore, it is necessary to hide the string and identifier information in the program effectively. Although OLLVM does not provide obfuscation protection for strings, both Armariris [27] and Hikari [28] give specific open-source schemes to protect certain programs. However, no explicit protection method is given for identifiers. Although the identifier information in the program can be partially removed using strip against the program, it is beneficial but not harmful to provide the obfuscation of identifiers in terms of the completeness of security and the uncertainty of the usage scenario. Therefore, this paper adds to this aspect by

providing four obfuscation algorithms for identifiers in LLVM IR: Random, Directory, Illegal, and Overload.

3. ALGORITHMS FOR IDENTIFIER OBFUSCATION

This paper proposed four algorithms to provide software security, namely, Random, Directory, Illegal, and Overload. The first three belong to the general category of substitution algorithms, while the fourth is a separate algorithm category.

Algorithm 1 Substitution algorithm

```
1: init parameters that algorithm needs
2: Read LLVM IR file
3: get all custom identifiers noted as setNames
4: select a algorithm from Set(Random, Directory, Illegal)
5: for item in setNames:
6:     change item in global
7: end
```

Among them, the replacement data sources for Directory and Illegal need to be initialized beforehand, and the replacement data sources for Random can be generated beforehand or dynamically at runtime.

Algorithm 2 Overload algorithm

```
1: Read LLVM IR file
2: get all custom identifiers noted as setNames
3: for item in setNames:
4:     generate a new function name item with random parameters, noted as newname
5:     if(newname is legal)
6:         added it to LLVM module
7: end
```

The critical point of the overload algorithm is that some languages support the overloading feature internally. When designing, we need to judge whether the overloading feature is satisfied based on the name mangling of the parameter list setting information. Next, the algorithm details are explained as follows.

Random algorithm: First, obtain the identifiers in the program. Randomly generate 11 identifiers containing letters, numbers, and special symbols and conform to the naming convention. Then, perform the global replacement for each identifier.

Directory algorithm: Unlike the idea of randomly generating identifiers for replacement, the purpose of the obfuscation dictionary is to use standard identifiers to replace identifiers in the program. Standard identifiers can play a preemptive role in obfuscation, unlike meaningless random characters.

Illegal algorithm: There will be reserved words in the program design, and it is not possible to use reserved words to name the identifiers, but since all characters are coded at the code set, characters of different codes set may be highly similar in appearance, so the Greek letters are used to replace the English letters thus playing the role of replacement confusion.

Overload algorithm: To further enhance the obfuscation of the program, the program is processed using the overload idea. Also, to avoid the name-mangling mechanism of the program itself, the

parameter list information is kept when obtaining the identifier information. Then the overloaded identifier of the virtual parameter list is constructed.

Because the idea of the first three is the idea of substitution, the fourth is the idea of addition. Therefore, in the user use, provide the default way, from the first three randomly choose one, and then with the fourth to reuse the previous confusing characters as much as possible.

4. EXPERIMENTS

We carry out three experiments: a) control flows obfuscation versus OLLVM; b) identifier obfuscation effects; c) time and space overhead.

Experiment environment: Ubuntu18.04; Inter(R) Core(TM) i5-4210U; 4G Memory; LLVM10.

Experimental data is selected from the algorithm library [29] implemented in c language on GitHub. In order to facilitate the automatic execution of the script, the part of the algorithm that involves input is converted from user input to fixed-value input. Finally, only 123 source files are kept.

4.1. Experiment 1- Control Flows Obfuscation Versus OLLVM

Function kthSmallest in kth_smallest.cpp in the test set is used to illustrate the effect of obfuscation. First, use clang to process the source program to get the corresponding LLVM intermediate code bc file; then use the so file generated by the obfuscation system to obfuscate the specified file. Here we can further specify the function to be obfuscated. The sample below specifies the function kthSmallest for processing. Because C++ has a name mangling mechanism compiled into intermediate code, so the function name seen in the intermediate code layer is _Z11kthSmallestP8TreeNodei. But the user only needs to provide the function's name in the source code when specifying it. Finally, Graphviz is used to generate the graph summary as follows.

4.1.1. Nested Switch

The control flow diagram of the function before obfuscation is shown in Figure 4, the control flow diagram after flattening with OLLVM is shown in Figure 5, and the control flow diagram after nest switch processing is shown in Figure 6. For ease of display, only show the logical relationships of the basic blocks while the instructions in Figure 5 and Figure 6 are removed.

Figure 4. KthSmalles: origin cfg

As shown in Figure 5, the original basic block relationship between the upper and lower levels is transformed into the same hierarchy level, with efficient scheduling through the switch. The attacker can infer flattened processing based on the current control flow graph and then use the existing attack script to approximate the program.

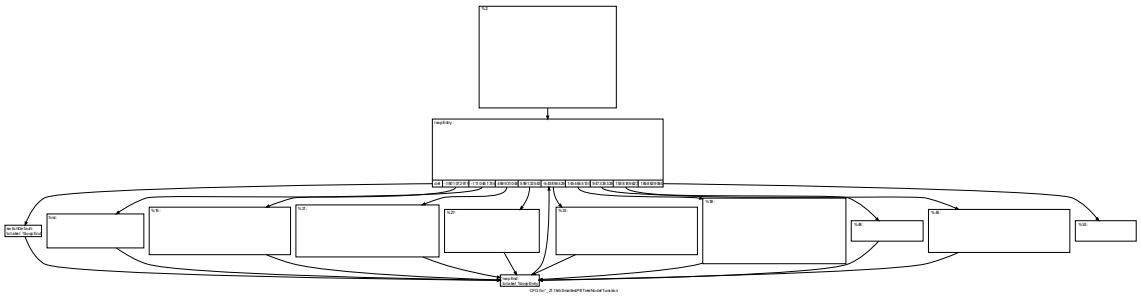


Figure 5. KthSmalles: ollvm flattening cfg

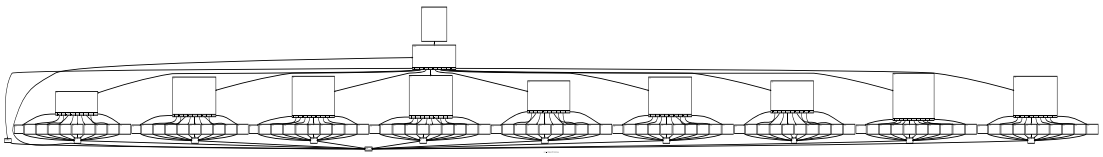


Figure 6. KthSmalles: nested switch flattening cfg

As shown in Figures 4, 5, and 6, nested switches can make an order of magnitude increase in complexity. The nested structure is different from the existing flattening processing. Even if an attacker uses an existing attack script for processing, it can effectively resist the existing attack script because the characteristics of the obfuscation algorithm are different. In addition, for control flow obfuscation, the similarity of basic blocks (BB), jump instructions(JI), and functions(F) after obfuscation and the similarity of programs(P) before and after obfuscation can be used as indicators to measure the effectiveness of obfuscation. Therefore, BinDiff [30] counts the above four indicators in the test set. Table 1 shows the statistics of the metrics after OLLVM obfuscation, and Table 2 shows the statistics of the metrics after obfuscation using nested switches and in degrees.

Table 1. Similarity of OLLVM obfuscated files to source files

OLLVM flatten	Average value	Minimum value	Maximum value	Standard deviation
similarity of BB	68.33	40.2	100	12.93
similarity of JI	36.55	10.9	100	21.4
similarity of F	76.72	55	92.6	8.79
similarity of P	0.74	0.56	0.99	0.1

Table 2. Similarity of nested switch, in-degree obfuscation files to source files

Nested switch, in-degree	Average value	Minimum value	Maximum value	Standard deviation
similarity of BB	29.09	6.7	100	21.01
similarity of JI	14.19	1.6	100	21
similarity of F	76.67	55	92.6	8.79
similarity of P	0.5	0.34	0.99	0.15

In terms of jump instructions, methods in this paper are on average 61% lower than OLLVM and 57% lower in terms of similarity of basic blocks. The reason is twofold: firstly, this method constructs more basic blocks and jump instructions, which significantly reduces the similarity with the native program and increases the confusion; secondly, the jump instructions in some of the original basic blocks are changed, so the decrease in the similarity of jump instructions is more prominent.

There is essentially no difference in the number of functions considering the effect of errors generated by BinDiff decompilation. The similarity between the obfuscated program and the original program, the method in this paper decreases by 32% compared to OLVM. Meanwhile, the maximum values of the four metrics, OLLVM, and this paper's method agree, i.e., for some particular files, the effect of the control flow obfuscation method proposed in this paper is consistent with OLLVM. However, the minimum values and standard deviations show that this paper's method performs better than OLLVM on the applicable files. It should be added that the similarity is one of the indicators of the obfuscation effect. However, suppose we only increase the number of basic blocks and jump instructions. In that case, such processing will not play a corresponding resistance to the substantial reverse cracking. However, this paper combines the

construction of basic blocks and jump instructions with the control flow and data flow of the original program itself, so the obfuscation effect is guaranteed to a certain extent.

4.1.2. Indirect Jump

The control flow diagram of the function before obfuscation is shown in Figure 4, and the function after obfuscation using the proposed obfuscation process for in-degree analysis is shown in the following figure, Figure 7. The image has been simplified for easy display.

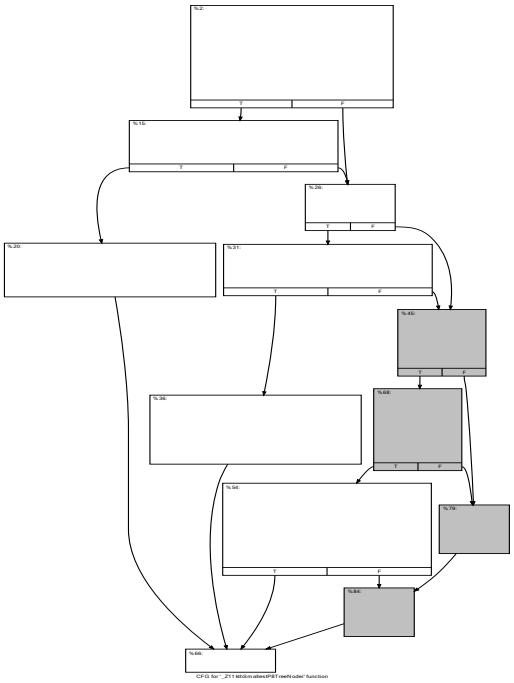


Figure 7. In-degree obfuscation effect

The grey blocks in Figure 7 are newly bogus, while white blocks are original blocks in the origin program. It can be seen that by introducing opaque predicates, the degree of entry of the bogus block is significantly higher than that of the actual block while ensuring the original semantics of the program, thus avoiding the threat of entry analysis. Limited to the constructional characteristics of the algorithm itself, it will have higher security when used simultaneously with other control flow methods.

4.2. Experiment 2- Identifier Obfuscation Effects

After using the identifier obfuscation algorithm for the 123 files in the test set, the function identifier replacement rate is 65.2% (4875/7515). In order to ensure the standard semantics of the program and avoid modifications to third-party library functions, only user-defined function names are selected here. It should be noted that for large projects in terms of identifier replacement, we can consider using wllvm [31] to link the project code into a single bc file and then obfuscate the identifiers for the bc file to avoid errors caused by symbolic links.

4.3. Experiment 3-File Size and Performance Penalty of Protected Program

Table 3. Time and space overhead of obfuscation methods

Obfuscation Method	Time overhead	Space overhead
OLLVM Flattening	1.01	1.05
Identifier obfuscation	0.98	1.00
Nested switch, in-degree	1.02	1.59

The space overhead is measured by comparing the increase of the file size after obfuscation with the increase in file size before obfuscation. As for time overhead, we use a script to count the time of multiple file executions before and after obfuscation. Furthermore, taking the average value as a benchmark for comparison. From Table 3, we can see that the control flow enhancement scheme proposed in this paper has a slight increase in time and space compared to OLLVM. In particular, considering that the program may run slightly differently in time under different execution states, the existing test sets are all small volume algorithm files, so they are more sensitive to execution time. Therefore, it can be considered that the identifier obfuscation has almost no overhead impact in terms of time and space.

It should also be noted that in large projects, as the volume of the protected code increases, the corresponding time and space overhead also increases, and the two should be linear. When using, different parameter values should be set in conjunction with specific usage scenarios to meet the security and performance requirements of the scenario.

5. CONCLUSIONS

This paper addresses the lack of strength of OLLVM obfuscation in control flow protection and the gap in identifier obfuscation by proposing two broad categories of enhancements. In control flow obfuscation, first, adding nested switches at the control flow level and adding the switch structure again in the flattened code, thus increasing the complexity of the code while resisting existing scripting attacks; second, proposing an in-degree treatment for bogus blocks to increase the confusion of bogus blocks further. Further, at the level of identifier obfuscation, four algorithms are proposed and bridge the gap of OLLVM in identifier obfuscation. By comparing with OLLVM, this paper can significantly improve the original control flow complexity in obfuscation effect; replace 65.2% of custom identifiers while guaranteeing program functionality. Furthermore, the time overhead from obfuscation is almost negligible. The space overhead is at 1.5 times.

In future work, we will pay attention to generating more secure opaque predicates and are not limited to the number-theoretic model. Meanwhile, the practical effectiveness of existing obfuscation algorithms in large projects remains tested. Therefore, we will focus on how to provide more accessible use of the obfuscation framework model in large projects.

ACKNOWLEDGEMENTS

Thanks to every author who participated in the paper, it is a joint effort to produce this article.

REFERENCES

[1] C. Iwendi et al., (2020) "KeySplitWatermark: Zero Watermarking Algorithm for Software Protection Against Cyber-Attacks," vol. 8, pp. 72650-72660.

[2] J. V. d. Broeck, B. Coppens, and B. D. J. A. Sutter, (2020) "Flexible Software Protection," vol. abs/2012.12603.
[3] B. Abrath, B. Coppens, S. Volckaert, J. Wijnant, B. D. J. P. o. t. t. W. o. S. S. Sutter, Protection,, and R. Engineering, (2016) "Tightly-coupled self-debugging software protection,".
[4] Q. Ge, Y. Yarom, and G. J. P. o. t. t. A.-P. W. o. S. Heiser, (2018) "No Security Without Time Protection: We Need a New Hardware-Software Contract,"
[5] G. Nishant and S. J. Shubhnandan, (2013) "Watermark, Hardware Parameters and License Key: An Integrated Approach of Software Protection,".
[6] L. Davi, P. Koeberl, and A.-R. J. s. A. E. I. D. A. C. Sadeghi, (2014) "Hardware-assisted fine-grained control-flow integrity: Towards efficient protection of embedded systems against software exploitation," pp. 1-6.
[7] C. S. Collberg and C. D. Thomborson, (2002) "Watermarking, Tamper-Proofing, and Obfuscation-Tools for Software Protection," {IEEE} Trans. Software Eng., vol. 28, no. 8, pp. 735-746.
[8] C. S. Collberg, C. D. Thomborson, and D. Low, (1998) "Manufacturing cheap, resilient, and stealthy opaque constructs," in POPL '98.
[9] C. Linn and S. K. Debray, (2003) "Obfuscation of executable code to improve resistance to static disassembly," in CCS '03.
[10] C. Collberg, C. Thomborson, and D. Low, (1997) "A taxonomy of obfuscating transformations," Department of Computer Science, The University of Auckland, New Zealand1173-3500.
[11] S. Banescu, C. S. Collberg, V. Ganesh, Z. Newsham, and A. J. P. o. t. n. A. C. o. C. S. A. Pretschner, (2016) "Code obfuscation against symbolic execution attacks,".
[12] R. N. Ismanto and M. Salman, (2017) "Improving security level through obfuscation technique for source code protection using AES algorithm," pp. 18-22: Association for Computing Machinery.
[13] S. Chow, P. A. Eisen, H. Johnson, and P. C. van Oorschot, (2002) "White-Box Cryptography and an AES Implementation," in Selected Areas in Cryptography, 9th Annual International Workshop, {SAC} 2002, St. John's, Newfoundland, Canada, August 15-16, 2002. Revised Papers, 2002, vol. 2595, pp. 250-270: Springer.
[14] E. N. d. L. F. Jorge et al., (2015) "SensorWatermark: a scheme of software watermark using code obfuscation and tamperproofing for WSN," pp. 916-922.
[15] S. Nazir, S. Shahzad, N. J. A. J. f. S. Mukhtar, and Engineering, (2019) "Software Birthmark Design and Estimation: A Systematic Literature Review," vol. 44, pp. 3905-3927.
[16] K. Guan, S. Nazir, X. Kong, and S. u. J. S. P. Rehman, (2021) "Software Birthmark Usability for Source Code Transformation Using Machine Learning Algorithms," vol. 2021, pp. 5547766:1-5547766:7.
[17] Z. Tang et al., (2018) "VMGuards:A Novel Virtual Machine Based Code Protection System with VM Security as the First Class Design Concern," vol. 8, p. 771.
[18] Y. Shi, K. Casey, M. A. Ertl, and D. J. A. T. A. C. O. Gregg, (2008) "Virtual machine showdown: Stack versus registers," vol. 4, pp. 2:1-2:36.
[19] S. J. A. Datta, (2019) "DeepObfusCode: Source Code Obfuscation Through Sequence-to-Sequence Networks," vol. abs/1909.01837.
[20] P. Junod, J. Rinaldini, J. Wehrli, and J. Michielin, (2015) "Obfuscator-LLVM - Software Protection for the Masses," (in English), 2015 Ieee/Acm 1st International Workshop on Software Protection (Spro), pp. 3-9.
[21] M. A. Kinsy, D. Kava, A. Ehret, and M. J. A. Mark, (2018) "Sphinx: A Secure Architecture Based on Binary Code Diversification and Execution Obfuscation," vol. abs/1802.04259.
[22] (2022, 1/7). isrc-cas/flounder. Available: https://github.com/isrc-cas/flounder
[23] (2018, 1/7). yazhiwang/ollvm-tll: Ollvm+Armariris+LLVM 6.0.0. Available: https://github.com/yazhiwang/ollvm-tll
[24] V. Arnaoudova, L. M. Eshkevari, M. D. Penta, R. Oliveto, G. Antoniol, and Y.-G. J. I. T. o. S. E. Guéhéneuc, (2014) "REPENT: Analyzing the Nature of Identifier Renamings," vol. 40, pp. 502-532.
[25] (2019, 1/7). GeT1t/deollvm64: deobfuscator llvm arm64 script. Available: https://github.com/GeT1t/deollvm64
[26] Z. Liang, W. Li, J. Guo, D. Qi, J. J. I. C. o. P. i. I. Zeng, and Computing, (2017) "A parameterized flattening control flow based obfuscation algorithm with opaque predicate for reduplicate obfuscation," pp. 372-378.
[27] (2021, 11/28). GoSSIP-SJTU/Armariris: Available: https://github.com/GoSSIP-SJTU/Armariris

[28] (2020, 11/28). HikariObfuscator/Hikari: LLVM Obfuscator. Available: https://github.com/HikariObfuscator/Hikari
[29] (2019, 11/28). mandliya/algorithms_and_data_structures. Available: https://github.com/mandliya/algorithms_and_data_structures
[30] (2021, 11/28). zynamics.com - BinDiff. Available: https://www.zynamics.com/bindiff.html
[31] (2022, 1/8). travitch/whole-program-llvm. Available: https://github.com/travitch/whole-program-llvm

AUTHOR

Chengyang Li, born in 1996, M. S. His research interests include code obfuscation.

HUANG Tianbo, born in 1997, M. S. His research interests include cyberspace security, malicious code detection and code obfuscation.

Xiarun Chen, born in 1997, M.S. His research interests include system and network security, blockchain security and malicious code detection.

C**henglinXie**, born in 1996, M. S. His research interests include Operating System Security, Intrusion Detection and Malware Detection.

WEN Weiping, born in 1976, Ph. D. professor. His research interests include system and network security, big data and cloud security, intelligent computing security.

HOW DID HUMANS, ANIMALS AND PLANTS ORIGINATE?

Mei Yin

Nanjing, China

ABSTRACT

With black hole explosion under incredibly high temperatures leading to cosmic information billions of years ago, all matters had been in gaseous phase. With temperature dropping, under atomic attractive forces, adjacent atoms which made up gases attracted each other and formed a variety of big, small or tiny gaseous lumps. With the temperature persistently dropping, the tiny gaseous lump became colder and contracted and got smaller and turned into one in liquid state and subsequently in solid state according to the principle of expanding when heated and contracting when cooled in general cases. Gradually it developed and formed a human- or animal-like fetus or a plant-like seed. If it had the same compositions as a human, cow or sunflower, the human, the cow or the sunflower formed. Similar cases happened to other humans, animals and plants. Humans neither evolved from apes nor shared a common ancestor with apes.

KEYWORDS

Origins of humans, animals and plants, cosmic formation, property and counts of brain neuron in language and thinking areas, brain structure, black hole.

1. INTRODUCTION

Charles Darwin's theory of evolution was published in 1859 and has been dominant for more than 150 years, indicating that humans evolved from apes (Hale, 2012, Johnson, 2001). However, some studies demonstrated that humans shared a common ancestor with apes, but did not evolve from apes (e.g., Christensen, 2001, White et al., 2009). Recent neuroscience research showed that the human brain was organized differently from the ape brain (Premack, 2007, Holloway et al., 2009). In 1999 Preuss and Coleman's microscopic study first demonstrated that "In one layer of the human primary visual cortex, nerve cells were organized in a complex meshlike pattern very different from the simpler vertical arrays of cells in other primates." which was noted by Premack (2007, p. 13861). These studies demonstrated different points of view or possibilities concerning the origin of human life. How did human life actually originate? Besides, it remains unknown regarding origins of animals and plants. The atomic structures of the cell were discovered in the 20th century (Johnson, 2001) and the cosmic formation theory was just built (Yin, 2020). That is to say that in the Darwin's era, the atomic structures of the cell had not been discovered until the 20th century and the cosmic formation theory had not been built until recently and the microscopic study of human brains in the field of neuroscience had not started until 1999. With appearances of these new scientific discoveries and theories in the last more than one hundred years, it is significant to make new scientific research on origins of human life from a brand new perspective and on origins of animals and plants which are still gaps needing filling. The paper aims to clarify when and how humans, animals and plants originated and whether humans evolved from apes or shared a common ancestor with apes by providing a new theoretically-based analysis and explanations or a new theoretical point of view based on comic formation

David C. Wyld et al. (Eds): AIAP, IT, MoWiN, CCSEIT, CNSA, ICBB - 2022
pp. 119-136, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120410

theory, hereditary theory and related physical and chemical theories including atomic motion theory, electromagnetic field theory, atomic and molecular structure formation theory.

2. HOW DID LIVES OF HUMANS, ANIMALS AND PLANTS ORIGINATE?

Billions of years ago the black hole exploded under incredibly high temperatures (Hawking, 1974), which led to the birth of the universe on the basis of the cosmic formation theory (Yin, 2020). Simultaneously non-life forms of all life started to appear. That indicated that the earliest non-life forms of all life were born together with the non-life forms of stars (e.g., the Sun), planets (e.g., the earth, the mars), etc. at the beginning of the universe. All solid (e.g., metal, rock, wood, glass) and liquid (water, alcohol, etc.) matters spewing out of the black hole had been vaporized or in gaseous phase besides those original gases also from the black hole (Yin, 2020). All these gases contain compositions which make up all non-life matter (e.g., water, air) and objects (e.g., stars, planets) (Yin, 2020) and all forms of life (humans, animals, plants) in the universe. When all these gaseous matters spewed out of the black hole, they instantly scattered to all over the universe (Yin, 2020). Now we see that the universe is filled with stars, planets, asteroids, etc. Actually at that time the universe was also full of various non-life forms of life. However, survival of life was much more difficult and an individual life is much shorter, just as if it exists just for an instant compared with non-life matter and most of objects.

With temperature dropping, adjacent atoms which made up gases attracted each other and formed a big or small gaseous lump (Yin, 2020). The electromagnetic field theory indicates that opposite magnetic poles and electric charges attract each other and that any atom is a dipole with opposite magnetic poles and positive and negative electric charges (Blehl, 2006, Dorfmann and Ogden, 2014). When the temperature became colder, the gaseous lump also became colder and contracted and got smaller according to the principle of expanding when heated and contracting when cooled in general cases (Yin, 2020). It rapidly turned into one in liquid state and subsequently one in solid state and formed a star- or planet- or asteroid-shaped object (Yin, 2020). If the star- or planet- or asteroid-shaped object had the same compositions of the Sun, the earth or an asteroid, the Sun, the earth or the asteroid was born (Yin, 2020). Similar cases also happened to other stars, planets, asteroids and small and tiny objects (Yin, 2020). As such, the universe is full of all types of stars, planets, asteroids, small and tiny objects which lied in locations close to or far away from each other just as we see numerous stars in the nearby or far-off sky at night on a clear day. Of course, there must be numerous stars, planets, etc that we have not been able to observe or see by using the latest technology which humans have invented or created by now.

Similarly, adjacent atoms which made up gases also attracted each other and formed various extremely small or tiny gaseous lumps. Under atomic attractive forces, adjacent gaseous lumps similarly attracted mutually and connected and fused with one another and formed a bigger gaseous lump. Similar cases also happened to other adjacent gaseous lumps. If some of the gaseous lumps contained complicated atomic and molecular structures or compositions of the fertilized egg cell and related development environment of humans, animals, or plants (e.g., tree, flower, etc.), they made good preparations for the potential appearance of different forms of life. With the temperature persistently dropping, these gaseous lumps became colder and contracted and got smaller according to the principle of expanding when heated and contracting when cooled in general cases. They rapidly turned into ones in solid phase or in a normal state and gradually developed into ones with simple forms of life (human, animal or plant). They continued growing and became bigger just as a fetus curled up in the solid- or ball-like womb or just as a solid- or ball-like seed. If one of them contained the same compositions as a human or cow or sunflower, the human or the cow or the sunflower formed. Obviously, the human or cow embryo and fetus or sunflower seed embryo in the solid- or ball-like shape must have got enough

nutrients and had suitable temperatures and other necessary environmental conditions for survival, growth and maturity. If the newborn human or cow or sunflower could continue getting enough nutrients and having suitable temperatures and other necessary environmental conditions for survival and growth, the newborn human or cow or sunflower could continue surviving and growing. Similar cases happened to other humans, animals and plants. As such, there were man, woman, various animals and plants in the natural world. We have seen that seeds of flowers, grasses, trees and other plants are in geometric solids (e.g., sphere, oval, and cylinder) just like human and animal embryos and stars, planets, asteroids in geometrical solids (e.g., sphere and oval) despite the difference in width, length, height and weight (e.g., Brunken, de Wet and Harlan, 1977). In any case, only those humans or animals or plants that have the body structure and cell atomic and molecular structures and components which were consistent with the law of life survival and growth could survive and grow in addition to having enough nutrients, suitable temperatures and other necessary environmental conditions for survival and growth. If the body structure and cell atomic and molecular structures or components which were completely or basically inconsistent with the law of life survival and growth, they could not form normal living things like humans, animals or plants. They stopped growing and gradually or immediately died. The Mermaid described in the fairy tale or the Sphinx in Egypt most probably once existed as a natural organism. That is to say that they are most possibly neither made up by our ancestors based on their imagination nor artificial products created by scientists in a high-technology experiment, but once appeared as a natural form of life. Up to now, people still can see a variety of physical deformities in humans, animals and plants. As far as humans are concerned, these physical deformities are varied indeed, serious or not serious so that some of people with physical deformities can still be kept to live a normal life and their life quality and lifespan is not negatively affected by the physical deformities. However, unfortunately some others of them are more or less negatively affected by the physical deformities. They have a short or relatively long lifespan, depending on the severity of the deformity threatening life, whether they can get timely treatments and the status of medical science and technology development. Pantoja Zarza and Diez Morrondo (2014) demonstrated that skeletal and facial deformities in Mucolipidosis III start at the age of 3-5, belonging to genetically metabolic diseases and patients are expected to reach up to the fifth decade of life due to the slow evolution of the disease. Li et al.'s (2012) study revealed that inborn spinal deformities in rat offspring were caused by maternal Vitamin A deficiency, indicating that animals' deformities exist indeed. They noted that there are also other organ deformities, e.g., inborn kidney and heart deformities. Pruyn, Gartner and Harmon (2002) recorded that trees without stem deformities and diseases and broken tops were chosen as samples, indicating that deformities of trees as plants also really exist.

When humans, animals, trees, flowers, grasses or other plants were born, if they fell on the surface of the earth and got sufficient nutrients and suitable temperatures and other necessary environmental conditions for survival and growth, they could continue being alive until they died of illness or other causes. If some of them could not get sufficient nutrients or suitable temperatures or other necessary environmental conditions for survival, they would not survive. If they fell inside the earth, similarly they must have enough nutrients, suitable temperatures and other environmental conditions for survival. If they could get what were required for survival, they could survive. Otherwise, they would die. That indicated that humans, animals, flowers, grasses and trees and other plants could fall not only on the earth's surface but also in different parts inside the earth. They could also fall on and inside and even between other stars, planets, asteroids, etc. in the universe. If they have above mentioned conditions (enough nutrients, suitable temperatures and other necessary environmental conditions for survival) for survival, they could survive. If they lacked enough nutrients required for survival or the temperature was too hot or too cold to be suitable for survival, they would not survive. That is to say that human, animals, flowers, trees, grasses and other plants could be located in different parts of the universe. Whether they could survive or for how long they could live completely relied on survival

conditions they could get and whether their body structure and cells with atomic and molecular structures and components were consistent with the law of life survival. In addition to necessary survival conditions, when humans, animals, trees, grasses, flowers and other plants had normal reproductive abilities and other necessary conditions for reproduction, they would produce next generations. Some of them lasted until nowadays by reproduction of generations whereas others unfortunately failed to reproduce a new generation or reproduced one, two or more generations, but the latest generation died unexpectedly before reproduction or lost reproductive capability and hence did not keep reproduction of generations to last until now.

It is well-known that newborn babies are too young to get food independently or feed themselves. They acquire nutrients usually with their parents' or other people's help. When the earliest humans were born on the earth, how did they get nutrients for survival and growth? As the foregoing part mentioned, after black hole explosion, preparations for human and animal development and births started simultaneously and automatically. When the fertilized egg cell of human or animal in related development environment started differentiation and grew up into the fetus until human or animal was born, the time for this process is different. Actually, many animals before birth have shorter or much shorter period of time for their fetus formation, development and maturity than humans. When the earliest human was born, the earliest human baby must have been raised by a certain adult animal or animal family. That is to say that the animals including many mammalian animals were born on the earth or appeared in the world earlier than humans due to many mammalian animals' (e.g., wolf, leopard, dog and raccoon) shorter or much shorter embryo and fetus growth or gestation periods than humans in addition to many other animals. As a result, the earliest humans survived and grew with the animal's help. Research has shown that the average length of animal gestation time depends on the animal, e.g., wolf (63 days), raccoon (63 days), dog (63 days), leopard (95 days), lion (108 days), tiger (103 days), pig (114 days), goat (150 days) and sheep (150 days), shorter or even far shorter than human gestation period 274 days on average though some other animals, e.g., camel (410 days), donkey (365 days) have longer gestation periods than humans (Hill, 2021). Bettelheim (1959) noted that Ogburn believed that there was no sound evidence of human children raised by animals. However, people still heard true stories about contemporary human children raised by animals such as wolf, bear or leopard, etc. up to now. LaPointe (2005) noted that wolf-children meant that human children had been allegedly raised by animals, especially wolves and there were some records or reports, e.g., Lithuanian bear child in 1661, leopard child of India in 1920, sheep child of Ireland in 1672, Syrian gazelle child in 1946, pig boy of Holland in the 19th century, etc. Candland (1993) also noted that there was the earliest wolf-boy from Hesse in 1344, an Irish sheep-boy described in 1672, a Lithuanian bear-child observed to have been suckled by bears in 1661, and two girls named Kamala and Amala suggested to have been raised by wolves in northeastern India this century. It was reported that human children were raised by wolves in India (Bettelheim, 1959). Human babies raised by animals behaved more like an animal after they grew up (Bettelheim, 1959). These studies or reports indicated the case that human babies or children were raised by animals existed indeed. They also demonstrated that though people were told to be far away from those fierce animals (e.g., wolf, leopard, bear, etc.) for avoiding being harmed when they received education during childhood, several or multiple human children were raised by fierce and non-fierce animals even just in the past few centuries, which demonstrates the rarely seen or observed unusual behavior of animals related to humans and also provides a powerful link with the earliest human survival and growth.

When did humans leave animals that once helped them survive and grow and start to live independently? It is inferred that when human ancestors felt that their lives were threatened by fierce animal family members, animal partners, friends or/and neighbors especially when there were severe famines or the food which these fierce animals could seek to find was insufficient to satisfy their stomach needs, human ancestors felt secure no longer and started to escape from

these animals. They attempted and finally found a secure place to live in. Nowadays, it is well-known that people are generally afraid of directly staying together with fierce animals except animal keepers in the zoo. This case is obviously because fierce animals usually need a large amount of food daily and rarely can independently find so much food in places which humans live in, control and manage. As such, it is almost impossible without harming humans they directly meet except in general cases animal keepers in the zoo who take care of their food and are familiar to them. If the earliest human survival and growth were helped by non-fierce animals like sheep, etc., they left the animal family without being harmed by these animals. However, they must have been endangered by fierce animals nearby or coming from far-off places. As a result, they no longer felt secure and escaped from the place where they originally lived. Similarly, they tried and finally found a secure place and started to live independently. In any case, since humans left animals, they started to experience the process from crawling like animals to gradually standing up for more food, new tools, etc. for a more comfortable survival and growth.

3. DIFFERENCE IN THE BRAIN OF HUMAN, APE AND ANIMAL STARTED FROM THE ORIGIN OF THEIR RESPECTIVE LIFE

As above-mentioned, at the birth of the universe, under gravitational force of the atomic magnetic field, adjacent atoms attracted mutually and formed atomic and molecular structures of the fertilized egg cell and related development environment of humans or animals. The fertilized egg cell in the related development environment differentiated, grew and formed different tissues and organs such as brain, heart, lung, etc. Among these tissues and organs, brain plays a crucial role in intelligence activity. Actually, humans are more intelligent than apes whereas humans and apes are more intelligent than monkeys (Roth and Dicke, 2005). What lead to the difference in human, ape and animal intelligence? The paper also attempts to find the root causes leading to the difference in the human, ape and animal intelligence and clarify why humans neither evolved from apes nor shared a common ancestor with apes from a physiological perspective. In general cases, normal adult human brain is characterized by brain neuron property and counts in specialized areas and brain structure which match the most complicated cognitive functions involved in most complicated language, strong analysis, reasoning and judgment, etc. Normal adult gorilla brain is characterized by brain neuron property and counts in specialized areas and brain structure which match enough less complicated cognitive functions involved in less complicated language, weaker analysis, reasoning and judgment, etc. compared with the normal adult human brain. Normal adult animal brain is typical of brain neuron property and counts in specialized areas and brain structure which match simpler cognitive functions involved in simpler language, weaker analysis, reasoning and judgment, etc. compared with the normal adult gorilla brain. In other words, intelligence difference in normal adult human, ape and animal brains can be explained based primarily on brain neuron property and counts in specialized language and thinking areas and brain structure though there is also difference related to intelligence from other related aspects of brain and of the whole body.

Research has shown that human neocortical pyramidal neurons are three times larger than rodent pyramidal neurons, which is associated with the three times larger and more complex dendrites in human pyramidal neurons than in mouse or macaque (Mohan et al., 2015, see Goriounova et al., 2018), but also human pyramidal neuron dendrites receive one time more synapses than rodent pyramidal neuron dendrites (DeFelipe et al., 2002, see Goriounova et al., 2018). Besides, Mora-Bermúdez et al. (2016) showed that humans and apes have striking resemblances in cytoarchitecture, cell type composition and neurogenic gene expression programs. About 99% of human genetic matter is shared with gorilla and chimpanzees (Dobrovolsky, 2009). However, Schörnig et al. (2021) noted that human cortical pyramidal neuron dendrites are longer and more

branched than chimpanzee pyramidal neuron dendrites (Bianchi et al., 2013a) whereas pyramidal neuron synaptogenesis and development in humans and chimpanzees may be longer than in monkeys (Bianchi et al., 2013b). Thrombospondins are extracellular-matrix glycoproteins controlling synapse formation and neurite development, however, human cerebral cortex has approximately 6 times greater expression of thrombospondin 4 messenger RNA than the cerebral cortex of champanzees and macques (Cáceres et al., 2007), which indicated that more thrombospondins provided a basis for normal production of more synapses, and further demonstrated the physiological causes why cerebral cortex neuron synapses in humans were more than in apes and animals. Dendritic and spine number, size and morphology play one of crucial roles in integrating signals coming from individual synapses (Martínez-Cerdeño, 2016), which is closely correlated with cognitive ability. Neuron dendrites and dendritic spines significantly affect the connectivity of the brain and are responsible for long-term synaptic plasticity associated with memory and learning (Smrt and Zhao, 2010) or reasoning. Analysts learned to find solutions to complex tasks through mixed-initiative reasoning, integrated with previous knowledge, experience, etc. (Boicu et al., 2005), which indicates that learning requires reasoning in processing complex problems. The above studies also exhibited that neuron dendrites and dendritic spines are correlated with reasoning or thinking and memory. Other study also demonstrated that the integrated and functional roles of the neuron rely primarily on dendritic and synaptic structure, function and plasticity (Seong et al., 2015). The size of dendrites in these neurons is indeed positively associated with the whole cerebral cortex connectivity (Scholtens et al., 2014, van den Heuvel et al., 2015, see Goriounova et al., 2018), displaying that the larger dendrite in the neuron is closely correlated with the faster synaptic inputs (Testa-Silva et al., 2014) whereas synapse acts as a connection between two neurons (Babu, 2020). Larger neurons caused by larger dendrites can encode a larger number of rapidly changing temporal information more precisely in their output than smaller neurons caused by smaller dendrites (Goriounova et al., 2018). As a result, larger and more complex dendrites are closely linked with higher intelligence (Goriounova et al., 2018). Other research has also shown that pyramid neurons with dendrites and synapses are linked with higher cognitive functions (Goldman-Rakic, 1999, see Schörnig et al., 2021). Gemmell et al.'s (2012) study revealed that the volume or size of pyramidal neuron in hippocampal subfields CA1 and CA2 was 10%-20% smaller in those patients with delayed poststroke dementia or ischemic vascular dementia or mixed Alzheimer disease and vascular dementia syndrome or Alzheimer's disease compared with elderly groups with normal cognitive functions. This study indicated that neuronal dendrite size and complexity is significantly associated with normality of cognitive functions. However, Freeman et al (2008) made an examination of the cerebral cortex of 27 normal elderly people at the age of 56-103 and found that all these examined old people had normal cognitive functions and had no relatively reduced total neuronal numbers in frontal and temporal neocortex areas in spite of the relatively big age difference, but had obvious neuron shrinkage which signified the reduction of dendritic size and complexity. Actually, this study indicated that the reduction of dendritic size and complexity in a normal range did not negatively affect the normality of cognitive functions. Simultaneously, these studies also reflected the important role of the basic shape of brain neuronal dendrites in cognitive functions. Research has shown that abnormal or abnormally immature dendrite and synapse morphogenesis in human brain neurons can imply a variety of nerve developmental disorders (Seong et al., 2015), which indicated that abnormal or abnormally immature human brain neuron dendrites, synapses or property predicted the abnormalities of human cognitive ability and behavior. Studies on brain neurons of both humans and animals with autism and related neurodevelopmental diseases revealed the decreased size and number and altered morphology of dendrites and increased spine immature morphology (Martínez-Cerdeño, 2016), which indicated that the decrease of dendrite size and complexity and increase of immature structure of dendritic spine had exceeded the normal scope, which are correlated with autism, similarly displaying the importance of synapse morphology and size and mature synapse number in cognitive functions in addition to dendrite morphology, size and number. Research has

shown that abnormal alterations of the dendritic spine structure and distribution were found in diseases such as various mental retardations, etc. due to failure in formation of normal spines causing cognitive and motor deficits (Smrt and Zhao, 2010), which further indicated the importance of dendritic spine number, size and morphology in cognitive activities. All these above studies also indicated that the difference of human, ape and animal in brain neuron property supported the difference in their respective brain functions involved in language, analysis and reasoning and judgment, etc.

In addition, humans have much more neurons in the cerebral cortex than apes and other animals. Herculano-Houzel et al. (2014) demonstrated the African elephant brain contains 257 billion neurons, about three times more than the human brain on average, but 97.5% of the elephant brain neurons are located in the cerebellum and only 5.6 billion neurons are located in the elephant cerebral cortex, which is about one third of human cerebral cortex neuron counts. It is noted that elephant cortical neuron number is 11 billions and chimpanzee cortical neuron number is 6.2 billions (Roth and Dicke, 2005, see Roth, 2012). In any case, human cortical neurons are 12-15 billions, indeed more than the cortical neurons of whales and dolphins and elephants with much larger brains and of apes and other animals (Roth, 2012). Actually, humans have much more neurons in specialized language and thinking areas than apes and animals. The great reduction of neurons in hippocampus and cortex is robustly correlated with dementia characterizing Alzheimer's disease symptom, but there is no significant difference between the elders with normal cognitive functions and asymptomatic patients with Alzheimer's disease (Andrade-Moraes et al., 2013), which indicated that normal neuron number in specialized language and thinking areas in the human brain plays an indispensable role in normal language cognition, memory, processing and thinking. Demented patients with Alzheimer's disease presented language and cognitive impairment such as inability to process the discourse as a whole, marked lexical and discourse deficits at the early and intermediate stages, which are involved in temporal neocortex concerning lexical retrieval (Caramelli et al., 1998), which similarly demonstrated that these demented patients experience difficulty in language communication, memory, processing and cognition and thinking. Grouios and Ypsilanti (2011) demonstrated that Down syndrome is characterized by chromosomal abnormality, language and cognitive impairments, abnormal neuromotor, early aging, etc. frequently correlated with Alzheimer's disease symptom, which also similarly indicated that people with Down syndrome have difficulty in language communication, memory, thinking, movement, etc. Indeed, research has revealed that children with Down syndrome from birth had 20-50% fewer neurons, lower neuronal density and distribution particularly in cortical layers II and IV (Wisniewski, 1990), which also similarly indicated that lower than normal neuron number in certain areas of the cortex could hamper the normal work of human language and cognitive and neuromotor functions and further demonstrated that sufficient cortex neuron number in specialized language and thinking areas is indispensable for normal human language communication, memory, cognition and thinking. These studies also indicated that since humans have stronger language and thinking abilities than apes and animals, humans must have more neurons in specialized language and thinking areas than apes and animals. Premack (2007) noted that compared with apes, humans have many more slender tapered neurons (labeled VENs) situated in the frontoinsular cortex and the anterior cingulate cortex involved in emotional cognition, which was rediscovered by Hof and associates in 1999 through microscope. Their research result indicated that humans have much more brain neurons in emotional cognitive areas than apes. Functional Magnetic Resonance Imaging have revealed that memory and reasoning activities were demonstrated in certain areas of cerebral cortex (Ruff et al., 2003). Certain areas of cerebral cortex are indeed closely associated with language such as lexical and semantic retrieval, semantic integration, as well as memory and reasoning, etc. (Bunge et al., 2005). Prefrontal cortex is correlated with reasoning and solving problems (Bunge et al., 2005, see also Ruff et al., 2003). The Parietal cortex neurons are linked with the possible quantity addition and subtraction

activities supporting task decision-making (Yang and Shadlen, 2007). These studies indicated the crucial role of some areas of cerebral cortex in intelligence activities such as language and thinking, etc.

Premack (2007) showed that human minicolumns with 80-100 neuros bundled vertically located in the left planum temporale supported language and perhaps music, but were organized differently than those of chimpanzees and rhesus monkeys. The author also recorded that human minicolumns were 51μm on average, much wider than those of the chimpanzees and monkeys in 36 μm. This study also revealed that humans have the brain structure distinguished from the brain structure of apes and animals and corresponding neurons specializing in language and perhaps music to support much more complicated human language and perhaps music skills than apes and animals. The whole human prefrontal cortex neurons in layer III are significantly more widely spaced than those in great apes (Teffer and Semendeferi, 2012), which further indicated the difference in the brain structure of humans and apes. In fact, research had found that early human brain was organized differently from ape brain (Holloway et al., 2009). Passingham (2009) noted that humans and macaque monkeys showed a difference in the proportions of different regions and microstructures of their brains. He also noted that "The prefrontal cortex, defined as the granular frontal cortex forms 28.5% of the neocortex in the human brain but only 11.3% in the macaque brain."; "When related to the brain as a whole, the frontal polar cortex, area 10 is proportionately twice as large in the human brain as in that of the chimpanzee." (p. 2). These studies suggested that there was a difference in the brain structure of humans, apes and animals, which match their different number and property of specialized neuron involved in language, analysis, reasoning, judgment, etc.

Many studies have displayed the difference among humans and apes and other animals in cognitive capability and behavior. After taught by humans, animals could connect artificial labels with objects or properties of objects, indicating animals could learn words and corresponding meanings (Cheney and Seyfarth, 2010). Meijer's (2021) observation demonstrated that the gray parrot Alex knew over 100 words. Larger-brained dolphins and smaller-brained sea lions could understand novel and unique gesturally conveyed messages under human instruction (Schusterman and Gisiner, 1988). However, animal communication seems to be limited to message conveyance and is constrained to a different extent (severe extent, e.g. bees, less severe, e.g., parrot) based on different animals (Dobrovolsky, 2009). Human cortex information processing capabilities are stronger compared with animals, even those large-brained mammals such as elephants, dolphins and whales (Roth, 2012). Systematic investigations have recorded that the great apes can learn, comprehend and use symbols to exchange information and use relatively simple rules to organize these symbols (Rumbaugh et al., 2009). Other research has showed that gorillas Koko could understand around 1000-word sign language and around 2000 spoken English words taught by humans, which signified that she could express her own ideas in sign language, and gorilla Michael learned about 600-word sign language at a similar pace (The Gorilla Foundation, 2018). It noted that other apes also have similar capabilities of learning and acquiring a sign language at this level in other ape studies (see also Rumbaugh and Savage-Rumbaugh, 2001). However, little evidence showed that apes could combine words taught by humans into a sentence like phrases (Cheney and Seyfarth, 2010). Though sign language invented and learned by apes is much more complicated than one, two or a few words spoken language invented and learned by apes, their sign language is still much simpler than human sign language. Bonvillian and Patterson's (1997) investigation on gorilla Koko and young deaf children at similar ages showed that these children learned sign language and gestures more rapidly than the gorilla and the gorilla's vocabulary size and her sign language complexity were relatively limited and could not keep up with those of children especially over time. In fact, human language is more productive and more flexible in controlling and using these symbols than any of known ape and all other animal communication systems (Dobrovolsky, 2009). Brinck and Gärdenfors (2001)

demonstrated that humans have higher attentional complexities to become more cooperative and have more ways to achieve a goal than apes, indicating that higher cognitive ability of humans than apes promoted their higher cognitive behaviors. Whiten (2013) noted that apes have stronger imitative abilities (e.g., gestures) than monkeys, e.g., apes could imitate how to crack a walnut and other nuts using a stone hammer whereas monkeys failed, which demonstrated that higher cognitive ability of apes than animals like monkeys worked in a way matching their higher cognitive behavior. Indeed, humans are more intelligent than great apes, which are more intelligent than monkeys and other animals (Roth and Dicke, 2005). These studies also demonstrated the difference in cognitive capability and behavior among humans and apes and animals, which match with their respective property and counts of brain neuron in specialized language and thinking areas and corresponding brain structure.

It is not excluded that a very small number of animals show the property and count of more brain neurons more beneficial to specialized language and thinking accompanied with corresponding brain structures compared with the vast majority of animals of the same genre. However, generally speaking, birds of a feather flock together. When very few animals with unusually cleverer brains grew together with their genre of population with normal animal brains, unusual advantages of their brains could not be constantly developed or inherited through generation after another. In other words, it is unrealistic that they passed on their intellectual characteristics to their offspring of generation after generation since such animals are rare or in a very small number. As far as humans are concerned, if some people congenitally partly or even almost did not have property or enough number of normal human brain neuron in specialized language and thinking areas for normal thinking and language learning, they could not belong to normal human groups. For example, a baby was born mentally disabled, which signified that when he/she was old enough to go to school or even grew up into adulthood, he/she could not learn how to speak or understand what other people said like a mentally normal child or adult. Some of such babies cannot live until adulthood due to multiple abnormal health problems; after growing up into adulthood, generally some others had fewer or no offspring. As a result, such human intellectual heritability stopped or gradually decreased. Anencephaly is a neural tube birth defect which causes lack of a major portion of the brain, and infants with anencephaly die before birth or within a few hours or days of birth (Diniz, D. (2007, see Ballantyne et al., 2009, see also Ryskamp, 2010), rarely live past one year, only one recorded female infant lived until the age of four (Foreman, 1999, see Ryskamp, 2010), which display that persons with so severe brain defects have a very short lifespan and cannot live until adulthood to have offspring. Such infants are generally also accompanied with other abnormal health problems such as the absence of the neck and of cranial vault, bulging of eyes, abnormal spine (spinal cord absence or dysplasia), etc. (Korpova et al., 2012). Children born with Down syndrome displaying language and cognitive and other related health impairments can live to about 47 years old on average in 2007 whereas in 1960 such children can live only to about 10 years old on average (CDC, 2020), which exhibit an obvious increase of survival rates during approximately 50 years. Other research also demonstrated that people with Down syndrome generally die 28 years earlier than the general population (O'Leary et al., 2018). However, in any case, research has demonstrated that women and girls with birth defects had reduced survival rates and less possibility of having a child compared with those women and girls without birth defects in Norway (Skjaerven et al., 1999). Similarly, research has also shown that males with registered birth defects in Norway have higher mortality rates and survivors have less possibility of having a child compared with those males without birth defects (Lie et al., 2001). The registered birth defects in Norway included an estimated 60% of Down syndrome and an estimated 80% of cleft lip cases, anencephaly, spina bifida, etc. (Lie et al., 1994, see Lie et al., 2001). These studies indicated that though there are still mentally disabled persons in the world, such population is in a very small number compared with the vast majority of cognitively normal population. In fact, it has been extensively accepted that learning and speaking a complicated language is one of remarkable characteristics of humans

distinguished from animals from the ancient time to present. Therefore, during billions of years of human evolution, those people who could not learn to speak a human language were gradually reduced. Nowadays, cognitively normal people we see have the ability to learn to speak at least one human language. As such, present cognitively normal humans on the earth generally have no big difference in intelligence, but have an essential difference in intelligence from the gorilla and animal.

During billions of years of physical and mental evolution, the brains of humans, apes and animals became cleverer, but humans, apes and animals have an essential difference in the earliest brain structure formed and the number and property of earliest brain neuron in specialized areas involving in language, analysis and reasoning and judgment. That is to say that human did not evolve from apes. Human became humans, which were decided by the human brain structure and the property and number of brain neuron in specialized language and thinking areas formed at the birth of the universe. Such brain structures and minimum counts and property of brain neuron in specialized language and thinking areas for becoming a human decided that humans could gradually grow to invent and learn new things (e.g., complicated and abstract languages, complicated tools) whereas the brain structures of ape and animal and the numbers and properties of their brain neuron in specialized language and thinking areas decided that they could speak only one or two or a few words language each time invented or learned by themselves and use simple tools.

Though humans and animals became cleverer during billions of years of evolution, the total trend of intellectual development progressed very slowly and intellectual evolution of family generations was subtly up and down like a tiny wave, but on the whole gradually went like an upward spiral. Actually human intellectual development is complicated, depended primarily on heredity and external stimulations and nutrients. Killeen (2004) noted that individual traits such as physical appearance, personality, intellect, etc. are affected by heredity, environmental factors and nutrients (e.g., whether to eat high-fat diets and smoke), indicating that human mental and physical characteristics is closely associated with heredity, external factors including environmental stimulation, etc. and nutrient factors including selected food nutrient quality, quantity and variety, etc. Shakeshaft et al.'s (2015) study also demonstrated that high intelligence is determined by genetic and environmental factors related to intellectual development. It is well-known to plant and harvest melons and plant and harvest beans, which is described as the genetic law of living things. It indicated that living things pass on traits to their offspring. Human as one of living things, its genetic law is no exception. That is to say human parents can also pass on their mental and physical characteristics to their children. The hereditary characteristic of intelligence is determined by a large number of genes of very small effect (Plomin and Deary, 2014, see Shakeshaft et al., 2015). Savage et al.'s (2018, p. 912) genome-wide study of 269867 individuals revealed that "Associated genes are strongly expressed in the brain, specifically in striatal medium spiny neurons and hippocampal pyramidal neurons.", indicating strong genetic correlation with brain neurons linked with intelligence. Trzaskowski et al.'s (2014) survey on DNA (Deoxyribonucleic acid with genetic codes) of 3000 unrelated children aged between 7 and 12 showed that genetic influence is closely correlated with children's intellectual quotient. Marioni et al.'s (2014) research on DNA-derived heritability of 6815 unrelated Generation Scotland participants at median age 57 years demonstrated that heritability of intelligence partly impacted cognitive ability associated with education and was compatible with other researchers' results: education extent was primarily environmentally or genetically driven. If father and mother have similar intellectual levels, children could be cleverer than their parents if the intellectual heredity from their parents had good variation or children received more external stimulations than their parents (e.g., sufficiently more education, much more complicated social communications and individual experiences). Research has shown that limited but converging evidence displayed that process-based complex working memory training have a positive impact

on academic capabilities (Titz and Karbach, 2014), which indicated the beneficial role of external stimuli like training or education in improvement of cognitive and academic ability. Human babies were raised by non-human parents or socially and physically isolated when they grew up. They could neither speak a human language nor behave as well as a normal human, but more like an animal after they grew up (Bettelheim, 1959), indicating that their cognitive abilities in language and behavior did not develop as a normal human. When they were raised by animals or socially and physically isolated when growing up, their advanced potential of human intelligence or cognitive ability were not developed as being normally raised by human with age, which demonstrated the significant impact of human external stimulations like complicated social communications and rich experiences on brain development in addition to the minimum number and property of normal human brain neuron in specialized language and thinking areas and corresponding brain structures and other related factors for being able to speak a human language.

Some children have intellectual levels similar to those of their parents if they had development experiences similar to those of their parents and the intellectual heredity from their parents had no detectable variation and others are less clever than their parents if the intellectual heredity from their parents had bad mutations or they have received sufficiently fewer external stimulations than their parents (e.g., sufficiently less education due to the war, sufficiently less and simpler social communications and individual experiences). Plomin and Von Stumm (2018) recorded that recent research revealed that difference in 20% of heritability of intelligence is due to inherited genome sequence differences, indicating that heritability of intelligence has good or bad genetic variants. Arslan and Penke (2015) noted that intelligence disability is caused mainly by new mutation, which was regarded as a bad variant. If father and mother have some or bigger difference in intelligence, children maybe inherited their cleverer father's or mother's intelligence or less clever mother's or father's intelligence or almost half of their father's and mother's intelligence each, etc. As such, children are as clever as their father or mother or as less clever as their father or mother or between their parents' intellects, etc. if there were no detectable mutations in intellectual heritability and they had developmental experiences similar to those of their parents. That is to say that an intelligent man or woman does not ensure that they would have a more intelligent child than he or she. With the progress of human society and constant technological innovations, in most cases new generations often had more external stimulations (e.g., more extensive information reading and knowledge learning and better food nutrients to develop their brain than their parents), but some exceptional cases also happened. For example, children during the war had fewer opportunities to learn new knowledge, obtain extensive information and good food nutrients, etc. to develop their brain than their parents who never experienced a war or experienced a war over a much shorter time. Since there were written records several thousands of years ago, it is well-known that wars had used to happen, far more frequently than nowadays, which hampered affected children from developing their brains better. Kesternich et al.'s (2012) study noted that World War Two caused severe famines in some regions, which resulted in the cognitive and mental problems of individuals including children and adults as shown by Neugebauer et al. (1999) and Rooij et al. (2010). Lack of proteins cadherins and catenins can lead to abnormal dendrite and synapse morphogenesis, which predicted various disorders of nerve development (Seong et al., 2015) closely associated with cognitive and mental problems. Besides, when children were very young, if both or either of parents were seriously ill or lost employment over a long period of time and the family had no other financial sources or aids which are enough to support their lives, their children's healthy diets would be negatively affected, which were also similarly disadvantageous to their children's brain development.

Moreover, more than 2000 years ago, some of books and articles and poems, etc. of China's and many other countries' scholars extensively read by generation after another in various fields

showed high intelligence and wisdom. For example, Aldrete (2011, p. 2) recorded that between 500 BC and 600 BC, new thinkers around the world occurred and wrote new philosophies and religions, e.g., "Confucianism and Daoism in China, pre-Socratic philosophy in Greece, Buddhism and Jainism in India, and Zoroastrianism in Persia", which still have a significant influence on present day related countries. Han and Qiao (2009) noted that the book "Wujing Tongyi" recorded the mechanism of solar eclipses to improve the ancient calendar around 2000 years ago and provided valuable contributions to research on present day related fields (e.g., astrophysics concerning the rotation rate of the earth). Numerous poems were created, e.g., The Tale of the Shipwrecked Sailor 2040-1782 BC（Mark, 2012), the Vedas around 1500 BC (Mark, 2020), Shijin (The Book of Poetry) 1100 BC – 600 BC (Chen et al, 2008) and Homer's epic poems "Iliad" 800 BC and "Odyssey" around 800-600 BC (Finkelberg, 2011, see also Aldrete, 2011). Invention of written language based on written records kept around 3500 BC (Wei, 2012) was not easy, evidently requiring high intelligence. Around 2500 BC use of boat was discovered (Dixit, Hazarika and Davim, 2017). Invention of boat enabled people to easily travel on water for visiting far-off places, meeting new people, exchanging goods and ideas with more people, live a life on water and find more food in sea, rivers and lakes, etc. About 2000 years ago, many other deeply impressive inventions also showed high creativity. For example, inventions of oil pressing device, water-driven grinder for food processing, weaving technology, paddle blade machines for continuous water drawing, crossbows, diverse manufactured metal products (Hsiao and Yan, 2014) and paper making and compass and three-mast ship and primitive steam engine and windmills for producing motive sources, e.g., generating electricity, milling grains, lifting water from wells, sawing wood, etc. (Dixit, Hazarika and Davim, 2017) further provided convenience and comfort for human life (e.g., travelling at sea and on land, agricultural work, everyday life, keeping records, hunting animals and against invasion). Around simultaneously the construction of the Step Pyramid also showed quite astonishing achievements (Aldrete, 2011).

Undoubtedly, all these astonishing achievements achieved at least 2000 years ago years required creative ideas and high cognitive abilities. Inventions of all types of tools and technologies and oral and written languages in ancient times have left deep traces for development of modern civilization. That is to say that in the past several thousand years, on the whole human intellect had no obvious change. As such, it is referred that during past billions of years, globally human intellectual development must be gradual and very slow. Research demonstrated that 2-4 million years ago fossil hominid brain sizes varied from around 385 to 1700 ml whereas modern human brain size is around 1400 ml on average (Holloway et al., 2009), indicating that there was no remarkable change in human brain size during the past 2-4 million years and different persons have similar or different brain sizes. That is also to say that human intellectual development depended primarily on heredity and external stimulations and good nutrients, which did not result in a radical or essential change in intellectual increase on the whole and further indicated that the formation of complicated molecular structures and an enough large number of brain neuron in specialized language and thinking areas billions of years ago played a crucial role in difference of human brain from ape and animal brains because the big difference in the number and property of brain neuron in specialized language and thinking areas and brain structure among humans and apes and animals decided that humans could find and generate much more external stimulations and seek and produce much better nutrients and invent and use much more complicated languages and tools than apes and animals except common natural external stimulations (e.g., complicated mountainous and hilly areas).

4. HUMAN SKIN COLOR AND SUNLIGHT SHINNING

Since humans can fall on and inside the earth or other planets, etc. at the beginning of the universe as the foregoing mentioned, why don't we now see native people in the same place have

no different skin colors except some migrants and their descendants? It is observed that human skin colors seem to be closely correlated with sunlight. That is to say that due to difference in the amount of the sunlight shining over their bodies, it is seen that human skin color gradually varies from white in Europe to light brown and brown in North Africa to black in Central and Southern Africa, also from North Pole with the least sunlight to North Africa with mild sunlight and then to the vicinity of the equator with the most sunlight. Similarly, human skin color also varies from light yellow in the northeast of China to brown or dark brown in India, also gradually from North with less sunlight to South with the more sunlight in Asia. Though Western Europe and the northeast of China are in the similar latitude, Western European skin color is white while the skin color of the northeasterner of China is light yellowish. That indicated that Western Europeans have a lighter skin color than the northeasterners of China, which is also closely associated with the amounts of the sunlight shining over them. Western Europe with the oceanic climate has sufficiently less sunlight compared with the northeast of China with the monsoon climate. That further demonstrated that human skin color is closely associated with the amounts of sunlight shining over their skin. In addition, human skin color is also closely correlated with the heredity mainly from or between their parents' skin colors. Prud'homme-Généreux (2011) showed that parents' genetic components including skin color, etc. will be disproportionally passed on to their offspring. As a result, difference in human skin color is decided primarily by the total amount of sunlight shining on the skin of generations after family generations except for possible genetic mutation.

An inference is made that if humans with white or yellow skin are persistently shone by the sunlight as in Africa, their skins would gradually become dark or even black in millions or billions of years. However, with the development of industrial revolution in the past few centuries and the gradual spread of agricultural mechanization, people who live in Africa where there is the largest amount of sunlight are rarely exposed to as much sunlight as people there used to. People stay indoors and are shielded from the Sun in most of time. As a result, if people with white or yellow skin migrate and live there, in millions or billions of years their skin colors should have no obvious change or get a little darker than before at most, nevertheless, depending on individual amounts exposed to sunlight. African black people's skin color should become some lighter (e.g., light black, even brown) after a long time (e.g., millions or billions of years) if they stay outdoors and are not exposed to the Sun in most of time. Similarly, the skin color of descendants of people with black or yellow skin who migrated to Europe would also get lighter than before after a long time (e.g., millions or billions of years).

It is also inferred that change of skin color caused by change of amounts of sunlight shining over immigrants' descendants should link mutations of skin color genes after numerous generations. If ancestors migrated to places with sunlight shining less or much less than those places where they lived before migration, the skin color of the descendants after their numerous generations would change more particularly when migration ancestor used to work under the Sun e.g., in the farmland. Besides, it was also possible that billions of years ago when humans were born, humans carried different skin color genes, which presents different skin colors. Millions or billions of years, similarly migration descendant's skin color varied to different extents, depending also primarily on the amounts of sunlight which their numerous older generations once received. As a result, we see that native people in the same place have the same or similar skin color except some immigrants and their descendants. In either case, research is also expected to confirm that change of skin color gene(s) should be able to lead to immediate change of the skin color.

5. SUMMARY AND CONCLUSIONS

With black hole explosion under incredibly high temperatures leading to cosmic information billions of years ago, all matters had been in gaseous phase. With temperature dropping, adjacent atoms which made up gases attracted each other and formed a variety of big, small or tiny gaseous lumps. Under atomic attractive forces, adjacent gaseous lumps attracted mutually and connected and fused with one another into a bigger gaseous lump. With the temperature persistently dropping, the gaseous lump became colder and contracted and got smaller according to the principle of expanding when heated and contracting when cooled in general cases. Gradually it developed and formed a human- or animal- like embryo and fetus or a plant-like embryo and seed. If it had the same compositions as a human, cow or sunflower, the human, the cow or the sunflower formed. Similar cases happened to other humans, animals and plants. Due to their much shorter gestation periods than humans, many mammalian animals (e.g., wolf, leopard, dog and raccoon) came to the world or were born on the earth much earlier than humans. As a result, the earliest humans survived and grew with the adult animal's help. That is to say that the earliest non-life forms of all life were born together with stars, planets, etc. Or preparations for human and animal and plant development and birth started simultaneously and automatically at the early period of the universe. Humans neither evolved from apes nor shared a common ancestor with apes.

ACKNOWLEDGEMENTS

I would very much like to thank anonymous supporters for their strong scientific spirit and great kindness and also for providing Bibliography for me reading online free of charge. Without their persistent support, the completion of the paper would be impossible. Many thanks!

REFERENCES

[1] Aldrete, G. S. (2011). History of the Ancient World: A Global Perspective. Virginia: The Great Courses.

[2] Andrade-Moraes, C. H., Oliveira-Pinto, A. V., Castro-Fonseca, E., da Silva, C. G., Guimarães, D. M., Szczupak, D., Parente-Bruno, D. R., Carvalho, L. R. B., Polichiso, L., Gomes, B. V., Oliveira, L. M., Rodriguez, R. D., Leite, R. E. P., Ferretti-Rebustini, R. E. L., Jacob-Filho, W., Pasqualucci, C. A., Grinberg, L. T. and Lent, R. (2013). Cell Number Changes in Alzheimer's Disease Relate to Dementia, Not to Plaques and Tangles. Brain, 136, 3738–3752.

[3] Arslan, R. C. and Penke, L. (2015). Zeroing in on the Genetics of Intelligence. Journal of. Intelligence, 3, 41-45; doi:10.3390/jintelligence3020041

[4] Babu, K. (2020) Sticking and Signalling at the Synapse. Resonance, 25, 419–429. https://doi.org/10.1007/s12045-020-0954-6

[5] Ballantyne, A., Newson, A. J., Luna, F. and Ashcroft, R. (2009) Prenatal Diagnosis and Abortion for Congenital Abnormalities: Is It Ethical to Provide One without the Other? American Journal of Bioethics, 9(8): 48-56

[6] Bettelheim, B. (1959). Feral Children and Autistic Children. American Journal of Sociology, Vol. 64, No. 5, 455

[7] Bianchi, S., Stimpson, C. D., Bauernfeind, A. L., Schapiro, S. J., Baze, W. B., McArthur, M. J., Bronson, E., Hopkins, W. D., Semendeferi, K., Jacobs, B., Hof, P. R., Sherwood, C. C. (2013a) Dendritic Morphology of Pyramidal Neurons in the Chimpanzee Neocortex: Regional Specializations and Comparison to Humans. Cerebral Cortex, 23:2429–2436. https://doi.org/10.1093/cercor/bhs239

[8] Bianchi, S., Stimpson, C. D., Duka, T., Larsen, M. D., Janssen, W. G., Collins, Z., Bauernfeind, A. L., Schapiro, S. J., Baze, W. B., McArthur, M. J., Hopkins, W. D., Wildman, D. E., Lipovich, L., Kuzawa, C. W., Jacobs, B., Hof, P. R. and Sherwood, C. C. (2013b) Synaptogenesis and Development of Pyramidal Neuron Dendritic Morphology in the Chimpanzee Neocortex Resembles Humans. PNAS, 110, Suppl 2:10395–10401. https://doi.org/10.1073/pnas.1301224110

[9] Blehl, M. F. (2006). Electric and Magnetic Power Paperback. USA: Pearson Scott Foresman.
[10] Boicu, M., Tecuci, G., Ayers, C., Marcu, D., Boicu, C., Barbulescu, M., Stanescu, B., Wagner, W., Le, V., Apostolova, D. and Ciubotariu, A. (2005) A Learning and Reasoning System for Intelligence Analysis. In Proceedings of the Twentieth National Conference on Artificial Intelligence, AAAI-05, Pittsburgh, Pennsylvania, USA, July 9-13, 2005
[11] Bonvillian, J. D. and Patterson, F. G. P. (1997). Sign Language Acquisition and the Development of Meaning in a Lowland Gorilla. In: C. Mandell and A. McCabe (Editors) The Problem of Meaning: Behavioral and Cognitive Perspectives, pp. 181-219, Elsevier Science B. V.
[12] Brinck, I. and Gärdenfors, P. (2001). Co-operation and Communication in Apes and Humans. Lund University Cognitive Studies, 88, ISSN 1101 8453.
[13] Brunken, J., de Wet, J. M. J. and Harlan, J. R. (1977). The Morphology and Domestication of Pearl Millet. Economic Botany, Vol. 31, No. 2, 163-174.
[14] Bunge, S. A., Wendelken, C., Badre, D. and Wagner, A. D. (2005) Analogical Reasoning and Prefrontal Cortex: Evidence for Separable Retrieval and Integration Mechanisms. Cerebral Cortex, Vol. 15, No. 3, 239–249 https://doi.org/10.1093/cercor/bhh126
[15] Cáceres, M., Suwyn, C., Maddox, M., Thomas, J. W., Preuss, T. M. (2007) Increased Cortical Expression of Two Synaptogenic Thrombospondins in Human Brain Evolution. Cereb Cortex, 17(10):2312–2321. doi: 10.1093/cercor/bhl140
[16] CDC (Centers for Disease Control and Prevention) (2020) Data and Statistics on Down Syndrome. Available from: https://www.cdc.gov/ncbddd/birthdefects/downsyndrome/data.html
[17] Candland, D. K. (1993). Feral Children and Clever Animals: Reflections on Human Nature. New York: Oxford University Press.
[18] Caramelli, P., Mansur, L. L. and Nitrini, R. (1998) Chapter 32 - Language and Communication Disorders in Dementia of the Alzheimer Type. In: B. Stemmer and H. A. Whitaker (eds.) Handbook of Neurolinguistics, pp. 463-473. Elsevier Inc: Academic Press
[19] Chen, J. Y. and Jiang, J. Y. (Translator, Modern Chinese), Wang, R. P. (Translator, English) (2008) The Book of Poetry (Chinese-English), 2 Volumes, 727 pages. Changsha: Hunan People's Publishing House.
[20] Cheney, D. L.and Seyfarth, R. M. (2010). Why Animals Don't Have Language? (The Tanner Lectures on Human Values at Cambridge University March 10-12, 1997).
[21] Christensen, K. R. (2001). The Co-evolution of Language and the Brain: A Review of Two Contrastive Views (Pinker & Deacon). Grazer Linguistische Studien, 55, 1-20.
[22] DeFelipe, J., Alonso-Nanclares, L. and Arellano, J. I. (2002) Microstructure of the Neocortex: Comparative Aspects. Journal of Neurocytology, 31: 299–316. https://doi.org/10.1023/A:1024130211265
[23] Diniz, D. (2007). Selective Abortion in Brazil: The Anencephaly Case. Developing World Bioethics, 7(2), 64-67.
[24] Dixit, U.S., Hazarika, M. and Davim, J. P. (2017). A Brief History of Mechanical Engineering. Springer International Publishing Switzerland.
[25] Dobrovolsky, M. (2009) Chapter 16 Animal Communication. Pearson Education Canada
[26] Dorfmann, L. and Ogden, R. W. (2014). Nonlinear Theory of Electroelastic and Magnetoelastic Interactions. Springer Science + Business Media New York.
[27] Finkelberg, M. (ed.) (2011) The Homer Encyclopedia, Vol. III, 706-954. Blackwell Publishing Ltd.
[28] Foreman, M. W. (1999). Christianity & Bioethics: Confronting Clinical Issues. Joplin, Missouri: College Press Publishing Co.
[29] Freeman, S. H., Kandel, R, Cruz, L., Rozkalne, A., Newell, K., Frosch, M. P., Hedley-Whyte, E. T., Locascio, J. J., Lipsitz, L. A. and Hyman, B. T. (2008) Preservation of Neuronal Number Despite Age-related Cortical Brain Atrophy in Elderly Subjects without Alzheimer Disease. Journal of Neuropathology Experimental Neurology, 67: 1205– 1212.
[30] Gemmell, E., Bosomworth, H., Allan, L., Hall, R., Khundakar, A., Oakley, A. E., Deramecourt, V., Polvikoski, T. M., O'Brien, J. T. and Kalaria, R. N. (2012) Hippocampal Neuronal Atrophy and Cognitive Function in Delayed Poststroke and Aging-Related Dementias. Stroke, Vol. 43, No. 3, 808–814 https://doi.org/10.1161/STROKEAHA.111.636498
[31] Goldman-Rakic, P. S. (1999) The "Psychic" Neuron of the Cerebral Cortex. Annals of the New York Academy of Sciences, 868: 13–26. https://doi.org/10.1111/j.1749-6632.1999.tb11270.x
[32] Goriounova, N. A., Heyer, D. B., Wilbers, R., Verhoog, M. B., Giugliano, M., Verbist, C., Obermayer, J., Kerkhofs, A., Smeding, H., Verberne, M., Idema, S., Baayen, J. C., Pieneman, A.

W., de Kock, C. P. J., Klein, M. and Mansvelder, H. D. (2018) Large and fast human pyramidal neurons associate with intelligence. eLife, 7:e41714 DOI: 10.7554/eLife.41714

[33] Grouios, G. and Ypsilanti, A. (August 29th 2011). Language and Visuospatial Abilities in Down Syndrome Phenotype: A Cognitive Neuroscience Perspective. In: S. Dey (ed.) Genetics and Etiology of Down Syndrome, IntechOpen, DOI: 10.5772/20483.

[34] Hale, P. J. (2012). Darwin's Other Bulldog: Charles Kingsley and the Popularisation of Evolution in Victorian England. Science and Education, 21, 977-1013.

[35] Han, Y. B. and Qiao, Q. Y. (2009). Records of Solar Eclipse Observations in Ancient China. Science in China Series G: Physics, Mechanics and Astronomy, Vol. 52, No. 11, 1639-1645.

[36] Hawking, S. W. (1974). Black hole explosions? Nature Volume 248, pages 30-31.

[37] Herculano-Houzel, S., Avelino-de-Souza, K., Neves, K., Porfírio, J., Messeder, D., Mattos Feijó, L., Maldonado, J. and Manger, P. R. (2014) The Elephant Brain in Numbers. Frontier in Neuroanatomy, 8:46. doi: 10.3389/fnana.2014.00046

[38] Hill, M. A. (2021, Jan. 11) Embryology K12 Animal Development Times. Retrieved from https://embryology.med.unsw.edu.au/embryology/index.php/K12_Animal_Development_Times

[39] Holloway R. L., Sherwood C. C., Hof P. R. and Rilling J. K. (2009) Evolution of the Brain in Humans – Paleoneurology. In: Binder M.D., Hirokawa N., Windhorst U. (eds) Encyclopedia of Neuroscience. Springer, Berlin, Heidelberg, 1326-1338.

[40] Hsiao, K. H. and Yan, H. S. (2014). Mechanisms in Ancient Chinese Books with Illustrations. History of Mechanism and Machine Science 23. Springer International Publishing Switzerland.

[41] Johnson, R. (2001). Biology (6th edition). New York: McGraw-Hill Science/Engineering/Math.

[42] Kesternich, I., Siflinger, B., Smith, J. P. and Winter, J. K. (2012). The Effects of World War II on Economic and Health Outcomes across Europe, IZA DP No. 6296

[43] Killeen, A. (2004). Principles of Molecular Pathology. Totowa, NJ: Humana Press.

[44] Korpova, M., Miklosova, M., Dankovcik, R., Dudas, M. and Sivrev, D. (2012) Etiopathogenetic Similarities and Differences in Neural Tube Deffects. Trakia Journal of Sciences, Vol. 10, Suppl. 2, 262-266

[45] LaPointe, L. L. (2005). Feral Children. Journal of Medical Speech-Language Pathology, 13(1), VII-IX.

[46] Lie, R. T., Heuch, I. and Irgens, L. M. (1994) Maximum Likelihood Estimation of the Proportion of Congenital Malformations Using Double Registration Systems. Biometrics, 50, 433-444

[47] Lie, R. T., Wilcox, A. J. and Skjaerven, R. (2001) Survival and Reproduction among Males with Birth Defects and Risk of Recurrence in Their Children. JAMA, 285(6):755-60. doi: 10.1001/jama.285.6.755.

[48] Li, Z., Shen, J., Wu, W. K. K., Wang, X., Liang, J., Qiu, G. X. and Liu, J. M. (2012). Vitamin A Deficiency Induces Congenital Spinal Deformities in Rats. PLOS ONE, Vol. 7, No. 10, e46565.

[49] Marioni, R. E., Davies, G., Hayward, C., Liewald, D., Kerr, S. M., Campbell, A., Luciano, M., Smith, B. H., Padmanabhan, S., Hocking, L. J., Hastie, N. D., Wright, A. F., Porteous, D. J., Visscher, P. M. and Deary, I. J. (2014). Molecular genetic contributions to socioeconomic status and intelligence. Intelligence, Vol. 44, 26-32.

[50] Mark, J. J. (2012) The Tale of the Shipwrecked Sailor: An Egyptian Epic. World History Encyclopedia.

[51] Mark, J. J. (2020) The Vedas. World History Encyclopedia.

[52] Martínez-Cerdeño, V. (2016) Dendrite and Spine Modifications in Autism and Related Neurodevelopmental Disorders in Patients and Animal Models. Developmental Neurobiology, Vol. 77, No. 4, 393-404 https://doi.org/10.1002/dneu.22417

[53] Meijer, E. (2021) Animal Languages. Cambridge: The MIT Press.

[54] Mohan,H., Verhoog, M. B., Doreswamy, K. K., Eyal, G., Aardse, R., Lodder, B. N., Goriounova, N. A., Asamoah, B., Brakspear, A. B. B., Groot, C., van der Sluis, S., Testa-Silva, G., Obermayer, J., Boudewijns, Z. S., Narayanan, R. T., Baayen, J. C., Segev, I., Mansvelder, H. D. and de Kock, C. P. (2015) Dendritic and Axonal Architecture of Individual Pyramidal Neurons across Layers of Adult Human Neocortex. Cerebral Cortex, 25: 4839–4853. https://doi.org/10.1093/cercor/bhv188

[55] Mora-Bermúdez, F., Badsha, F., Kanton, S., Gray Camp, J., Vernot, B., Köhler, K., Voigt, B., Okita, K., Maricic, T., He, Z. S., Lachmann, R., Pääbo, S., Treutlein, B. and Huttner, W. B. (2016). Differences and similarities between human and chimpanzee neural progenitors during cerebral cortex development. eLife, 5, e18683. DOI: 10.7554/eLife.18683

[56] Neugebauer, R., Hoek, H. W. and Susser, E. (1999). "Prenatal Exposure to Wartime Famine and Development of Antisocial Personality Disorder in Early Adulthood," JAMA 282, 455–462.

[57] Nimchimsky, E. A., Gilissen, E, Allman, J. M., Perl, D. P., Erwin, J. M., Hof, P. (1999). Neuronal Morphologic Type Unique to Humans and Great Apes. Proc Natl Acad Sci USA 96: 5268–5273.

[58] O'Leary, L., Hughes-McCormack, L., Dunn, K. and Cooper, S. A. (2018) Early Death and Causes of Death of People with Down Syndrome: A Systematic Review. J Appl Res Intellect Disabil, 31(5): 687-708. doi: 10.1111/jar.12446.

[59] Pantoja Zarza, L. and Diez Morrondo, C. (2014). Skeletal Deformities in Mucolipidosis III. Reumatol Clin. Vol. 10, 340-341.

[60] Passingham, R. (2009). How Good is the Macaque Monkey Model of the Human Brain? Current Opinion in Neurobiology, 19: 1-6.

[61] Plomin, R. and Deary, I. J. (2014). Genetics and Intelligence Differences: Five Special Findings. Molecular Psychiatry, 20, 98–108 (2015) http://dx.doi.org/10.1038/mp.2014.105 (Advance online pub).

[62] Plomin. R. and Von Stumm, S. (2018). The new genetics of intelligence. Nature Reviews Genetics, Vol. 19, 148–159.

[63] Premack, D. (2007). Human and Animal Cognition: Continuity and Discontinuity. PNAS (Proceedings of the National Academy of Sciences of the USA), Vol. 104, No. 35, 13861-13867.

[64] Preuss, T. M. and Coleman, G. Q. (2002). Cereb Cortex, 12: 671–691

[65] Prud'homme-Généreux, A. (2011). The Evolution of Human Skin Color. The National Center for Case Study Teaching in Science, University at Buffalo, State University of New York.

[66] Pruyn, M. L., Gartner, B. L. and Harmon, M. E. (2002). Respiratory Potential in Sapwood of Old versus Young Ponderosa Pine Trees in the Pacific Northwest. Tree Physiology, 22, 105-116.

[67] Rooij, W. H. Sr., Yonker, J. E., Painter, R. C. and Roseboom, T J. (2010). "Prenatal Under Nutrition and Cognitive Function in Late Adulthood," PNAS, Proceedings of the National Academy of Sciences USA 107 (2010), 16881-16886

[68] Roth, G. and Dicke, U. (2005) Evolution of the Brain and Intelligence. Trends in Cognitive Sciences, Vol. 9, No. 5, 250-257

[69] Roth, G. (2012) Is the Human Brain Unique? In: A. Fasolo (ed.) The Theory of Evolution and Its Impact, pp. 175-187. Springer-Verlag Italia 2012 DOI 10.1007/978-88-470-1974-4_11

[70] Ruff, C. C., Knauff, M., Fangmeier, T. and Spreer, J. (2003) Reasoning and Working Memory: Common and Distinct Neuronal Processes. Neuropsychologia, 41(9):1241-53 DOI: 10.1016/S0028-3932(03)00016-2

[71] Rumbaugh, D. M., Savage-Rumbaugh, E.S. and Taglialatela, J.P. (2009) Language: Nonhuman Animals. In: L. R. Squire (ed.) Encyclopedia of Neuroscience, Pages 367-371. Elsevier: Academic Press https://doi.org/10.1016/B978-008045046-9.01868-4.

[72] Rumbaugh, D. M. and Savage-Rumbaugh, E. S. (2001) Language and Animal Competencies. In: N. J. Smelser and P. B. Baltes (eds.) International Encyclopedia of the Social & Behavioral Sciences, pp. 8281-8285. Elsevier: Pergamon https://doi.org/10.1016/B0-08-043076-7/01628-4.

[73] Ryskamp, Jessica L. (2010) Anencephaly: Concepts of Personhood, Ethical Questions, and Nursing Care. Senior Honors Theses. 199, Liberty University https://digitalcommons.liberty.edu/honors/199

[74] Savage, Jeanne E., Philip R. Jansen, Sven Stringer et al. (2018). Genome-wide association meta-analysis in 269,867 individuals identifies new genetic and functional links to intelligence. Nature Genetics, Vol. 50, 912–919

[75] Scholtens, L. H., Schmidt, R., de Reus, M. A. and van den Heuvel, M. P. (2014) Linking Macroscale Graph Analytical Organization to Microscale Neuroarchitectonics in the Macaque Connectome. Journal of Neuroscience, 34:12192–12205. https://doi.org/10.1523/JNEUROSCI.0752-14.2014

[76] Schörnig, M., Ju, X., Fast, L., Ebert, S., Weigert, A., Kanton, S., Schaffer, T., Kasri, N. N., Treutlein, B., Peter, B. M., Hevers, W. And Taverna, E. (2021) Comparison of Induced Neurons Reveals Slower Structural and Functional Maturation in Humans than in Apes. eLife, 10:e59323 DOI: 10.7554/eLife.59323

[77] Schusterman, R. J. and Gisiner, R. (1988) Artificial Language Comprehension in Dolphins and Sea Lions: The Essential Cognitive Skills. The Psychological Record, 38, 311–348. https://doi.org/10.1007/BF03395027

[78] Seong, E., Yuan, L. & Arikkath, J. (2015) Cadherins and Catenins in Dendrite and Synapse Morphogenesis. Cell Adhesion & Migration, 9(3), 202-213, DOI: 10.4161/19336918.2014.994919

[79] Shakeshaft, N. G., Trzaskowski, M., McMillan, A., Krapohl, E., Simpson, M. A., Reichenberg, A., Cederlöf, M., Larsson, H., Lichtenstein, P. and Plomin, R. (2015). Thinking Positively: The Genetics of High Intelligence. Intelligence, 48, 123-132.

[80] Skjaerven, R., Wilcox, A. J. and Lie, R. T. (1999) A Population-based Study of Survival and Childbearing among Female Subjects with Birth Defects and the Risk of Recurrence in Their Children. N Engl J Med, 340(14): 1057-62. doi: 10.1056/NEJM199904083401401

[81] Smrt, R. D. and Zhao, X. Y. (2010) Epigenetic Regulation of Neuronal Dendrite and Dendritic Spine Development. Front Biol (Beijing), 5(4):304-323. doi: 10.1007/s11515-010-0650-0.

[82] Teffer, K. and Semendeferi, K. (2012) Human Prefrontal Cortex: Evolution, Development, and Pathology. In: M. A. Hofman and D. Falk (Eds.) Progress in Brain Research, Vol. 195, 191-218, ISSN: 0079-6123 DOI: 10.1016/B978-0-444-53860-4.00009-X

[83] Testa-Silva, G., Verhoog, M. B., Linaro, D., de Kock, C. P., Baayen, J. C., Meredith, R. M., De Zeeuw, C. I., Giugliano, M. and Mansvelder, H. D. (2014) High Bandwidth Synaptic Communication and Frequency Tracking in Human Neocortex. PLoS Biology 12:e1002007. https://doi.org/10.1371/journal.pbio.1002007

[84] The Gorilla Foundation (2018). Project Koko. http://www.koko.org/project-koko

[85] Titz, C. and Karbach, J. (2014) Working Memory and Executive Functions: Effects of Training on Academic Achievement. Psychological Research, 78, 852–868. https://doi.org/10.1007/s00426-013-0537-1

[86] Trzaskowski, M., Harlaar, N., Arden, R., Krapohl, E., Rimfeld, K., McMillan, A., Dale, P. S. And Plomin, R. (2014). Genetic influence on family socioeconomic status and children's intelligence. Intelligence, Vol. 42, 83-88.

[87] van den Heuvel, M. P., Scholtens, L. H., Feldman Barrett, L., Hilgetag, C. C. and de Reus, M. A. (2015) Bridging Cytoarchitectonics and Connectomics in Human Cerebral Cortex. Journal of Neuroscience, 35: 13943–13948. https://doi.org/10.1523/JNEUROSCI.2630-15.2015

[88] Wei, J. (2012). Great Inventions that Changed the World. New Jersey and Canada: John Wiley & Sons, Inc.

[89] White, T. D., Asfaw, B., Beyene, Y., Haile-Selassie, Y., Lovejoy, C. O., Suwa, G. and WoldeGabriel, G. (2009). Ardipithecus ramidus and the Paleobiology of Early Hominids. Science, Vol. 326, No. 5949, pp. 64-86.

[90] Whiten, A. (2013). Monkeys, Apes, Imitation and Mirror Neurons. Cortex, 49, 2941-2943.

[91] Wisniewski, K. E. (1990) Down Syndrome Children Often Have Brain with Maturation Delay, Retardation of Growth, and Cortical Dysgenesis. Am J Med Genet Suppl, 7:274-81. doi: 10.1002/ajmg.1320370755.

[92] Yang, T. M. and Shadlen, M. N. (2007) Probabilistic Reasoning by Neurons. Nature, 447, 1075-1080 https://doi.org/10.1038/nature05852

[93] Yin, Mei (2020) How Do the Black Hole and the Universe Form? International Journal of Psychosocial Rehabilitation, Vol. 24, No. 7, 243-247, ISSN: 1475-7192

AUTHOR INDEX