

David C. Wyld
Jan Zizka (Eds)

Computer Science & Information Technology

Fifth International conference on Computer Science and Information
Technology (CCSIT - 2015)
Sydney, Australia, February 21 ~ 22 - 2015



AIRCC

Volume Editors

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

Jan Zizka,
Mendel University in Brno, Czech Republic
E-mail: zizka.jan@gmail.com

ISSN: 2231 - 5403
ISBN: 978-1-921987-32-8
DOI : 10.5121/csit.2015.50401 - 10.5121/csit.2015.50413

This work is subject to copyright. All rights are reserved, whether whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the International Copyright Law and permission for use must always be obtained from Academy & Industry Research Collaboration Center. Violations are liable to prosecution under the International Copyright Law.

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

Preface

Fifth International conference on Computer Science and Information Technology (CCSIT-2015) was held in Sydney, Australia, during February 21~22, 2015. Third International Conference on Signal, Image Processing and Pattern Recognition (SIPP-2015), Third International Conference on Artificial Intelligence, Soft Computing (AISC-2015) and Fourth International Conference on Natural Language Processing (NLP-2015) were collocated with the CCSIT-2015. 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 CCSIT-2015, SIPP-2015, AISC-2015, NLP-2015 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. All these efforts undertaken by the Organizing and Technical Committees led to an exciting, rich and a high quality technical conference program, which featured high-impact presentations for all attendees to enjoy, appreciate and expand their expertise in the latest developments in computer network and communications research.

In closing, CCSIT-2015, SIPP-2015, AISC-2015, NLP-2015 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 CCSIT-2015, SIPP-2015, AISC-2015, NLP-2015.

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
Jan Zizka

Organization

General Chair

Natarajan Meghanathan
Dhinaharan Nagamalai

Jackson State University, USA
Wireilla Net Solutions PTY LTD, Australia

Program Committee Members

Abri Nee. Badaoui Hadjira
Adamu Murtala Zungeru
Ahmad Hatam
Ahmed Y. Nada
Aiguo Song
AKHIL garg
Alaa S. Abuiteiwi
Ali Azimi
Ali Poorebrahimi
Amani K Samha
Amol D Mali
Ashraf A. Shahin
Atheer Yousif Oudah
Aws Zuheer Yonis
Bai Li
Barbaros Preveze
Cheng fang
Chennai Sali
Chin-Chih Chang
Christian Esposito
Dac-Nhuong Le
Danda B. Rawat
Daniel Mihalyi
Efren Gorrostieta
Fatih Korkmaz
Fei J
George Totkov
Haitao Xu
Hao Che
Hicham Behja
Hyunsung Kim
Isa Maleki
Ishfaq Ahmad
Islam Atef
Israa SH.Tawfic

University of Tlemecen, Algeria
Federal University Oye, Nigeria
Horozgan University, Iran
Al-Quds University, Palestine
Southeast University, China
Nanyang Technological University, Singapore
The Islamic University of Gaza, Gaza
Ferdowsi University of Mashhad, Iran
Islamic Azad University, Iran
Queensland university of Technology, Australia
University of Wisconsin-Milwaukee, USA
Cairo University, Egypt
Thi-Qar University, Iraq
University of Mosul, Iraq
Woodside Energy Ltd, Australia
Cankaya University, Turkey
Zhejiang University, China
Biskra University, Algeria
Chung Hua University, Taiwan
National Research Council, Italy
Vietnam National University, Vietnam
Georgia Southern University, USA
Technical University of Kosice, Slovakia
Universidad Autónoma de Querétaro, Mexico
Cankiri Karatekin University, Turkey
Hohai university, China
Plovdiv University, Bulgaria
University of Science and Technology, China
The University of Texas, US
University Hassan Ii Casablanca, Morocco
Kyungil University, Korea
Islamic Azad University, Iran
The University of Texas, US
Alexandria university, Egypt
Gaziantep University, Turkey

Ivanov V	University of Craiova, Romania
Jan Saliga	Technical University of Kosice, Slovakia
Jasmine Seng	Edith Cowan University, Australia
Joao Ricardo Silva	Autonomous University of Lisbon, Portugal
Karol Matiaško	Katedra informatiky, Slovakia
Keivan Borna	Kharazmi University, Iran
Keneilwe Zuva	University of Botswana, Botswana
Kwan Hee Han	Gyeongsang national University, South Korea
Li Wern Chew	Intel Archit. Group (IAG), Malaysia
Linda Yang	University of Portsmouth, England
Lisa Gandy	Central Michigan University, US
Lubomir Brancik	Brno University of Technology, Czech Republic
Mahdi Mazinani	Azad University Shahreqods Branch, Iran
Manoj Vasanth Ram	Advanced Micro Devices, USA
Mohammad Masdar	Islamic Azad university,Iran
Mohiy M. Hadhoud	Menoufia University,Egypt
Muhammad Abrar	Massey university, New Zealand
Mustafa Aktas	Karabuk University, Turkey
Nabila Labraoui	University of Tlemcen, Algeria
Najib A. Odhah	IBB university, Yemen
Neetesh Saxena	The State University of New York, USA
Nigel McKelvey	Letterkenny Institute of Technology, Ireland
Peiman Mohammadi	Islamic Azad University,Iran
Pr Smain Femmam	University of Haute Alsace UHA,France
R A Carrasco	Newcastle University, United Kingdom
Rahil Hosseini	Azad University, Iran
Rajkumar Patro	Haramaya University, Ethiopia
Ramayah T	Universiti Sains Malaysia, Malaysia
Reda Mohamed Hamou	Tahar Moulay University of Saida, Algeria
Riaan Stopforth	University of KwaZulu-Natal, South Africa
Saad M. Darwish	Alexandria University, Egypt
Saeed M Agbariah	George Mason University, USA
Saeid Asgari Taghanaki	Azad University, Iran
Saurabh Garg	University of Tasmania, Australia
Seyyed AmirReza Abedini	Islamic Azad University, Iran
Seyyed Reza Khaze	Islamic Azad University, Iran
Shuxiang Xu	University of Tasmania, Australia
Steffen Heber	NC State University, US
Subidh Ali	New York University Abu Dhabi, UAE
Tanweer Alam	Islamic University, Kingdom of Saudia Arabia
V Shandilya	University of Memphis, USA
Vasanth Ram Rajarathinam	Advanced Micro Devices, USA
Vimal Kumar	University of West Florida, USA
Virasit Imtawil	Khon Kaen University, Thailand
VojislavMiltenovic	University in NIS, Sebria
Wai Lok Woo	Newcastle University, United Kingdom
William R Simpson	Institute for Defense Analyses, USA
Yasuko Kawahata	Kyushu University, Japan
Zhe Chen	Dalian University of Technology, China

Technically Sponsored by

Networks & Communications Community (NCC)



Computer Science & Information Technology Community (CSITC)



Digital Signal & Image Processing Community (DSIPC)



Organized By



Academy & Industry Research Collaboration Center (AIRCC)

TABLE OF CONTENTS

Fifth International conference on Computer Science and Information Technology (CCSIT - 2015)

Towards Enhancing Resource Scarce Cloudlet Performance in Mobile Cloud Computing	01 - 11
<i>Md Whaiduzzaman, Abdullah Gani and Anjum Naveed</i>	
A Novel Implementation of Hardware Based Hybrid Embedded RTOS	13 - 26
<i>Qiang Huang, Yongbin Bai, QiRui Huang and XiaoMeng Zhou</i>	
Mobile Application Testing Matrix and Challenges	27 - 40
<i>Bakhtiar M. Amen, Sardasht M. Mahmood and Joan Lu</i>	
An Overview of Fragmentation Design for Distributed XML Databases	41 - 52
<i>Kok-Leong Koong, Su-Cheng Haw and Lay-Ki Soon</i>	

Third International Conference on Signal, Image Processing and Pattern Recognition (SIPP 2015)

A Case Study in Computer Understanding of Printed-Forms	53 - 62
<i>Davood Falahati, Hojat Cheraghi and Kazem Ghalamchi</i>	
Unsupervised Region of Interest Detection Using Fast and Surf	63 - 72
<i>Abass A. Olaode, Golshah Naghdy and Catherine A. Todd</i>	
A Decision Tree Based Pedometer and its Implementation on the Android Platform	73 - 83
<i>Juanying Lin, Leanne Chan and Hong Yan</i>	

Third International Conference on Artificial Intelligence, Soft Computing (AISC-2015)

E-Learning Scenarios Using Intelligent Multiagent Systems	85 - 88
<i>Ali M. Aseere</i>	

Fourth International Conference on Natural Language Processing (NLP - 2015)

- Improvement WSD Dictionary Using Annotated Corpus and Testing it
with Simplified Lesk Algorithm.....** 89 - 97
Ahmed H. Aliwy and Ayad R. Abbas
- Myanmar Web Pages Crawler.....** 99 - 110
Su Mon Khine and Yadana Thein
- Improving a Japanese-Spanish Machine Translation System Using
Wikipedia Medical Articles.....** 111 - 116
Jessica C. Ramírez, Yuji Matsumoto and Darwin Muñoz
- Recent Approaches to Arabic Dialogue Acts Classifications.....** 117 - 129
AbdelRahim A. Elmadany, Sherif M. Abdou and Mervat Gheith
- Quick Pad Tagger : An Efficient Graphical User Interface for Building
Annotated Corpora with Multiple Annotation Layers.....** 131 - 143
Marc Schreiber, Kai Barkschat, Bodo Kraft and Albert Zündorf

TOWARDS ENHANCING RESOURCE SCARCE CLOUDLET PERFORMANCE IN MOBILE CLOUD COMPUTING

Md Whaiduzzaman¹, Abdullah Gani² and Anjum Naveed³

^{1,2,3}Center for Mobile Cloud Computing Research (C4MCCR)
Faculty of Computer science & Information Technology,
University of Malaya, Kuala Lumpur, Malaysia

¹wzaman@ieee.org, ²abdullah@um.edu.my, ³anjumnaveed@ieee.org

ABSTRACT

In recent years, mobile devices such as smart phones, tablets empowered with tremendous technological advancements. Augmenting the computing capability to the distant cloud help us to envision a new computing era named as mobile cloud computing (MCC). However, distant cloud has several limitations such as communication delay and bandwidth which brings the idea of proximate cloud of cloudlet. Cloudlet has distinct advantages and is free from several limitations of distant cloud. However, limited resources of cloudlet negatively impact the cloudlet performance with the increasing number of substantial users. Hence, cloudlet is a viable solution to augment the mobile device task to the nearest small scale cloud known as cloudlet. However, this cloudlet resource is finite which in some point appear as resource scarcity problem. In this paper, we analyse the cloudlet resource scarcity problem on overall performance in the cloudlet for mobile cloud computing. In addition, for empirical analysis, we make some definitions, assumptions and research boundaries. Moreover, we experimentally examine the finite resource impact on cloudlet overall performance. By, empirical analysis, we explicitly establish the research gap and present cloudlet finite resource problem in mobile cloud computing. In this paper, we propose a Performance Enhancement Framework of Cloudlet (PEFC) which enhances the finite resource cloudlet performance. Our aim is to increase the cloudlet performance with this limited cloudlet resource and make the better user experience for the cloudlet user in mobile cloud computing.

KEYWORDS

Mobile Cloud computing, Cloudlet, Resource Scarcity, Performance Enhancement.

1. INTRODUCTION

Cloudlet is a small cloud located in close vicinity to the mobile users connected through wireless communication. Cloudlet is installed on discoverable, localized, stateless servers running one or more virtual machines (VMs) on which mobile devices can augment resource intensive applications offloaded for computational resources [1]. It is a set of relatively resourceful computers that is well-connected to the Internet and is available for use by nearby mobile devices [2]. Satyanarayanan, et al. [1] first introduced the cloudlet concept and mentioned it as a “data center in a box”. It is self-managing, requiring little power, Internet connectivity, and access control for setup. This simplicity of management make it feasible to use as a model of computing

resources and to deploy on a business premises such as a coffee shop or a doctor's office. Internally, a cloudlet resembles a cluster of multicore computers, with internal connectivity and a high-bandwidth wireless LAN for external access and having the virtualization capability of cloud computing. Hence, a cloudlet can be viewed as a surrogate or proxy of the real cloud, located as the middle tier of a three-tier hierarchy: mobile device, cloudlet, and cloud [3-5].

Mobile cloud computing liberates mobile devices from resource constraints by enabling resource-intensive applications to leverage cloud computing. Researchers named it as a cyber-foraging which can be realized using distant remote cloud. However, due to WAN latency, jitter, congestion, slow data transfer resulting increased power consumption and cost for user side [6, 7].

Hence, to alleviate these problems, clouds are being taken closer to the user by cloudlet concept. The benefits of utilizing cloudlet are the speed of service accessibility, the support of mobility, the enhanced application performance, the elongated battery life, and the reduced roaming data cost[8] . Cloudlet has a major important role in Mobile cloud computing for its several distinguished advantages and features. Recently, researchers have found cloudlet as a viable solution for mobile cloud computing. [3, 9].Cloudlet and distant cloud have the same functionality with some differences. Cloudlet performs the tasks that are offloaded to the cloudlet using different offloading mechanism. Cloudlet has relatively higher resources compared to the mobile devices and effectively, task completion time is lesser on cloudlet, compared to the mobile device. However, unlike cloud where the user OS instance is stored along with modifications permanently, in case of cloudlet, the basic OS instances are available while user snapshots of the customized OS instances cannot be saved because of limited storage and lesser probability of reuse [10, 11]. Cloudlet can be situated in common public area, business center, airport, coffee shops, shopping mall which facilitated the offloading facility to the mobile user by connecting the mobile devices as a thin client to the cloudlet [12, 13].

There are several methods and offloading techniques introduced for application migration from mobile device to the computational clouds. One is VM migration, in which an already executing VM is suspended; its processor, disk, and memory state are transferred; and finally VM execution is resumed at the destination from the exact point of suspension. For application migration, Satyanarayanan, et al. [4] introduced a dynamic VM synthesis that enable mobile devices to deliver a small VM overlay to the cloudlet infrastructure that already possesses the base VM from which this overlay was derived. The infrastructure applies the overlay to the base to derive the launch VM, which starts the suspended execution of the suspension at the exact precise point [14, 15].

To realizing the Cyber foraging using a cloudlet, the VM-based cloudlet concept has recently evolved to component-based cloudlets consisting of a group of computing nodes (both fixed and mobile) that are sharing resources with one another. Software components on the mobile device can then be redeployed at runtime to other nodes in the cloudlet according to some optimization criteria, such as the execution time, energy consumption and throughput[7, 10] . These applications are involving multiple users interacting with each other in a real-time fashion. The resource-sharing concept of component-based cloudlets opens a promising research area for collaborative applications which not only sharing computing resources but also share user data such as processing results and context information [9, 15]. Cloudlet has a major important role in mobile cloud computing for its several distinguished features as follows [11].

The mobile users get instant direct access to the cloudlet, due to the close proximity of the user and the cloudlet which eliminates several drawbacks introduced by the communication latency, jitter, and slow data transfer of cellular network. The conventional benefit of offloading the computational resource intensive tasks into cloud is still preserved in cloudlet in which the mobile device can get rid of resource starvation. Finally, since cloudlet is the near vicinity of the mobile user, it can save money by avoiding expensive data charging in roaming situation [8, 16]. Hence,

cloudlet can be situated in common public area, business center, airport, coffee shops, shopping mall which facilitated the offloading facility to the mobile user by connecting the mobile devices as a thin client to the cloudlet[13].Section 2 explains the problem analysis of resource scarce cloudlet, Section 3 describes the PEFC framework with different components, and Section 4 presents the significance of the framework and finally Section 5 draws conclusive remarks with our future research direction.

2. PROBLEM ANALYSIS

Cloudlet has finite resource and this is an intrinsic problem of cloudlet which has a negative effect of its overall performance. In this study, we aim to highlight the resource scarcity problem and establish the impact of the resource scarcity on cloudlet performance by empirical analysis.

2.1 System parameters

We first define the resources, tasks and workload for cloudlet. Subsequently we define the performance parameters. Our defined several definitions for analysis are as follows:

Definition 1 (Resource): A resource represents an available unit which is required for executing a task. We can denote r as a resource and R can be denoted as a set of available resource.

Definition 2 (Cloudlet resources): For cloudlet, three types of resources are available such as CPU, memory and storage. The fundamental operation of most CPUs, regardless of the physical form they take, is to execute a sequence of stored instructions.

Definition 3 (Task): A task can be a logical unit of work which is executed by resource.

Definition 4 (User service time): User service time indicate the total time taken by the cloudlet and other related transfers time and network delay to deliver the computation service to the mobile user. User service time indicate the total time, including execution time and wait time, taken by the cloudlet to deliver the computation service to the mobile user.

Definition 5 (Remote application execution time): The total time is taken for execute the application in cloudlet

Definition 6 (Local application execution Time): The time is taken to execute the program in local mobile device.

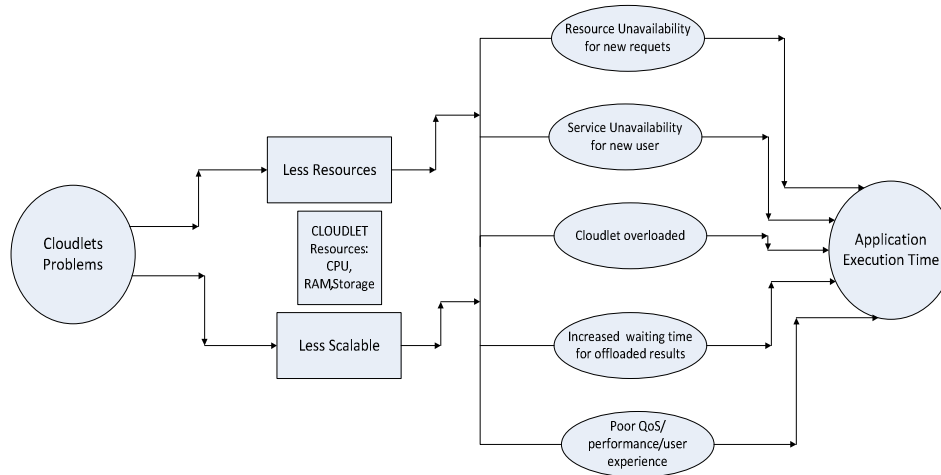
Definition 7 (Large application/program): We purposely make this application as a compute-intensive which comparatively take longer time than our small program.

Definition 8 (Small Application/program): We purposely make this application as a compute-intensive which comparatively take lesser time than our large program.

2.2 Experiment

In this section, we describe experimental model, mobile client and cloud servers specification, communication infrastructure and data design. Here, we analyze the impact and verify the severity of resource scarcity impact on cloudlet. Considering Open stack as a cloudlet service provider, we use mobile devices for a local user for the test bed.

One is constrained of resources of cloudlet, since these resources are free to use, therefore the available resources are not adequate when the number of users or applications request for computational services from the cloudlet. The increasing user load, in some point make the cloud resource scarceness and ultimately the resource constraint of cloudlet take more time to application execution, which affect the user experience with long application execution delay. It also creates the problem of on demand resource availability by the new user request since resources are provisioned already and lack of available resources for the new requests. These effects severely hindered the main purpose of using cloudlet.



Cloudlet Problem Analysis with possible effects in cloud services

Figure 1. Cloudlet problem Analysis with possible effects in application execution time

If we consider the aforementioned problems in cloudlet which are fewer resources and less scalable, ultimately it affects the application execution time. Among the resources in cloudlets, for our experiment we consider the processor speed, and how effective it will be to execute the application in cloudlet. In application execution, we consider two aspects: One is in remote application execution time which should be in cloudlet, and other one is in local application execution time which we consider the time taken by the local execution by the mobile device. For our experiment, we define and design a large program and a small program to differentiate it by two types of execution time. It is obvious that large program computation time is more than the small program computation time. We made it purposely to done our experiment. Several important terms for this experiment:

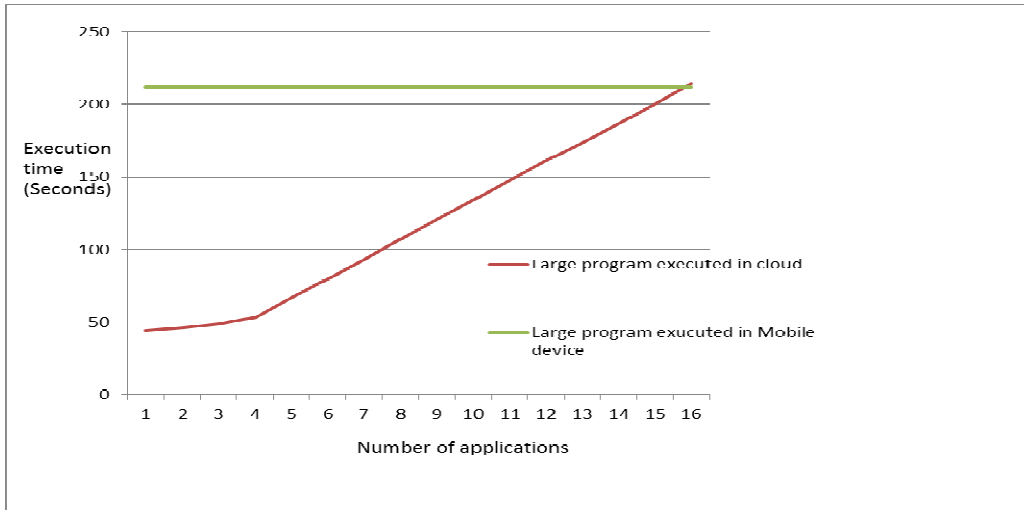


Figure 2. Comparison of small program execution time in cloudlet and mobile device

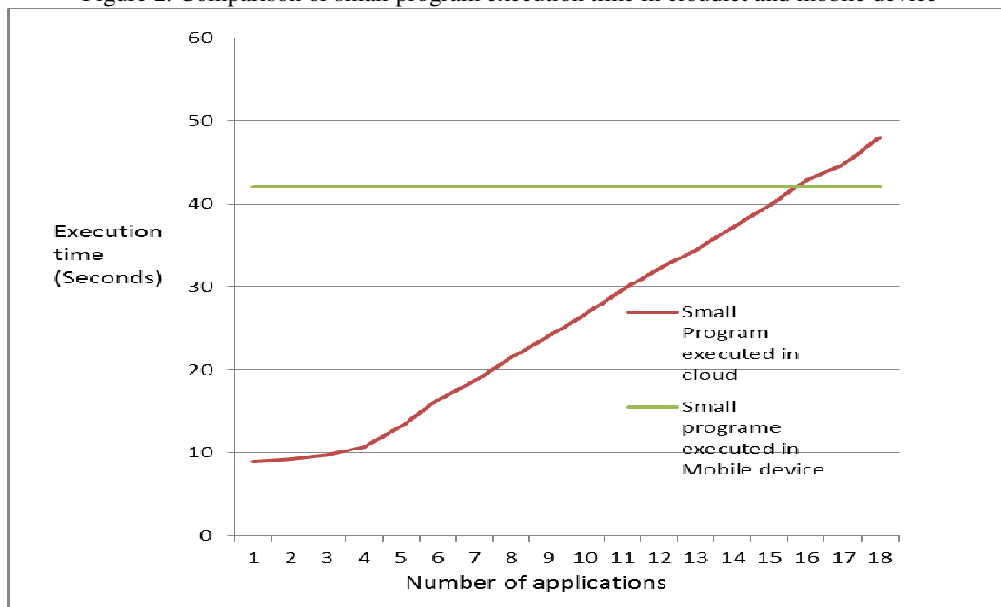


Figure 3. Comparison of Large Program execution time in cloudlet and mobile device

We have done this experiment in Open stack Havana Cloud running in Ubuntu12.04 Linux server and Samsung Galaxy S2 is used as a mobile device in the lab environment real test bed.

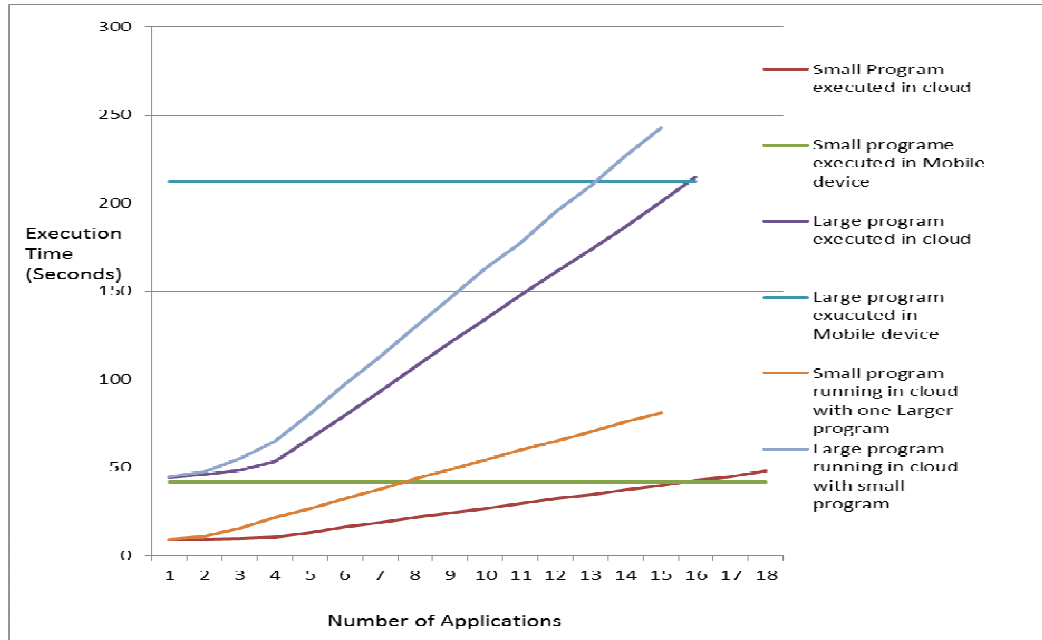


Figure 4. Comparison of different types program execution time in cloudlet and mobile device

From Experiments and Figure 2,3,and 4 , we can confirm that all of the cases if 16 users using the cloudlet, then 17th number user have no benefit from the cloudlet in terms of program execution time in cloudlet. Because, in this case, the mobile device can run faster, if execute in locally.

3. PERFORMANCE ENHANCEMENT FRAMEWORK FOR CLOUDLET

This section reports a Performance Enhancement Framework for Cloudlet (PEFC) for MCC. The objective is to explain and address the issues of resource scarcity problems which hindered the cloudlet performance. This section explains the architecture and operating procedure and performance of the proposed framework.

This framework basically consists with two major building blocks. One is the user side; we assume it a mobile device. In this study, we assume all the devices are same type in operation perspective. In the cloud side, we call it cloudlet, a small scale cloud which is built and operated in open source cloud software named as open stack. We propose a novel Performance Enhancement Framework for Cloudlet (PEFC) for MCC. PEFC address the issues of resource scariness in cloudlet by offloading and shifting some of its workload or process to the nearby mobile device using the Wi-Fi communication and finally sent the processed output to the specific user. To design and development of the framework, we consider several important offloading aspects. For decision making, we employed several decision making algorithms and we consider the mobility pattern for the framework. Two tiers architecture of our framework basically builds up with the mobile user which we consider as an end user. This user can be identified as a smartphone, tables and even the laptop user.

After user sending the tasks to the cloudlet, then the second part is executed by the cloudlet for task completion. This step involves several serious consideration and efficient consecutive process for the whole task completion cycle. First of all, the framework check the usability of the cloudlet whether it is practically beneficial for the user to offload the task. If yes then it goes

through several sections instructed and directed by our algorithms. We explain the each steps individually the main job and their internal execution to complete the whole task and send back the result to the user. We present the framework components and building blocks in Fig 5. The following section explains the components of PEFC framework:

3.1 Task Handler

Task handler is situated in Cloudlet side which first receives the task request from the mobile device. Task handler mainly analyzes the task and makes a weight for the computational benefit from the cloud. This task handler initially assesses the whole task and the priority and importance. For the further analysis then it send to the aggregator and profile section.

3.2 Aggregator

The aggregator mainly aggregates all the components of the task and reorganizes and rearranges it according to the sequence of the application. It basically incorporates the different components of the task and reshuffles and reorganizes the random part to make it in an orderly meaningful manner.

3.3 Profiler

Profiler mainly responsible for analyze the applications and its different components. It describes the need of demand for computation units need to complete the task execution. In addition it can make the all components of profile to identify and reorder the task and calculate the whole necessary resources needed to finish the work.

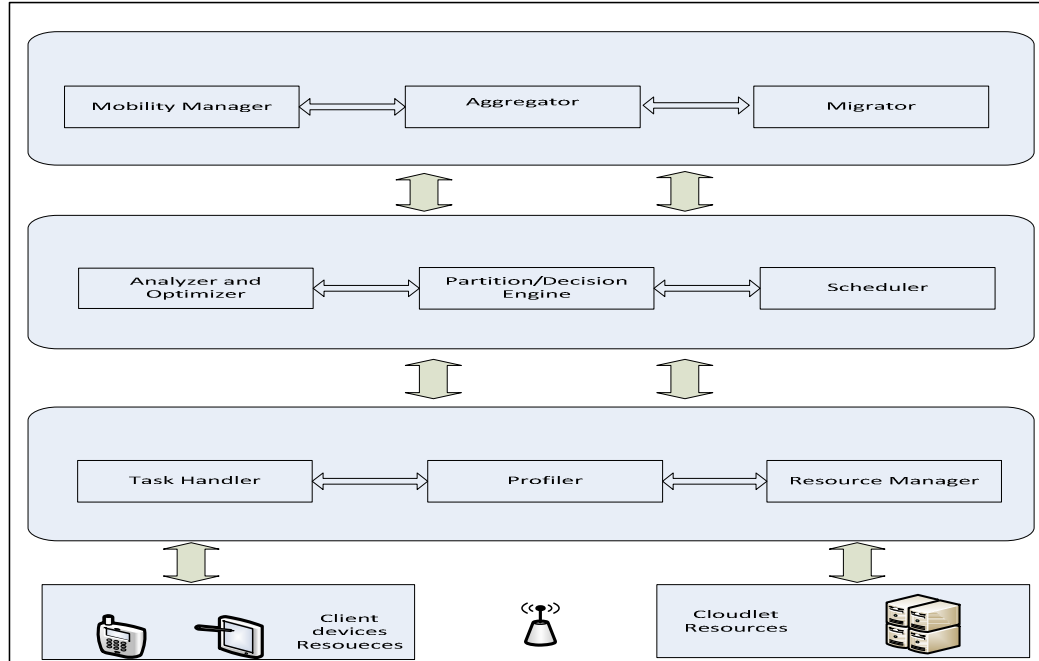


Figure 5. Main building blocks of Performance Enhancement Framework for Cloudlet

3.4 Analyzer and Optimizer

After profiling, tasks are handed to the analyzer and optimizer section for further analysis based on the profiler data. By using optimization techniques, analyzer and optimizer decide to further processing of the request tasks which should be optimally done by the available resources for the cloudlet and mobile devices.

3.5 Scheduler

Scheduler makes the scheduling for complete the task. It assigns the priority which devices the task will reschedule and overall available resources for task scheduling. When task finish then it update accordingly and make the resource free for the other sequential important task for completion.

3.6 Partition/Decision Engine

The decision engine makes the decision of application execution. It could be the in to cloudlet or to the mobile devices other than the sending devices. This section makes decision after analysing the resources associated with the mobile devices resources. We employ several decision making algorithm in this stage to get the optimized and enhanced performance by make the decision that should be the optimized one.

3.7 Mobility Manager

Mobility manager hold the status of all the mobile devices connected to the cloudlet in the proximity of cloudlet. It can be done by storing all the Wi-Fi signal strength registered with the Wi-Fi zone and their signal strength confirm us the proximity of the all the nearest devices. Among them, from the client profiler, we can have the resources latest update and the information who are waiting for the cloudlet services by the receiving the results from the cloudlet. Mobility is an important aspect which we also consider in our framework. From the mobility pattern analysis, when decision engine make decision which mobile device, it is going to offloaded task to the client devices can help the predicted mobile device.

3.8 Resource Manager

Resources indexing are keeping as a database to the cloudlet. All the available and presently used resources are keep track by the resource manager. Therefore resource manager is responsible for keep all the updated data and information in the cloudlet which is very important for the decision engine to estimate the available resources from the cloudlet. Hence, every new device joining in the network should register their resources by the profiler to the cloudlet and again, when any device leave the network, instantly the resource manager remove the device and it's available resources from the database.

3.9 Migrator

Cloudlet use Migrator to transfer the portion of data or code segment or process to migrate to the surrounded mobile devices. Obviously, decision engine using our algorithms make decision to choose the best case mobile device. After that for sending the code, migrator is responsible of sending to the mobile device. Migrator receives the result from the mobile device to process it again by sending to the aggregator and other further steps.

3.10 Client Devices Resources

The resources which are embedded with the client mobile devices are mentioned or marked here as user device resources. It could be CPU, memory or even storage. Several cases it could be the installed software that cannot be processed by the cloudlet or cloudlet has not installed with the software facility. We can have huge sensors now a day that could be great resources as well such as GPS, camera, thermometer, location apps, and embedded other latest sensors.

3.11 Cloudlet Resources

Resources which are belongs to cloudlet are normally considered as CPU, memory, and hard disks. In this experiment and our frame work, mainly we consider the compute-intensive applications, hence we identified the CPU is the main resources. As, we proof in chapter 3 that certain cases, this resources are not adequate and we call it some point it as a resource scarcity which degrade the overall cloudlet performance negatively. We aim to consider the problem and propose the solution to shift some application or process from the cloudlet to the nearest mobile devices that they can act as collaborative resources for the cloudlet. Since when the loads are reduced from the cloudlet, ultimately the performance has been increased and the results we get from the nearest mobile devices as working as a resource node or hub eventually make the whole process faster and make the cloudlet to increase the performance and can get rid of resource scarcity.

4. ADVANTAGE AND SIGNIFICANT

Our proposed framework has several important and significant features and charters tics which we outline below:

4.1 Enhanced Resources

One of the most significant features of this framework is it eventually increase the resources of cloudlet. In fact, practically, it does not increase the physical resources of the cloudlet .However; it does ultimately the same thing, if we increase the physical resources to the cloudlet. Especially when it dealt with the shifting tasks from cloudlet to the mobile device, in reality, it reduces the load from the cloudlet resources which free the cloudlet resources.

4.2 Nearest Proximity

The resource rich mobile devices are located to the close proximity of the cloudlet which brings the mobile devices additional or extra resources nearby to the cloudlet. Moreover, this nearness makes the framework more stable because it is less affected by the band width or network latency which is common for distant cloud.

4.3 No Cost

In the cost perspective, no costs are involved with this model and we know the cloudlet service is free. Hence this model is pretty straight forward no payment module involve with in it. This also help easier to implement the model and encourage the near vicinity of the cloudlet user to use their resources as free but still the mobile user who give his/her resources for the cloudlet ensure the advantage of time or energy benefit. This establishes the strong feasibility of free service and in this model the cloudlet and the mobile user both are getting mutual benefit with this model.

4.4 Performance Enhancement

Most important aspect of this framework, we get extra resources without any payment which ultimately increase the overall performance. It works both ways, at the same time it ensure the cloudlet performance increase and well as the mobile user performance by helping the mobile user to perform his tasks in timely manner with time and energy benefit.

4.5 Network delay

Our proposed framework considers the resources from the cloudlet and some are from the nearest mobile devices collaboratively. Therefore, this framework model do not need cellular or mobile network for communication between client and cloud. Moreover, it works using the wireless network with Wi-Fi connection, hence notably important that this proposed framework is free from communication latency and network delay.

4.6 Adaptive ness

To make the experiment simple and to avoid complexity, we make the framework simple and use one type of mobile devices. However, this model is suitable and appropriate for other devices with different operating systems. In short, it is feasible and supports the heterogeneity devices and operating system. Therefore, this framework has a strong adaptive ness with different platform and software which make it as strong business feasibility in the marketplace for its adaptive ness.

5. CONCLUSIONS

In this paper, we highlight the cloudlet resource scarcity impact on overall performance in the cloudlet for mobile cloud computing. First we explain several basic concepts of cloudlet, then, for empirical analysis, we make several assumptions and research boundaries. In addition, we experimentally examine the effects and impacts of finite resource on cloudlet overall performance. We establish the research gap and present cloudlet finite resource problem. In this study, we propose a framework, PEFC, to enhance the cloudlet performance. Our proposed framework, mitigate the resources scariness of cloudlet by shifting the load to the nearest idle mobile device to enhance the cloudlet performance. We describe the frameworks several components and highlights it's the important and distinct features. Finally, our framework shows the cloudlet performance enhancement and in our future work, we will implement the framework and empirically investigate the performance enhancement of the cloudlet in mobile cloud computing.

ACKNOWLEDGMENTS

This work is fully funded by the Malaysian Ministry of Education under the High Impact Research Grant of University of Malaya UM.C/625/1/HIR/MOE/FCSIT/03.

REFERENCES

- [1] M.Satyanarayanan, P.Bahl, R.Caceres, and N.Davies, "The case for vm-based cloudlets in mobile computing," *Pervasive Computing*, IEEE, vol. 8, pp. 14-23, 2009.
- [2] A.Bahtovski and M.Gusev, "Cloudlet Challenges," *Procedia Engineering*, vol. 69, pp.704-711, 2014.
- [3] M.R. Rahimi, "Exploiting an elastic 2-tiered cloud architecture for rich mobile applications," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2012 IEEE International Symposium on a, 2012, pp. 1-2.
- [4] M.Satyanarayanan, G.A.Lewis, E.J.Morris, S.Simanta, J.Boleng, and K.Ha, "The Role of Cloudlets in Hostile Environments," *IEEE Pervasive Computing*, vol. 12, pp. 40-49, 2013.

- [5] M.Whaiduzzaman, M.Sookhak, A.Gani, and R. Buyya, "A survey on vehicular cloud computing," *Journal of Network and Computer Applications*, vol. 40, pp. 325-344, 4// 2014.
- [6] S.Bohez, T.Verbelen, P.Simoens, and B.Dhoedt, "Allocation Algorithms for Autonomous Management of Collaborative Cloudlets."
- [7] E.Ahmed, A.Akhunzada, M.Whaiduzzaman, A.Gani, S.H.Ab Hamid, and R.Buyya, "Network-centric performance analysis of runtime application migration in mobile cloud computing," *Simulation Modelling Practice and Theory*.
- [8] M.Whaiduzzaman, M.N.Haque, M.Rejaul Karim Chowdhury, and A.Gani, "A Study on Strategic Provisioning of Cloud Computing Services," *The Scientific World Journal*, vol. 2014, p. 16, 2014.
- [9] A.Gani, G.M.Nayeem, M.Shiraz, M.Sookhak, M.Whaiduzzaman, and S.Khan, "A review on interworking and mobility techniques for seamless connectivity in mobile cloud computing," *Journal of Network and Computer Applications*, vol. 43, pp. 84-102, 8// 2014.
- [10] H.Qi, M.Shiraz, A.Gani, M. Whaiduzzaman, and S.Khan, "Sierpinski triangle based data center architecture in cloud computing," *The Journal of Supercomputing*, pp. 1-21, 2014/04/26 2014.
- [11] M.K. Nasir and M.Whaiduzzaman, "Use of Cell Phone Density for Intelligent Transportation System (ITS) in Bangladesh," *Jahangirnagar University Journal of Information Technology*, vol. 1, 2012.
- [12] N.Fernando, S.W.Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future Generation Computer Systems*, vol.29, pp. 84-106, 2013.
- [13] M.Shiraz, M.Whaiduzzaman, and A. Gani, "A study on anatomy of smartphone," *J Comput Commun Collab*, vol.1, pp. 24-31, 2013.
- [14] M.Whaiduzzaman, A.Gani, N.B.Anuar, M.Shiraz, M.N.Haque, and I.T.Haque, "Cloud Service Selection using Multi-Criteria Decision Analysis," *The Scientific World Journal*, 2013.
- [15] M.Whaiduzzaman and A. Gani, "Measuring security for cloud service provider: A Third Party approach," in *Electrical Information and Communication Technology (EICT), 2013 International Conference on*, 2014, pp. 1-6.
- [16] M.Whaiduzzaman, A.Gani, N.B.Anuar, M.Shiraz, M.N.Haque, and I.T.Haque, "Cloud Service Selection Using Multicriteria Decision Analysis," *The Scientific World Journal*, vol. 2014, p. 10, 2014.

INTENTIONAL BLANK

A NOVEL IMPLEMENTATION OF HARDWARE BASED HYBRID EMBEDDED RTOS

Qiang Huang¹, Yongbin Bai², QiRui Huang³ and XiaoMeng Zhou⁴

College of Computer Science and Software Engineering,
Shenzhen University, Shenzhen, 518060
jameshq@szu.edu.cn

ABSTRACT

Reliable embedded systems play an increasing role in modern life, especially in modern automotive designs. Many studies have proved that it performs better in many situations. Firstly, reliable embedded systems provide the system reliability improvements. Secondly, reliable embedded systems also can improve the development efficiency and make the development cycle shorter.

However, in the high real-time required occasion, the software implementation of the RTOS can't fully meet requirements. To have better real-time only through the algorithm improvement or just increase the processor speed. On the contrary, operating system based on a hardware implementation can make it more real-time and more reliable. The reason is due to that the hardware circuit is independent of the processor running and do not take up the processing time of the processor. Thereby it can save time to execute other tasks and improve real-time. In this paper, ARM+FPGA will be choose as the IP hardware development platform.

KEYWORDS

Time-triggered/event-triggered, jitter, hardware schedule.

1. INTRODUCTION

Since the 1980s, some international IT organizations have started to research the commercial embedded real-time operating system and specialized real-time operating system. Form that on, there appears many real-time operating systems, like VxWorks, LynxOS, embedded Linux, TRON and uC / OS-II.

In the 80s of the last century in the US, Jaehwan Lee and Vincent John MooneyIII[1] [2]compared the RTOS scheduler implementation from hardware to software, and make a RTOS scheduler accomplished by specialized hardware IP core which will greatly improve the work efficiency of the RTOS.

In Brazil, Mellissa Vetromille and Luciano OST[3] compare and analyze the RTOS scheduler accomplished by the software and hardware, the results was that hardware scheduler has higher reliability.

In Japan, Professor Takumi Nakano[4][5] developed a silicon wafer named STRON-I (Silicon OS) in 1990s. It used VLSI to hardened the operating system TROS to a chip. Therefore, the operating system chips can work in parallel with the microprocessor, which can further ensure the real-time and high reliability of real-time operating systems.

The TTE32-HR2[6] microprocessor made by TTE Systems, which used the cooperative scheduling hardware implementation as the TTE32 kernel peripherals and use it to achieve the task scheduling.

Summary: We can find that the research is mainly concentrated in the local module of the hardening operating system, while little research literature based on the overall hardware design and implementation of real-time operating system. Therefore, we should have deeper research in how to carry out the optimal software partitioning for real-time operating system and accomplish a hardware real-time operating system.

Nowadays, in our country, real-time operating system based software mainly has two different types: one of that is China's independent research and development of real-time operating system, for example: the open source RT-Thread, Delta OS, Hopen OS, CASSPDA developed by Chinese Academy of Sciences, Beijing Software Engineering Center and HBOS of Zhejiang university. Another one is completed by secondary development that based on foreign operating system. This kind of operating system is the exclusive use of the system such as the Chinese Academy of Sciences of the red flag Linux and Shenzhen blue Linux.

At present, the domestic research in literature hardware real-time operating system is nearly zero until some articles have been published recent years. For example, HouMi, from Shanghai Jiaotong University, proposes and designs a real-time task management device based on hardware. WangChuanfu and Zhou Xuehai, from the University of Science and Technology of China's, put forward a method to improve the performance of hardware multithread processor. Zhejiang University professor Chen Tianzhou[7] proposed a CPU FPGA hybrid architecture hardware thread execution mechanism method. Cui Jianhua, Sun Hongsheng and Wang Baojin, from PLA Information Engineering College design a simple hardware real-time operating system and realize the task scheduling, interrupt management and basic function of timer management RTOS with a FPGA development board.

Summary: The present study has focused on the hardware task scheduling and hardware interrupts processing. However, the communication and synchronization between tasks, memory management and implementation in hardware context switch are still a problem to be solved and research.

Our design is a hardware real-time operating system IP kernel which including task scheduling, interrupt processing, communication and synchronization between tasks, and time management. The kernel development use ARM+FPGA as the IP hardware development platform.

2. RELATED WORK

2.1 Hardware platform

There are two ways we can choose to realize hardware platform.

First: ARM+FPGA, the following picture is an overview which we realize on FPGA.

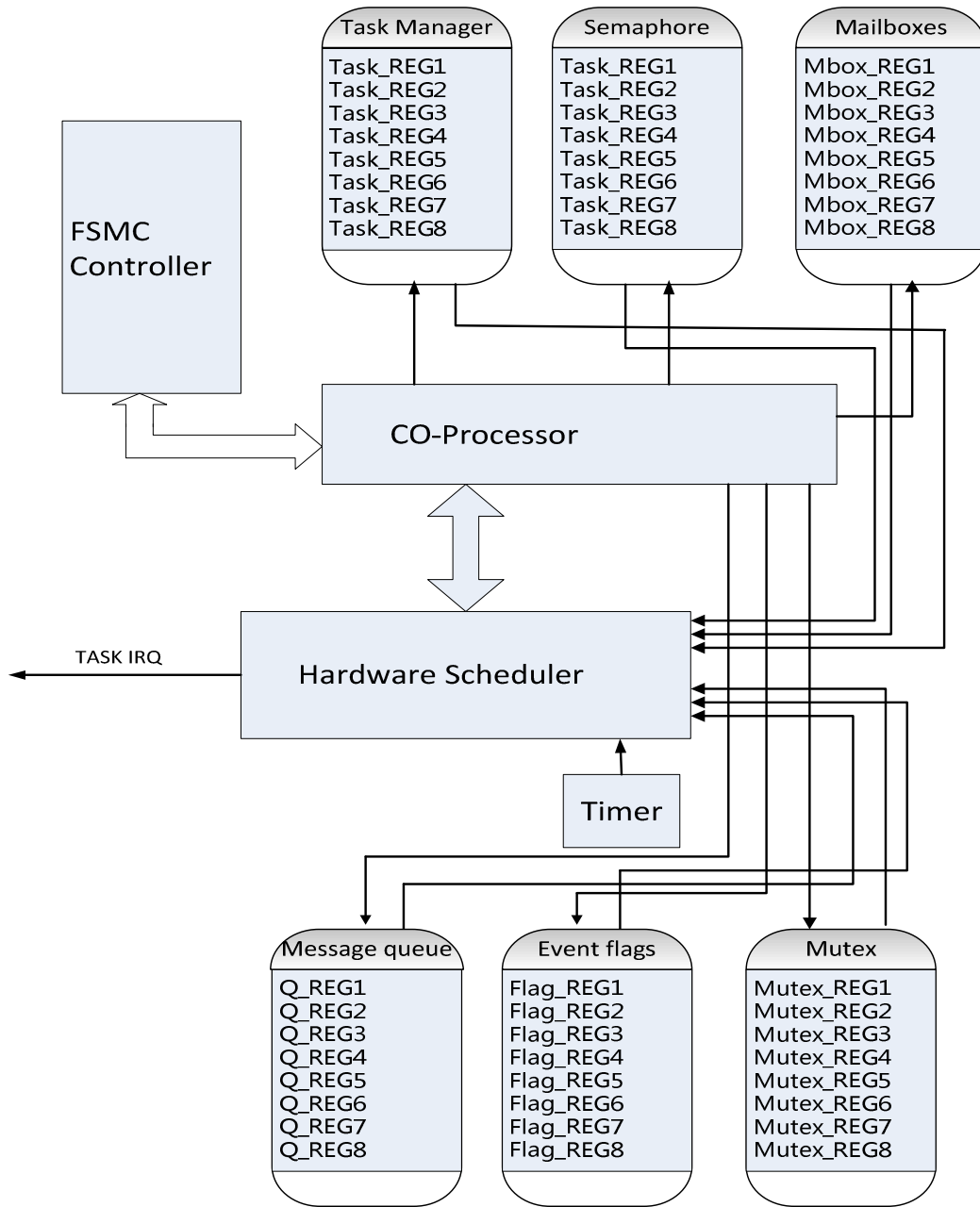


Figure 1. ARM+FPGA

Second: Only using FPGA, just like the following picture. This picture comes from TTE Systems

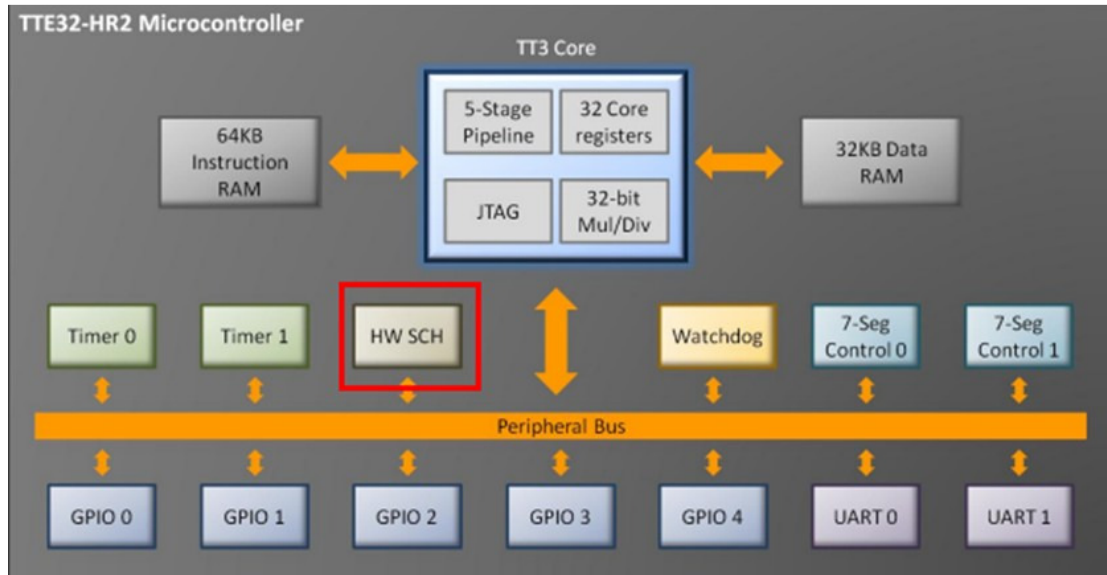


Figure 2. FPGA

Considering that the second method is inconvenient to debug. So we choose the first method as hardware platform.

In order to better test the hardware real-time system. We have made a total of 10 sets of development board.

Features of the platform include:

- (1) MCU uses ST Company's STM32F103VET6
- (2) FPGA uses Altera Company's EP4CE6E22C8
- (3) Supply voltage acquisition circuit
- (4) Supply voltage acquisition circuit
- (5) Ethernet module
- (6) Two Serial port modules
- (7) ZigBee module
- (8) CAN interface
- (9) Segment module
- (10) 4.3-inch LCD module
- (11) Analog signal acquisition circuit
- (12) Buttons and LEDs

2.2 Hardware RTOS features

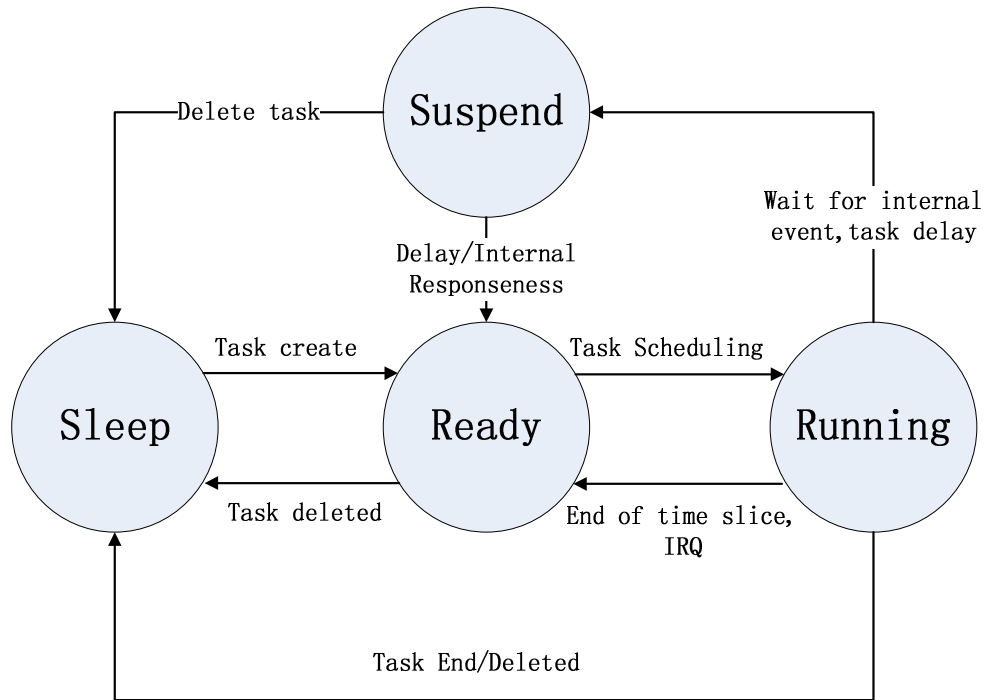


Figure3. State Switching

The Hardware RTOS mainly achieved functions as follows: preemptive scheduling, task management, semaphores, message mailboxes, message queues, mutexes and event flags group.

We have completed all the necessary components for small real-time embedded systems.

- Support the creation of 8 tasks
- Support the creation of 8 semaphores
- Support the creation of 8 message mailboxes
- Support the creation of 8 message queues
- Support the creation of 8 mutexes
- Support the creation of 8 event flags groups

The FPGA can easily extended to support more components and tasks.

2.3 The communication between ARM and FPGA

By using FSMC interface of STM32, we can realize the communication between ARM and FPGA. In order to make FPGA as one part of ARM kernel peripherals, we use Bus Interface instead of SPI or UART.

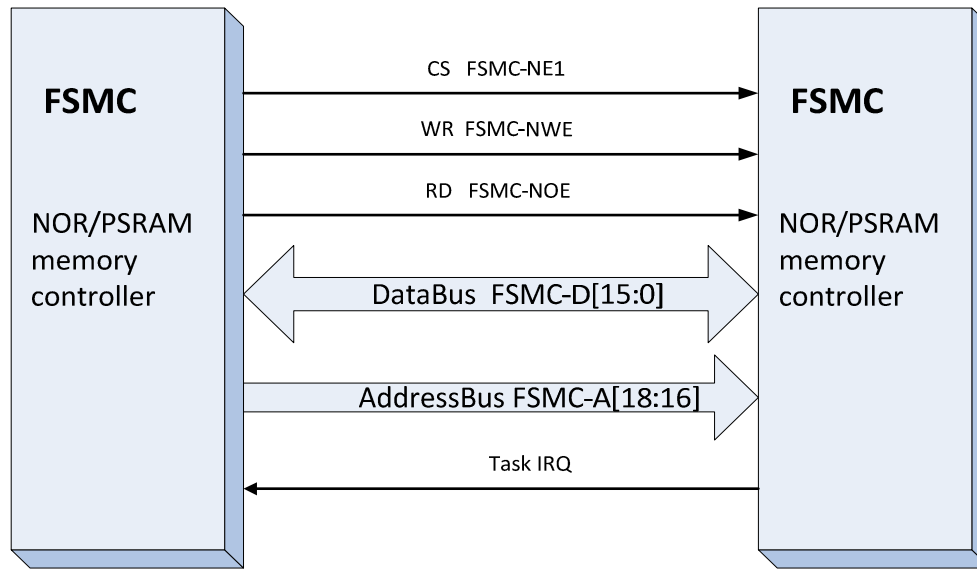


Figure4. ARM and FPGA interface

2.3.1 Writing data to FPGA

From the figure above, there are 16 data wires, 3 address wires. ARM can only access to eight 16-bit data on FPGA. In order to access to more data, We use a method similar to serial. By distinguishing different ID, we realize the access to more data, Each ID can access to eight 16-bit data on FPGA. For example. Writing data to FPGA is mainly used to initialize register.

When register `HW_ID = 0`, ARM can access to eight 16-bit data on FPGA.

```
HW_ID = 0;
HW_DELAY = 111;
HW_PERIOD = 0;
HW_GPT = 144;
HW_AOT = 155;
HW_BACKUP = 166;
HW_OVERRUN = 177;
HW_CONTROL = 1;
```

When register `HW_ID = 1`, ARM can access to eight 16-bit data on FPGA.

```
HW_ID = 1;
HW_DELAY = 211;
HW_PERIOD = 0xffff;
HW_GPT = 244;
HW_AOT = 255;
HW_BACKUP = 266;
HW_OVERRUN = 277;
HW_CONTROL = 1;
```

2.3.2 Reading data from FPGA

Reading data from FPGA is similar to writing data to FPGA.

Each ID can read eight 16-bit data from FPGA. Reading data is mainly used to read the current highest priority task which needs to execute from the FPGA.

2.3.3 Scheduler Tick Interrupt

Scheduler Tick Interrupt is generated every millisecond by FPGA.

After ARM receive external interrupt, the interrupt service routine read the current highest priority task which needs to execute from the FPGA.

3. THE TASK WE COMPLETED

We have mainly achieved functions as follows: preemptive scheduling, task management, semaphores, message mailboxes, message queues, mutexes and event flags group. We have completed all the necessary components for small real-time embedded systems.

- Support the creation of 8 tasks
- Support the creation of 8 semaphores
- Support the creation of 8 message mailboxes
- Support the creation of 8 message queues
- Support the creation of 8 mutexes
- Support the creation of 8 event flags groups

The following specific describes the various parts of the implementation.

3.1 Preemptive scheduler

There are three key points to realize preemptive scheduler:

When a task signals or sends a message to a higher-priority task, the current task suspended and the higher-priority task is given control of the CPU.

When each tick interrupt comes, if there is a high priority task is ready, high priority task will preemptive low priority task. When a task signals or sends a message to a higher-priority task, the message has been sent, the interrupted task remains suspend and the newer higher priority task resumes.

3.1.1 FPGA

A core job to realize preemptive scheduler is to figure out how to find the highest priority task inside the task ready list. We use the priority encoder to realize it. The method is as follows:

```
function[15:0] code;
    input[7:0] din;
    case x (din)
        8'b1000_0000 : code = 16'h7;
```

```

        8'bx100_0000 : code = 16'h6;
        8'bxx10_0000 : code = 16'h5;
        8'bxxx1_0000 : code = 16'h4;
        8'bxxxx_1000 : code = 16'h3;
        8'bxxxx_x100 : code = 16'h2;
        8'bxxxx_xx10 : code = 16'h1;
        8'bxxxx_xxx1 : code = 16'h0;
        default: code = 16'h7;
    end case
end function

```

We must pay special attention to a point that the idle task is always ready. Idle task has the lowest priority, when there is no task running, idle task will be executed.

3.1.2 ARM

For this hardware real-time systems, we just need pay attention to two points:

Task-level task switching, which is mainly to realize a high priority task switch to a low-priority task.

Interrupt-level task switching, to determine whether there is a higher priority task is ready when interrupt quit and switch to the high priority task.

3.2 Task management

The task management mainly to achieve three functions: Delay time setting, Suspend the task, Task resume.

Each task has 8 configurable registers

Task_REG2 Delay time setting
 = 0 add task to ready list
 = 0xffff delete task for ready list
 = others the task delay time to be set

Task_REG3 Task ID

0-7 8 task ID, read the register can get the current highest priority ready task

Task_REG8 initialization task execution
 = 1 task can be executed
 = others task can not be executed

3.2.1 FPGA

The task will be start when Task_REG8 = 1, every single task running on the FPGA is a separate process but not put them all in one process. This can make full use of hardware real-time system.

```

always @ (posedge clk)
begin
task1 ;
task2 ;
End

```

```

always @ (posedge clk) begin
task1 ;
end
always @ (posedge clk) begin
task1 ;
end

```

3.2.2 ARM

ARM just need simply set the register to configure all tasks.

- Setting the task delay time
Task_REG3 = 0; //set task 0
Task_REG3 = 100; //set task delay time
- Task suspend, delete task form ready list.
Task_REG3 = 0; //set task 0
Task_REG3 = 0xffff; //to suspend task
- Task recovery, add the task to ready list
Task_REG3 = 0; //set task 0
Task_REG3 = 0; //to recovery task

3.3 Semaphore

Semaphore is to establish a flag for shared resources. The flag indicates that the shared resources occupancy. Hardware RTOS can support to create 8 semaphores, each semaphore has 8 registers. Register Description:

Task_REG3 semaphore ID
8 - 15 represents the semaphore ID can be created
Task_REG4 wait semaphore events list
write to this register, add task to this semaphore's wait list.
read from this register, find the highest priority task form wait list.
Task_REG5 Semaphore count, indicates the number of available resources

3.3.1 FPGA

In the implementation of the semaphore, Hardware RTOS not only provide the required scheduler function but also can find out highest priority task in semaphore wait list.

3.3.2 ARM

Mainly provides the following three functions, which is used for the semaphore register initialization and implementation.

- void OSSemCreate(uint16_t ucSemID, uint16_t uiSemCnt);
This function is used to initialize the semaphore
When uiSemCnt = 0 can use semaphore for task synchronization
When uiSemCnt > 0 indicates the number of available resources

- `void OSSemPend(uint16_t ucSemID, uint16_t ucSemTime, uint16_t ucPendTaskID);`
This function is used to request the semaphore
When `ucSemTime = 0xffff` indicates the task suspend until there are available resources.
When `ucSemTime > 0` indicates the task suspend times

- `void OSSemPost(uint16_t ucSemID);`
This function is used to release the semaphore

3.4 Message mailboxes

Message mailbox is mainly used for the transmission of messages between the two tasks. Hardware RTOS support to create 8 message mailboxes, each message mailbox have 8 registers, Registers are described below:

Mbox _REG3 message mailbox ID

16-23 indicates the semaphore ID can be created.

Mbox _REG7 wait message mailbox events list

write to this register, add task to this message mailbox's wait list.

read from this register, find the highest priority task form wait list.

3.4.1 FPGA

In the implementation of the message mail box, Hardware RTOS not only provide the required scheduler function but also can find out highest priority task in message mailbox wait list.

3.4.2 ARM

Mainly provides the following three functions, which is use for the message mailbox register initialization and Implementation.

- `void OSMboxCreate(uint16_t ucMboxID);` This function is used to initialize the semaphore
- Used to create the message mailboxes
- `void *OSMboxPend(uint16_t uiMboxID, uint16_t uiMboxTime, uint16_t uiPendTaskID);`

This function is used to request message mailbox

When `uiMboxTime = 0xffff` Indicates the task suspend until there are available resources.

When `uiMboxTime > 0` Indicates the task suspend times

- `OSMboxPost(uint16_t uiMboxID, void *Pmsg);`

This function is used to send a message

3.5 Message queue

The realization method of the message queue is similar to the message mailbox, but it is necessary to do a circular queue for the message queue used for message's FIFO or LIFO. Hardware RTOS support to create 8 message queues, each message queue have 8 registers. Registers are described below :

Q_REG3 messagequeueID

24-31 Indicates the message queue ID can be create.

Q_REG6 Wait message queue events list

write to this register, add task to this message queue's wait list.

read from this register, find the highest priority task form wait list.

3.5.1 FPGA

In the implementation of the message queue, Hardware RTOS not only provide the required scheduler function but also can find out highest priority task message in queue wait list.

3.5.2 ARM

Mainly provides the following three functions, which use for the message queue register initialization and Implementation.

- void OSQCreate (void **start, uint16_t uiSize, uint16_t uiQueueID);

Used to create the message queue.

- void *OSQPend(uint16_t uiQID, uint16_t uiQTime, uint16_t uiPendTaskID);

This function is used to request message queue

When uiQTime= 0xffff Indicates the task suspenduntil there are available messages.

When uiQTime>0 Indicates the task suspend times

- uint8_t OSQPost(uint16_t uiQID, void *Pmsg);

This function is used to send a message.

3.6 Event flag group

In the real applications practical, The task often need to determine the operation mode of the task according to the result of the amount of a composition of a plurality of semaphore. So we provide event flag group for this. Hardware RTOS support to create 8 event flag group, each event flag group have 8 registers, Registers are described below:

Flag_REG1 wait event flag group' list

write to this register, add task to this event flag group's wait list.

read from this register, find the highest priority task form wait list.

Flag_REG3 event flag group ID

32-39 Indicates the event flag group can be created.

3.6.1 FPGA

In the implementation of the event flag group, Hardware RTOS not only provide the required scheduler function but also can find out highest priority task in event flag group wait list.

3.6.2 ARM

Mainly provides the following three functions, which is used for the event flag group register initialization and Implementation.

- void OSFlagCreate(uint16_t ucFlagID);
- Used to create the event flag group.
- void OSFlagPend(uint16_t uiFlagID, uint16_t uiFlagTime, uint16_t uiPendTaskID, uint16_t uiFlag);

This function is used to request event flag group.

uiFlagID Indicates the flag need to be get .

when uiFlagTime= 0xffff Indicates the task suspend until there are available resources.

When uiFlagTime>0 Indicates the task suspend times

- void OSFlagPost(uint16_t uiFlagID, uint16_t uiFlag);

This function is used to send event flag group

uiFlagID Indicates event flag group which is need to sent.

3.7 Mutual Exclusion Semaphore

Binary semaphore is so easy to cause priority inversion, so we use mutual exclusion semaphore to achieve exclusive use of shared resources. Hardware RTOS support to create 8 mutual exclusion semaphore, each mutual exclusion semaphore have 8 registers, Registers are described below:

Mutex_REG3 mutex ID

40-47 Indicates the mutex ID can be created

Mutex_REG8 Wait mutex events list

write to this register, add task to this mutex's wait list.

read from this register, find the highest priority task form wait list.

3.7.1 FPGA

In the implementation of the mutual exclusion semaphore, Hardware RTOS not only provide the required scheduler function but also can find out highest priority task in mutex wait list.

3.7.2 ARM

Mainly provides the following three functions, which is used for the mutex register initialization and Implementation.

- void OSMutexCreate(uint16_t uiMutexID, uint8_t uNewPrioty);

This function is used to initialize the mutex

- void OSMutexPend(uint16_t uiMutexID, uint16_t uiMutexTime, uint16_t uiPendTaskID);

This function is used to request the mutex

When `uiMutexTime= 0xffff` Indicates the task suspend until there are available resources.
 When `uiMutexTime>0` Indicates the task suspend times

- `void OSMutexPost(uint16_t uiMutexID);`

This function is used to release the mutex

4. DISTRIBUTED DEPLOYMENT

Many modern embedded systems contain more than one processor. For example, a modern passenger car might contain some forty such devices, controlling brakes, door windows and mirrors, steering, airbags, and so forth. Similarly, an industrial fire detection system might typically have 200 or more processors, associated - for example - with a range of different sensors and actuators. Two main reasons:

- Additional CPU performance and hardware facilities
- Benefits of modular design

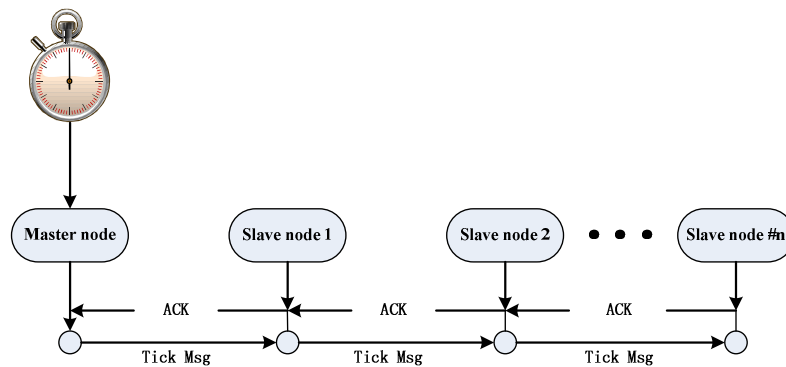


Figure5. S-C scheduler

By using a shared-clock (S-C) scheduler, we can link more than one processor. There are many ways to realize shared-clock scheduler. For examples, using external interrupts, using UART, can bus and so on. Here we will use ZigBee wireless to realize shared-clock scheduler.

5. CONCLUSIONS

The real-time operating system shows more real-time and reliability that based on the hardware implementation. Because the hardware implementation is running independent of the processor running, it does not consume the processing time and processor saves time to execute tasks, so that task scheduling and real-time is improved.

ACKNOWLEDGMENTS

This work was supported by Science & Technology Planning Project of Shenzhen City Grant No. JCYJ20120613112757342.

REFERENCES

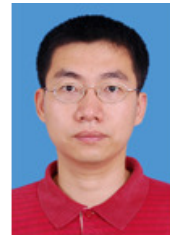
- [1] V.MOONEY III, J.LEE, A.DALEBY, et.al. A comparison of RTU hardware RTOS with a hardware/software RTOS[C]. Design Automation Conferece(ASP_DAC '03), 2003:683-688 .
- [2] V.MOONEY III, BLOUGH D.M.A Hardware-Software real-time operating system framework for SOCs[J]. IEEE Design and Test of Computers Magazine, 2002, 19(6):44-52
- [3] MELISSA VETROMILLE , LUCIANO OST, CESAR A.M.MARCON, et.al RTOS Scheduler Implementation in Hardare and Software for Real time Application [C]. proceedings of the senventeenth IEEE International workshop on rapid system prototyping(RSP 06). 2006:163-168
- [4] T.NAKANO, U.ANDY, M.ITABASHI, et.al. Hardware Implementation of a Real-time Operating System[J]. Proceedings of the Twelfth TRON Project International Symposium IEEE Computer Society Press, Nov,1995:34-32.
- [5] T.NAKANO, U.ANDY , M.ITABSSHI, et.al. VLSI Implementation of a Real-time Operating System[J]. Proceedings of the ASP-DAC '97 Asia and South Pacific, January 1997:679-680
- [6] TTE32-HR2 evaluation microcontroller programming guide. Datasheet and Programming Guide TTE32-HR2 Microcontroller (r1.2): March 2011. This document is copyright © TTE Systems Limited 2007-2011.
- [7] CHEN TIANSHOU, WU XINGLIANG, HU WEI. Research on OS-AwareEmbedded Power-Saving Archiectre[C]. The 2rd Joint Conference on Harmonious Human Machine Environment, HHME2006,PCC'06: 52-59
- [8] ADOMAT J, FURUNAS J, INDH L, etal. Real – time Kernal Hardware RTU: A step towards deterministic and high performance real-time systems[J]. Proceedings of eighth Euromicro Workshop on Real-time Sysrems, 1996:683-688.

AUTHOR

Qiang Huang

Professor of ShenZhen University, P.R. China. born on 1977. Graduated from the University of Liverpool in Electrical Engineering with Ph.D 2004.

He has published more than 30 papers in international journals and conference proceedings, 20 of which were indexed by SCI / EI / ISTP. His research work is supported by Chinese National Natural Science Foundation, Guangdong Province research foundation and Shenzhen Municipal Science-Technology foundation.



MOBILE APPLICATION TESTING MATRIX AND CHALLENGES

Bakhtiar M. Amen¹, Sardasht M. Mahmood² and Joan Lu³

^{1,3}School of Computing and Engineering,
University of Huddersfield, Huddersfield, UK
²Statistics and Computer, College of Commerce,
University of Sulaimani, Sulaimani, Iraq

ABSTRACT

The adoption of smart phones and the usages of mobile applications are increasing rapidly. Consequently, within limited time-range, mobile Internet usages have managed to take over the desktop usages particularly since the first smart phone-touched application released by iPhone in 2007. This paper is proposed to provide solution and answer the most demandable questions related to mobile application automated and manual testing limitations. Moreover, Mobile application testing requires agility and physically testing. Agile testing is to detect bugs through automated tools, whereas the compatibility testing is more to ensure that the apps operates on mobile OS (Operation Systems) as well as on the different real devices. Moreover, we have managed to answer automated or manual questions through two mobile application case studies MES (Mobile Exam System) and MLM (Mobile Lab Mate) by creating test scripts for both case studies and our experiment results have been discussed and evaluated on whether to adopt test on real devices or on emulators? In addition to this, we have introduced new mobile application testing matrix for the testers and some enterprises to obtain knowledge from.

KEYWORDS

Mobile App Testing, Testing Matrix, Automated and Manual Testing.

1. INTRODUCTION

The world of mobile application is emerging rapidly and it attracted extensive research interests [12]. In fact, due to easiness of technology, every day millions of mobile users are depending on their mobile apps to conduct and browse internet for social networking (e.g., Facebook, Twitter, LinkedIn, Instagram), for online banking (transaction and balance sheet), for emailing (arrange meeting and solving problems). According to [23] every year extraordinary numbers of applications are flooding onto the market with forecast of 76.9 billion global downloads in 2014 worth of US\$35 billion [34]. Therefore, the comprehensive mobile application testing is crucial to direct high quality of applications and satisfies user needs, whereas studies indicated that developers are more focusing on the application back end and functionality rather than use experiences. In fact, a user feedback is one of the fundamental parts of application's reputation to ensure app's owners with successful or failure of their application [20]. Commonly, users can easily drop interesting in problematic mobile app, and will abandon it after only one or two failed attempts.

The purpose of this paper is to investigate and provides solutions to firstly; whether agility testing or physical testing is the most appropriate to adopt? Secondly; identify new testing matrix for testers to obtaining knowledge from. Thirdly and finally; introduce new mobile application test

David C. Wyld et al. (Eds) : CCSIT, SIPP, AISC, NLP - 2015
pp. 27–40, 2015. © CS & IT-CSCP 2015

DOI : 10.5121/csit.2015.50403

strategy, testing state-of-art. More to this, we have analysed both case studies MLM and MES results and critically evaluate individual findings for experiment results.

This paper is organised as it follow; Section two is consists of mobile app definition, test definition, mobile test matrix including test environments, test techniques, test levels and test scopes. Section three presents existing mobile app testing tools while section four introduces testing strategy. Section five provides related work. Case studies experiment results illustrated in section six and section seven provide conclusion and future of work.

2. BACKGROUND

This section is consist of three parts; definitions of mobile application, testing definitions and mobile application testing matrix.

2.1 Mobile Application

Mobile application is a written source code in various programming languages (e.g. Java) and designed for smartphones to operate on Mobile OS platforms (e.g. Android, iOS). The purpose of mobile application is to enhance user's daily life throughout (online banking transactions and emails) or for entertainments like (social media and gaming). The novel of mobile app is designed for the user to input data from touch screen and expected output results efficiently and effectively regardless of the application's development knowledge.

2.2 Testing Definitions

Testing defined by [2] [25] [35] is 'the process of executing a program with the intent of finding errors'. In fact, test is one of the fundamental requirements of mobile app development methodology phases in the development life cycle to measure the quality of application's standard and to avoid vital bugs. Due to the rapid growth of mobile apps every year, developers and enterprises are losing confidence in to relays on the best testing techniques and adopt economical ways of delivering mobile apps in to the market [16] [19] [32].

2.3 Mobile Application Testing Matrix

Mobile Apps testing is more complicated than the software or web apps testing due to the nature of development specifications techniques like; OS platforms, devices and screen resolutions [14] [33]. However, we have managed to impalement and organise mobile application testing matrix from [40] to Test Techniques, Testing Environment, Test Level, and Test Scopes as depicted in Figure 1.

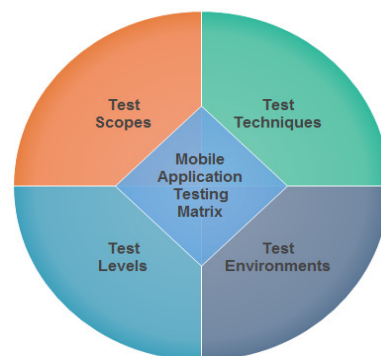


Figure 1: Mobile Application Testing Matrix

2.3.1. Test Techniques

According to Selvam in [40] the principal test challenge arise throughout mobile apps development process “how to test the apps”. The authors of [6] [7] [40] emphasized that, it’s very crucial to decide whether automated or manual testing are the most appropriate testing techniques to adopt in mobile apps testing stage, Figure 2 depicted the techniques. Moreover, we have conducted both techniques for our case studies of MES and MLM in order to obtain our paper’s objective questions. The experiment results of both case studies were demonstrated in the result section with emphasised issues in each technique. On the other hand, researchers are indicating that automated testing is more relying on programming development tool for instance Monkey Talk, Test Plant and other top mobile apps testing tools depicted in Table 3. Whereas, according to the researchers prospective, manual testing is more relying on human interaction like usability testing.

2.3.1.1 Automated Testing

Automated testing technique is highly desirable, for this reason automated testing is capable in decrease of human errors, efficiency in finding bugs, with less time consuming [3]. In fact, automated testing is permit tester to verify the main critical features of the application by testing different data sets [42]. According to Quilter in [39] automated testing has capability to execute large volumes of repeatable scenarios beyond human capabilities to undertake manually.

2.3.1.2 Manual Testing

Manual testing is very time-consuming compare to automated testing, and often it has limitation in testing through the limited user-interface of the mobile device. Manual testing acknowledge tester to create test case and follow the test case design, instruction design to achieve their specific test goals [19] [40]. In addition to this, automated Vs manual results would be demonstrate in the testing results section Seven.

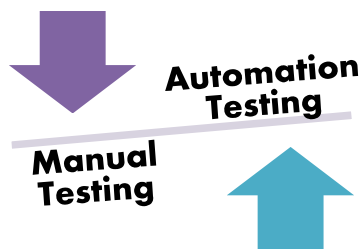


Figure 2: illustrâtes mobile App tests techniques

2.3.2 Test Environments

The critical demands on test environments are widely remained within scientist and enterprises. Kim in [28] argues that ‘developers establish mobile applications on a desktop computer using a development toolkit and emulator’. Therefore, this is indicating that developer is enabling to test on both real device and simulator. Whereas, Simulator has matching look and feel of real device and it executes on desktop operating system. According to Quilter in [39] Simulator-based approaches have various specific advantages like lower cost, scalability, time and easy of implement, opposite to the real device. Figure 3 depicted mobile application environments.

2.3.3 Test Levels

Test level is one of the fundamental crucial parts of mobile application development. Mobile apps test level consists of; Unit Testing [11] [24], Functionality Testing [24], Usability Testing [24] [35], Compatibility Testing, Regressions Testing [24], Security Testing [18][22], Acceptance Testing [18][22] and Network Testing [35].

Table 1 Different Testing Level in Mobile Application

Test Levels	Who does it?	Specification	Why this type?	When is Necessary?	Opacity
Unit Testing [11], [24]	Programmer	Complete the test automatically through run the test script to ensure that the test has turned from "red" (failure) to "green" (success) [11]	To check app code structures to find bugs and errors	When the Programmer wrote a piece codes	White box Testing
Functionality Testing [24]	Programmer	Verifies app/site content (images, text, controls, and links) as it is displayed on the actual mobile devices. [11] [22]	To check the app's functionality and compare the user's requirement	During and after the development stage	Black box and While box Testing
Usability Testing [24][35]	Client, Users	Refer to the app's effectiveness, efficient and satisfaction [24][35]	To check apps link validation, multiple browsers' support, screen resolution. [24]	After app's functionality completed.	Black box Testing
Compatibility Testing	Programmer Independent Tester	Refers to validation of the apps for different operating system, mobile devices [24]	To verify and validate of app's compatibility	When the app completed and before deliverable	Black box and While box Testing
Regressions Testing [24]	Client and Independent tester [24]	Expect apps operating as intends to [24][35]	To ensure the correctness of app's operation	Before the application deployment	Black box and While box Testing
Security Testing [18][22]	Programmer	To ensure with app's encryption/decryption techniques in used sensitive data of users (e.g. ID, Password, Credit card details) [35]	To ensure with information protection landscape [35]	At end of development process	Black box and While box Testing
Acceptance Testing [18][22]	Client	The objective of acceptance testing is to create confidence in the application [18][22]	To Delivery and evaluate the application in aspect of end user point of view	When the user acceptance criteria met with the requirements [18][22]	Black box and While box Testing
Network Testing [35]	Network expertise and Programmer	Compatibility app's with different Network signal (Wi-Fi, 2G, 3G, 4G) Impact of Connectivity Issues [35]	To check app's connection strength and weakness.	Before the app's deliverable phase	While box Testing

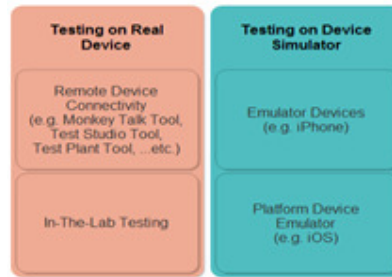


Figure 3: Mobile Application Testing Environments

2.3.4 Test Scopes

Generally, test scopes have been categorised in two major parts, functional (black box) and structural (white-box) [42] [14]. The following table is depicted the classification of each parts.



Table 2 Different Testing SCOPES in Mobile Application








Test Scopes	What is it?	Who does it?	Why this type?
Black Box Testing [14][42][43]	Known as functional and non-functional testing. Black box testing is a widely used in testing environments. The component under test has inputs, outputs, and a specification, which states the relationship between inputs and outputs. It ignores the internal mechanism of a system or component and focuses solely on the outputs generated without prior knowledge of it source code [14][24][42][43]	Independent undertake the test	To detect bugs, errors in the app's codes. Test app's functionalities [24]
White Box Testing [27][37]	Known as structural testing, cover the internal data structures that exercised to ensure validity of test conditions, with good knowledge of the source code [27][31][35][42][43]	Developers Execute This test	To detecting logical errors in the program code (Unit Test) [27]. [37]

3. STATE OF ART

In this section, testing tools that are supporting the testing techniques have been proposed specifically for mobile app testing environments. Each tool has been described in Table 3 in terms of their licenses whether they are open source tools, the table consists of tool's device support for instance Android, iPhone or multi platform as well as tool's scripting and languages. Finally provides the tool's specification testing types support.

Table 3: Mobile apps testing tools

Logo	License	Support Device	Scripting/ Language	Testing Types
	Open Source	Android	JAVA	Unit Testing, GUI interface [9]
 iOS Simulator	Open Source	Window or Mac iOS	Objective C	GUI interface Unit Testing [10]

	Open Source	iPhone & Android etc.	HTML5 & JAVA	Functional, GUI Testing [17]
	Open Source	Multi Platforms	Unit Test	GUI, Accepting Testing [46]
	Cost	iPhone & Android etc.	Test across mobile OS with a single script	GUI Test [44]
	Open Source	Variety of platforms	C#	Google Test UI
	Cost	Multi Platforms	A single Script	GUI Testing [38]
	Cost	Android, iPhone etc.	C#, Java, Perl & Python	Functionality & Speed Performance [13]
	Cost	iPhone, Android etc.	JAVA & Objective C	Functionality, Usability, Performance [26]

4. TEST STRATEGY

Before decide to adopt any test techniques on the mobile apps, it is necessary to have testing strategy in order to meet user's requirements, specifications and to avoid negative feedbacks from app's users. Furthermore, testing progress is important in terms of quality assurance. Figure 4 predicted test strategy plans for the testers to beware of from test document preparation to the application test acceptance/deliverable phase

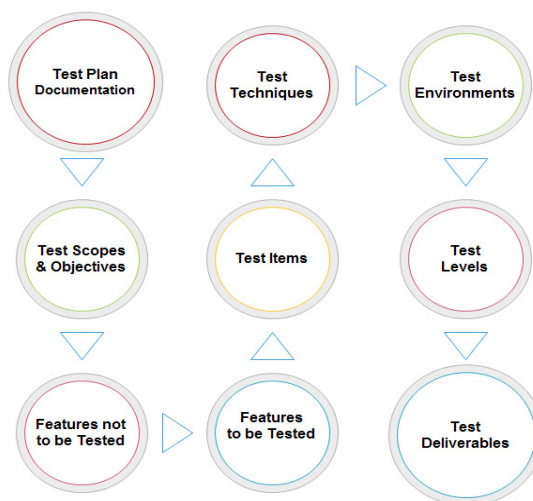


Figure 4: Mobile apps Test strategy plan

5. LITREATURE REVIEW

Haller in [20] proposed agile and compatibility testing for one a case study from Swisscom IT services to identify whether test on wild device and failure during the start-up application and focused on regression testing process.

Amalfitano et al. highlighted the results of automated testing experiment for Android mobile application platform [1]. The author specifically presents a technique for rapid crash testing and regression testing based on crawler that automatically builds models of application in GUI. Heo, Jeongyun et al. in [21] introduced new framework for evaluation of usability in mobile application that has been conducted based on a multi level, hierarchical model of usability factors, the author proposed case study for new framework to test his frameworks on in order to identify the characteristics of the framework.

Utest Inc. proposed usability testing mobile application for NHD Direct in 2011, whereas according to uTest Inc. the application is more likely to focus on symptom checking for mental health conditions, self-care and free advice [45]. Respectively, the objectives of NHS Direct Mobile application usability testing were to enhance the user's feedback and compotator app on top number one in iTunes charts for the best free apps within the first week of released app [4]. Knott suggested that it is necessary to implement some of the specific features of the application manually rather than conduct automated testing [29]. Functional testing consists of both input and output data. However, from the input aspect, mobile application receives two types of actions, whereas the first action is from GUI input by any keyboard keys and touch events, while the second action is the result output.

6. CASE STUDIES

Mobile Lab Mate (MLM) is one of the particular applications developed by the University of Huddersfield research team. The aim of this application is to support students in accessing into their account at anytime in anywhere in order to view their class materials and results effectively and efficiently. Furthermore, MLM application was a pilot case study and attempt to help developer to identify issues within application before the acceptance-testing phase. Figure 5 depicted the applications screen prints. Therefore, both testing techniques such as automated and manual have been conducted and the experiment result will be discussed in the result section.

On the other hand, Mobile Exam System (MES) was another pilot case study that has been conducted. The aim of this application was to support students throughout answering their questions online and save their exam time, to assist teachers to see the results efficiently and avoiding any misconduct mechanism during exam taken. In fact, both techniques of automated and manual testing have been carried out.



Figure 5: MLM & MES Mobile Application

Furthermore, both application case studies have been tested by open source automated tool known Monkey Talk. According to Corbett and Bridgwater 'Monkey Talk is the latest testing

platform from Gorilla Logic [5] [8]. Monkey Talk IDE (Integrated Development Environment) extends with Java Script API and it assist tester to creating, recording, and manage the test on actual devices. However, Monkey Talk is free and open source automated tool operates on iPhone and Android [5] [17]. The reason behind conducting Monkey Talk was to test the applications functionalities and have the test records while emphasis the demands of automated capabilities. Figure 6 depicted the use case design that we have made in the testing process in order to have better and clear of testing objectives

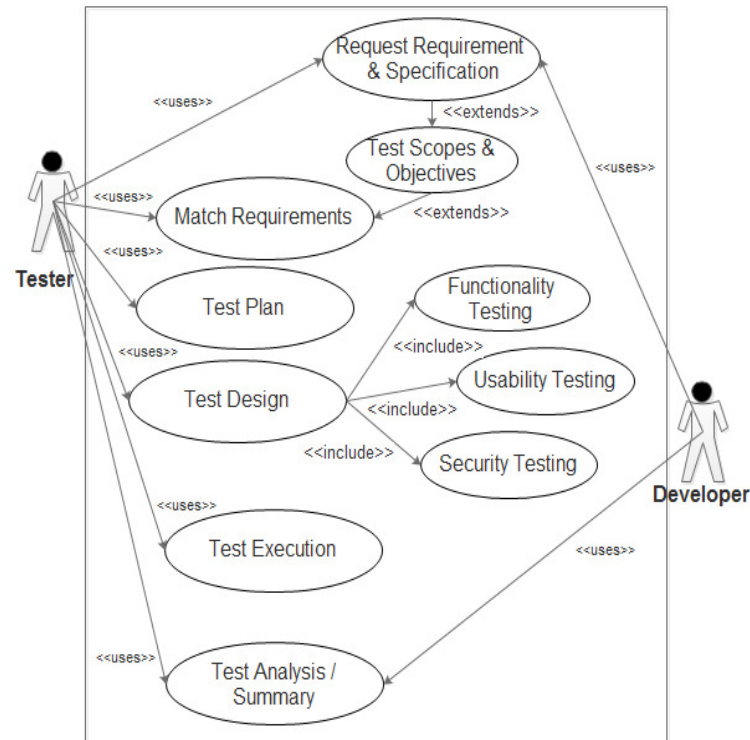


Figure 6: Case Study Use Case Test Process

7. RESULTS (EVALUATION AND ANALYSIS)

Test automated solution consists of: test scripts, connection between automated tool (PC) and the mobile device, remote control mechanism for the device, and an interaction strategy for the mobile device GUI depicted in (Figure 7,8,9 and 11). The selected solution affects the test script language. For example, Expertise, Keynote and Monkey Talk were only tools that capable of testing functionality as well as GUI. When scalable test configuration coverage is the main aim, the test scripts must run on multiple devices and potentially on various OS and OS versions. This requirement affects the connection between a test PC and the mobile device. First, a direct connection can exist from the PC to the device. Second, an indirect connection can exist that acts as a switch between various PCs and a large device pool.

The automated testing is a solution to improve the testing efficiency; it is the most important latest techniques to improve functionality testing as multiple device handlers, and to ensure that MES and MLM applications are resulting automated testing technique efficient and accurately. The following test script was for the MLM app's login function and result of the app's login function has predicted in Figure 8.

```

1) load("libs/MobileLabMate.js");
2) MobileLabMate.Login.prototype.run = function() {
3) this.app.login().run();
4) this.app.link("sign").click();
5) this.app.link("st").click();
6) this.app.input("studentname").enterText("Tester");
7) this.app.input("studentpassword").enterText("u0772370");
8) this.app.button("login").click();
9) this.app.link("Logout").click();
10)};

```

Figure 7: Login Test Script

Figure 8 depicted the test script for MLM new student who has not been registered before.

```

1) load("libs/MES.js");
2) MESapp.CreateAccount.prototype.run = function() {
3) this.app.createAccount().run();
4) this.app.link("sign").click();
5) this.app.link("i").click();
6) this.app.input("name").enterText("Tester2");
7) this.app.input("pass").enterText("1234567");
8) this.app.button("callAjax").click();
9) this.app.link("sign").click();
10) this.app.link("st").click();
11) this.app.input("studentname").enterText("tester2");
12) this.app.input("studentpassword").enterText("1234567");
13) this.app.button("login").click();
14) this.app.link("Logout").click();};

```

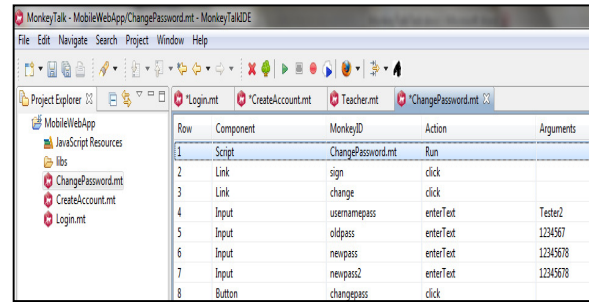
Figure 8: Create Account Test Script

```

1) load("libs/MobileLabMate.js");
2) MobileWebApp.ChangePassword.prototype.run = function() {
3) this.app.changePassword().run();
4) this.app.link("sign").click();
5) this.app.link("change").click();
6) this.app.input("usernamepass").enterText("Tester2");
7) this.app.input("oldpass").enterText("1234567");
8) this.app.input("newpass").enterText("12345678");
9) this.app.input("newpass2").enterText("12345678");
10) this.app.button("changepass").click();
11)};

```

Figure 9: chanhe password test script



Row	Component	MonkeyID	Action	Arguments
1	Script	ChangePassword.mt	Run	
2	Link	sign	click	
3	Link	change	click	
4	Input	usernamepass	enterText	Tester2
5	Input	oldpass	enterText	12345678
6	Input	newpass	enterText	12345678
7	Input	newpass2	enterText	12345678
8	Button	changepass	click	

Figure 10: change password test result

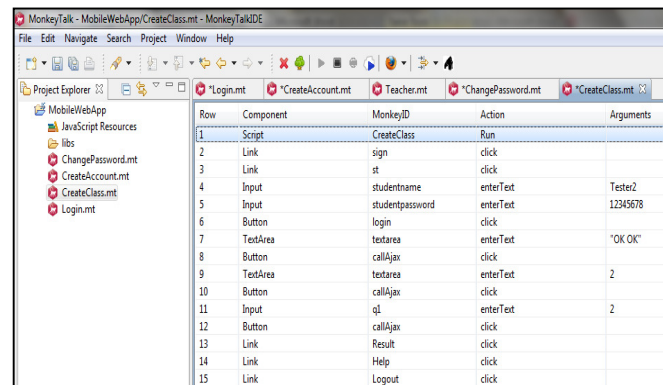
Figure 10 depicted the new class for the students and teachers screen print results and test scripts

```

1) load("libs/MobileLabMate.js");
2) MobileWebApp.CreateClass.prototype.run = function() {
3) this.app.createClass().run();
4) this.app.link("sign").click();
5) this.app.link("st").click();
6) this.app.input("studentname").enterText("Tester2");
7) this.app.input("studentpassword").enterText("12345678");
8) this.app.button("login").click();
9) this.app.textArea("textarea").enterText("OK OK");
10) this.app.button("callAjax").click();
11) this.app.textArea("textarea").enterText("2");
12) this.app.button("callAjax").click();
13) this.app.input("q1").enterText("2");
14) this.app.button("callAjax").click();
15) this.app.link("Result").click();
16) this.app.link("Help").click();
17) this.app.link("Logout").click();
18) };

```

Figure 11: Test script for new class



Row	Component	MonkeyID	Action	Arguments
1	Script	CreateClass	Run	
2	Link	sign	click	
3	Link	st	click	
4	Input	studentname	enterText	Tester2
5	Input	studentpassword	enterText	12345678
6	Button	login	click	
7	TextArea	textarea	enterText	"OK OK"
8	Button	callAjax	click	
9	TextArea	textarea	enterText	2
10	Button	callAjax	click	
11	Input	q1	enterText	2
12	Button	callAjax	click	
13	Link	Result	click	
14	Link	Help	click	
15	Link	Logout	click	

Figure 12: Test result for new class

On the other hand, one of the most important aspects was to consider and carry out functionality, usability and security testing. MLM application was operated normal, but still there were some bugs existed in the application during the functionality of “forgot password” link. However, change password functionality was not crucial and secure Figure 14 depicts the result of MLM functionality, usability as well as security.

In fact, due to the limited space, we have only illustrated a few initial test scripts while for each application of MLM and MES have had several test scripts. In fact, MES app was very secure in the aspects of authorisation, encryption and data store, but MLM apps has had some bugs within the application when the user have access to make more than 16 characters for username and password while in MLM user only unable to enter different characters accept numbers and letters between 8-20 length spaces. MLM apps do not have the limitation input. Therefore, these are some of weak points in MLM for the hacker to inject the database by random characters.

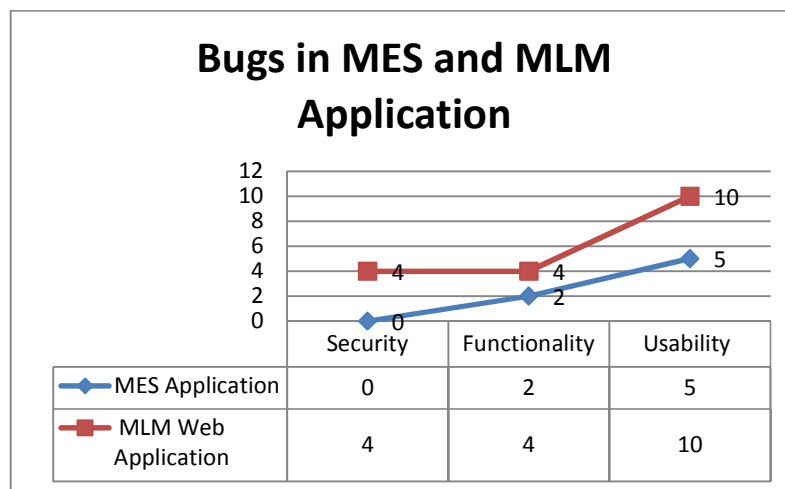


Figure 13: Bugs in both MES and MLM applications

Furthermore, testing functionality and usability activities were performed by real device as well as automated tool for each applications. Figure 13 indicates that MES apps have more bugs compare to MLM apps from manual testing results. Finally, the test scrip results are demonstrates that some functionality of MLM is not working as intends to do by automated limitation while they were working effectively on the real devices. On the other hand, the source code of the test scrip’s in Figure (7,8,9 and11) illustrated that some of functional of the MLM application is not structured accurately when two users were enable to create new account within the same email address and type in long characters or digits in the password field. However, from both case studies we have managed to highlight the limitation of each automated and manual testing in Table 4.

Table 4 DISTINCTIONS between Automated Testing and Manual Testing

Automated Testing	Manual Testing
<ul style="list-style-type: none"> - Testers require to conducting specific tool to execute the test. - Cost effectiveness - Programming knowledge is required. - Less staff’s required but tools are expensive. - Difficult to depend on automated, some app’s area still has to test manually. 	<ul style="list-style-type: none"> - Tester has to write a test case and executes on the application manually. - More likely to cost. - Not programming knowledge is required. - Skilled testers and staffs required. - Testing apps on real device is time-consuming [41]

<ul style="list-style-type: none"> - Automated avoid overloaded work and save more times. - Requirements does not changing frequently. 	<ul style="list-style-type: none"> - Staffs Training expensive. - Manual testing is more time considered to perform a test case. - Requirements more likely to changing frequently.
--	--

8. CONCLUSION AND FUTURE WORK

This paper is managed to answer the most demandable questions related to each mobile app's testing techniques, whether to conduct automated or manual testing. Tests were executed for both case study applications by Monkey Talk open source in order to identify bugs and errors in both case studies. Moreover, it is difficult decision for the testers to decide whether adopt automated or manual testing environments, for this reason, tester has to investigate in selected tool's limitation before the testing strategy had has planned. In fact, it is necessary for the testers to keep in mind testing objectives, testing has to be performed on many combination of devices, browsers, and operating systems rather than just depends on one test technique. Automated testing cannot to be judged by manual testing, for the following reasons:

1. Automated testing has only functional capabilities.
2. Automated testing has benefits of reducing time and cost.
3. Usability testing difficult to be conducted by automated testing.
4. More tests can be run in a shorter time in automated.

Finally, In order to obtain higher standard and quality mobile applications feedback, testing different activities throughout the application's development process and effective models, methods, techniques and tools are essential to be considered by the testers. Furthermore, we highly recommend testers to conduct both test techniques of automated and manual in order to cope with the fundamental necessity of the rapid delivery of these applications, for these reasons, combined both testing techniques will assist testers to identify some of the bugs and errors within the apps efficiently while it might be difficult to identifies them in automated testing on the real devices as we have predicted in our case studies result. To conclude, Automated testing is one of the efficient methods to guarantee of app's quality and performance within the mobile testing environments compare to manual testing. In the future, we implement our Mobile App's Testing Matrix and Testing Strategy in several real time applications within enterprises in order to enhance one powerful test technique for the testers to relays on.

REFERENCES

- [1] Amalfitano, D. Fasolino, A. Tramontana, P. and Federico, N. (2011). A GUI Crawling-based technique for Android Mobile Application Testing.
- [2] Bach, J.(1999). General Functionality and Stability Test Procedure. [online] Available at: <http://www.satisfice.com/tools/procedure.pdf> [Accessed 15th August 2014].
- [3] Bartley, M.(2008). "Improved time to market through au tomated software testing". Automation Testing, [Online] Available at: <http://www.testingexperience.com> [Accessed 23rd September 2014].
- [4] Bastien, C.(2008). "Usability Testing : A Review of Some Methodological and Technical Aspects of the Method." In ternational Journal of Medical Informatics 79(4): e18–e23. [Online] Available at: <http://dx.doi.org/10.1016/j.ijmedinf.2008.12.004>. [Accessed 23rd November 2014]. Available at:
- [5] Bridgwater, A.(2012). MonkeyTalk Tests iOS/Android Apps In The Mobile Jungle. [online] <http://www.drdoobs.com/open-source/monkeytalk-tests-iosandroid-apps-in-the/232602236> [Accessed 13th October 2014].
- [6] Brown, M.(2011). Mobile Developers Get Best Practices For App Testing. [online] Available at: <http://www.mobileapptesting.com/mobile-developers-get-best-practices-for-app-testing/2011/03/>. [Accessed 3rd September 2014].

- [7] Brown, M.(2011). MFiobile Functional Testing: Manual or Automated?. [online] Available at: <http://www.mobileapptesting.com/mobile-functional-testing-manual-orautomated/2011/05/>. [Accessed 8th August 2014].
- [8] Corbett, J.(2012). Quality Assurance testing in a mobile world. [online] Available at: <http://www.peteramayer.com/quality-assurance-testing-in-a-mobile-world>>. [Accessed 26th October 2014].
- [9] Developer.android.com (n. d). Testing. Retrieved from <https://developer.android.com/tools/testing/index.html> [Accessed 6th November 2014].
- [10] Developer.apple.com (2014). Testing and Debugging in iOS Simulator. Retrieved from https://developer.apple.com/library/ios/documentation/ides/conceptual/iOS_Simulator_Guide/TestingontheIOSSimulator/TestingontheIOSSimulator.html [Accessed 6thNovember 2014].
- [11] Dimitrios, V. (2013). "Estimation for Unit Root Testing."
- [12] Dongsong, Z. and Adipat, B. (2005). "Challenges Metho dologies, and Issues in the Usability Testing of Mobile Applications", International Journal of Human Computer Interaction, Vol. 18, 3
- [13] Experitest.com (n. d). SeeTestAutomation. Retrieved from <http://experitest.com/support/getting-started/download-2-2/> [Accessed 6th November 2014].
- [14] Freeman, H. (2002). "Softawre Testing". IEEE Instrumentation & Measurement Magazine, P.48-50. [online] Available at: <https://confluence.xpeqt.com/confluence/download/attachments/10322031/01028373.pdf> [Accessed 23rd June 2014].
- [15] Galorath, D. (2012). Software Project Failure Costs Billions. Better Estimation & Planning Can Help. [online] Available at: <http://www.galorath.com/wp/software-project-failure-costs-billions-better-estimation-planning-can-help.php> [Accessed 16 September 2014]
- [16] Gao Z. and Long, X. (2010). Adaptive Random Testing of Mobile Application. P.297, Vol 2. [Online] Available at: <http://ieeexplore.ieee.org.libaccess.hud.ac.uk/stamp/stamp.jsp?tp=&arnumber=5485442> [Accessed 26rd September 2013].
- [17] Gorillalogic.com (n.d). MonkeyTalk Open source Automation Testing Tool. [online] Available at: <https://www.gorillalogic.com/monkeytalk> [Accessed 18 August 2014].
- [18] Graham, D., Veenendaal, E., Evans, I. and Black, R. (2008). Foundations of Software Testing: ISTQB Certification. UK: Cengage Learning EMEA.
- [19] Harty, J.(2009). "A Practical Guide to Testing Wireless Smartphone Applications". Synthesis Lectures on Mobile and Pervasive Computing. 4(1), pp. 1 – 99. [online] Available at: www.morganclaypool.com [Accessed 11th October 2014].
- [20] Haller, K.(2013). "Mobile Testing". ACM SIGSOFT Software Engineering Notes, 38, 6, 1-8. [Online] Available at: <http://doi.acm.org/10.1145/2532780.2532813> (November 3rd, 2014).
- [21] Heo,J., Ham, D., Park, S., Song, C. and Yoon, W. (2009). "A Framework for Evaluating the Usability of Mobile Phones Based on Multi-level, Hierarchical Model of Usability Factors," Interacting with Computers, Vol. 21(4), pp. 263-275
- [22] Hetzel, B.(1998). The complete guide to software testing. (2nd ed.). Chichester: John Wiley & Sons.
- [23] Jacob, J.and Tharakan, M. (2012). "Roadblocks and their workaround, while testing Mobile Applications".The Magazine for Professional Testers. P8-16, Vol 19. [Online] Available at: <http://www.testingexperience.com/> [Accessed 23rd September 2014].
- [24] Jorgensen, P. (2002). "Software testing: acraftman's approach," CRCPress, p.359.
- [25] Kaner, C. (2002). Black Box Software Testing (Professional Seminar), Risk Based Testing And Risk-Based Test Management, [online] Available at: www.testing-education.org> [Accessed 12 August 2014]
- [26] Keynote.com (n. d). Mobile Testing. Retrieved from <http://www.keynote.com/solutions/testing/mobile-testing> [Accessed 6th November 2014].
- [27] Khan, M.and Sadiq, M. (2011). "Analysis of Black Box Software Testing Techniques: A Case Study". IEEE. pp. 1-5. [online] Available at: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6107931> [Accessed 2nd May 2014].
- [28] Kim, H., Choi, B. and Wong, W (2009). Performance Testing based on Test-Driven Development for Mobile Applications.P612-617. [online] Available at: <http://dl.acm.org.libaccess.hud.ac.uk/citation.cfm?id=1516349> [Accessed 24rd July 2014].
- [29] Knott, D. (2012). "The Enterprise Mobile Applications Development Framework", Best Practices in Mobile App Testing.P26, [online] Available at: <http://www.testingexperience.com/> [Accessed 23rd June 2014].

- [30] Kumiega, A. and Vliet, B. (2008). *Quality Money Management: Process Engineering and Best Practices for Systematic Trading and Investment*. USA: Academic Press.
- [31] Lewis, W. (2008). *Software Testing and Continuous Quality Improvement*. (3rd ed.). United States: Auerbach.
- [32] Liu, Z. Gao, X. Long, X. (2010). "Adaptive Random Testing of Mobile", *Computer Engineering and Technology (ICCET)*, 2010 2nd International Conference on , vol.2, no., pp.V2-297,V2-301, 16-18. [online] Available at: <http://ieeexplore.ieee.org.libaccess.hud.ac.uk/stamp/stamp.jsp?tp=&arnumber=5485442&tag=1> [Accessed 24rd June 2014].
- [33] Milano, D. (2001). *Android Application Testing Guide*. USA: Packt Publishing Ltd.
- [34] Muccini, H., Di Francesco, A. and Esposito, P. (2012), "Software testing of mobile applications: Challenges and future research directions," *Automation of Software Test (AST)*, 2012 7th International Workshop on , vol., no., pp.29,35 [online] Available at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6228987&isnumber=6228973> [Accessed 23rd September 2012].
- [35] Myers, G., Badgett, T. and Sandler, C. (2012) *The Art of Software Testing*. (3rd ed.). Hoboken, New Jersey: John Wiley & Sons, Inc.
- [36] Naik, S. and Tripathy, P. (2008) *Software Testing and Quality Assurance: Theory and Practice*. Hoboken: John Wiley & Sons, Inc.
- [37] Nidhra, S. and Dondeti, J. (2012). "Black Box and White Box Testing Techniques- A Literature Review". *International Journal of Embedded Systems and Applications (IJESA)* Vol.2, No.2 [online] Available at: <http://aircse.org/journal/ijesa/papers/2212ijesa04.pdf> (Accessed November 12, 2014).
- [38] Perfectomobile.com (n.d). *Test Automation*. Retrieved from <http://www.perfectomobile.com/solution/test-automation> [Accessed 6th November 2014].
- [39] Quilter, P. (2011). "Automated Software Testing with Traditional Tools & Beyond". *Automated Software Testing with Traditional Tools & Beyond*. P20, Vol 3 (5). [Online] Available at: www.automatedtestinginstitute.com[Accessed 23rd July 2014].
- [40] Selvam, R. (2011). 'Mobile Software Testing – Automated Test Case Design Strategies. *International Journal on Computer Science and Engineering*' (IJCSE) Vol.3.
- [41] She, S., Sivapalan, S. and Warren, I., (2009). *Hermes: A Tool for Testing Mobile Device Applications*. *Software Engineering Conference, 2009. ASWEC '09. Australian* , vol., no., pp.121,130, 14-17 [Online] Available at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5076634> [Accessed 12th October 2014].
- [42] Sofokleous, A. and Andreou, A. (2008). "Automatic, evolutionary test data generation for dynamic software testing". *Journal of Systems and Software*.P 1883–1898, Vol 81 (11). [online] Available at: <http://www.sciencedirect.com.libaccess.hud.ac.uk/science/article/pii/S0164121208000101> [Accessed 24rd July 2014].
- [43] Stottlemeyer, D. (2001). *Automated Web Testing Toolkit: Expert Methods for Testing and Managing Web Applications*. Canada: John Wiley & Sons.
- [44] Testplant.com (n. d). *eggplant*. Retrieved from <http://docs.testplant.com/?q=content/using-eggplant> [Accessed 6th November 2014].
- [45] uTest.com (2011). *Software Security Testing. The Keys to Launching a Safe, Secure Application* [online] Available at: http://c0954852.cdn.cloudfiles.rackspacecloud.com/uTest_eBook_Mobile_Testing.pdf [Accessed 20 Septemeber 2014].
- [46] Wynne, M. and Hellosoy, A. (2012). *The Cucumber Book: Behaviour-Driven Development for Testers and Developer*. (1st ed.) North Carolina: Pragmatic Programmers, LLC.

AUTHOR

Bakhtiar M. Amen is a PhD research candidate in School of Computer Science at the University of Huddersfield. Bakhtiar holds BSc in Software Engineering and MSc in advanced computer science at the University of Huddersfield. Bakhtiar's research interests are in the areas of mobile application testing, mobile data age, cloud computing, big data and big data analytics, distributed computing, and Internet services. He has published over 4 international journal and conference papers. He is a recipient of 2013 VC Scholarship from the University of Huddersfield. He is a member of British Computer Society BCS.



AN OVERVIEW OF FRAGMENTATION DESIGN FOR DISTRIBUTED XML DATABASES

Kok-Leong Koong, Su-Cheng Haw and Lay-Ki Soon

Faculty of Computing and Informatics,
Multimedia University, 63100 Cyberjaya, Malaysia

ABSTRACT

XML is a standard of data exchange between web applications such as in e-commerce, e-learning and other web portals. The data volume has grown substantially in the web and in order to effectively retrieve or store these data, it is recommended to be physically or virtually fragmented and distributed into different nodes. Basically, fragmentation design contains of two parts: fragmentation operation and fragmentation method. There are three different kinds of fragmentation operation: Horizontal, Vertical and Hybrid, determines how the XML should be fragmented. The aim of this paper is to give an overview on the fragmentation design consideration.

KEYWORDS

XML Database, Distributed Design, Fragmentation Distributed XML

1. INTRODUCTION

XML is a semi-structured, self describing and human-readable document. A native XML document is stored in a plain text format and thus it can be easily processed by any applications and systems. XML and HTML are both subset of Standard Generalized Markup Language (SGML) [1]. And, HTML is commonly used in web environment. It makes XML a good option for data exchange in web environment. Thus, XML has started to become a standard of data exchange between applications and systems. It has been extensively used in web environment and data exchange between web applications. However, as the nature of XML, it is also commonly used in standalone applications to store metadata or application data.

The emerging of smart phone and tablet market has generated big volume of data and it grown exponentially in every minute. This gigantic volume of data also has been named as Big Data. The cohesiveness between these data is low as data might or might not be related to each other. Thus, XML is a good choice to be used to handle these data. However, large volume data will be only effective to be stored and retrieved in distributed model as it can be making used of the parallelism processing.

There are three main advantages on distributed large database. First of all, a distributed system may require multiple normal specification computer system rather than a very high specification computer system. Thus, it will lower the cost but sustain the high performance on the distributed database. Secondly, it also increased scalability. There is always a boundary for a database to

expand within a single computer system. When it is design to be distributed, the database can expand beyond a single computer system. Thirdly, it will increase the availability. Normally, distributed design database will be replicated. This will make the database more resistance to the failure of a single computer system [2]. Thirdly, it will increase the performance of the database system as it used parallelism processing to store and retrieve data from the database system [3].

Distributed design of database normally includes three basic steps: fragmentation, allocation and replication [4]. Nevertheless, the focus on this paper is on fragmentation. Fragmentation is a process of divide database into smaller fragments. Fragmentation contains two steps: determine a fragmentation model to be used and select a method or an algorithm to use for the fragmentation. In the first step, it determines what structure or model of fragmentation to be used. It can be horizontal, vertical or mixed. In the second step, it determines how the data should be fragmented into fragments. It also sometimes refers to fragmentation method or technique.

The rest of the paper is organized as follows. Section 2 outlines the factors driven to distribution database. Section 3 gives an overview on fragmentation models, followed by Section 4, which discusses on the fragmentation methods. Section 5 presents our discussion. Finally, Section 6 concludes the paper.

2. FACTORS DRIVEN TO DISTRIBUTION

Main driving forces for having distributed database include:

- Lower costs: having distributed architectures made of hundreds of PC computers proves to be much cheaper and even more powerful the one mainframe systems serving hundred terminals
- Increased scalability: adding a new network node is the easiest way to respond to extensibility needs of the company,
- Increased availability: by replicating data over several network nodes data becomes closer to the end user and more resistant to system failures,

3. FRAGMENTATION MODEL

3.1. Fragmentation Model for Traditional Databases

There are three basic types of fragmentation models in traditional databases such as relational database and object oriented database. There are horizontal, vertical and mixed [5].

In the relational database, horizontal fragmentation referring a fragmentation database at the record, row or tuple level [3, 6]. To illustrate the scenario, assume a simple relational database contains the following fields in each record: *name*, *gender*, *address*, *phone*, *income* and *tax_id*. There are 56,000 records are stored in a single table for recording 56,000 person data (Table 1). A simple horizontal fragmentation might result into the first node storing the first 28,000 records and the second node storing the last part of 28,000 records. The structure of the fragmentation will be look similar to Figure 1.

Table 1. Sample data of Person Table

name	gender	address	phone	income	tax_id
xxx	x	xxxx	xxxx	xxxx	xxxx
yyy	y	yyyy	yyyy	yyyy	yyyy
.....					
Ooo	o	oooo	oooo	oooo	oooo

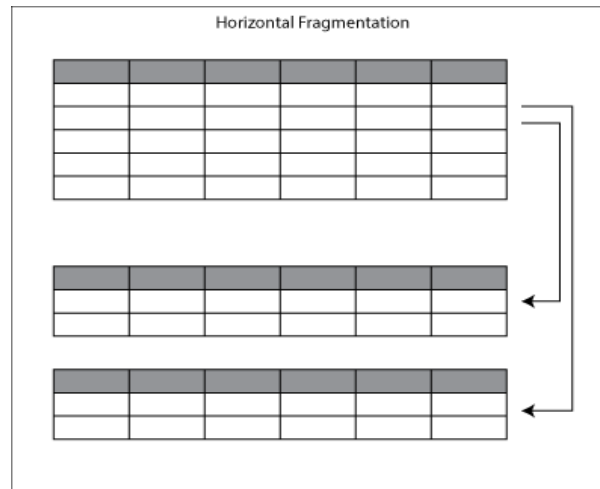


Figure 1: Horizontal Fragmentation for relational database

On the other hand, vertical fragmentation referring a fragmentation database by grouping fields or attributes of records. Using the previous relational database show on Table 1, vertical fragmentation will split this database by grouping fields such that it might group *name*, *gender*, *address* and *phone* fields and store in first node, while *income* and *tax_id* into the second node. The fragmentation structure will be look similar to Figure 2.

The mixed or hybrid is a combination of both horizontal and vertical fragmentation. It can be split horizontally then vertically or vice versa. Using the same relational example, a mixed can first split horizontally by grouping records that belong to particular level of income. Then, split further on the current records by splitting *name*, *gender*, *address* and *phone* on other node and the rest of the data *income* and *tax_id* on other node.

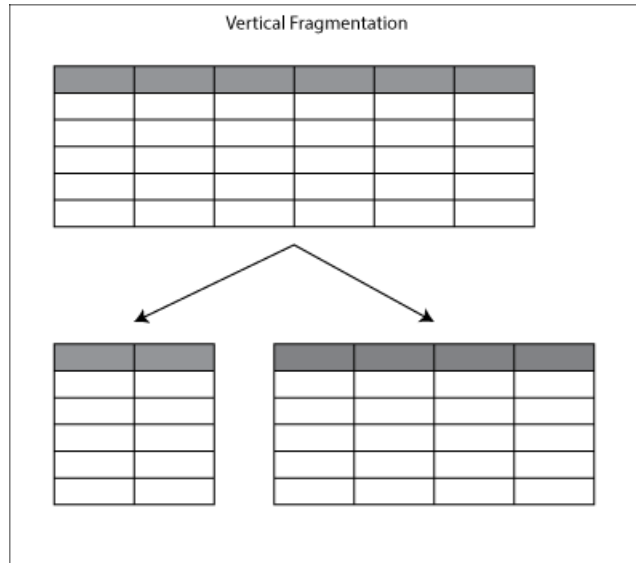


Figure 2: Vertical Fragmentation for relational database.

Object oriented database is totally different from relational database. The data is stored in object form and can be illustrated in a hierarchical or tree format. Fragmentation in object oriented has increased complexity of its hierarchical structure, methods or properties within an object [7]. In term of structure, XML is quite similar to object oriented database. Fragmentation in object oriented share the same fragmentation model like relational database aside the complication involved in object oriented database. It can be fragmented in horizontal, vertical or mixed. Figure 3 and Figure 4 shown the concept how object oriented database can be fragmented into horizontal and vertical model respectively.

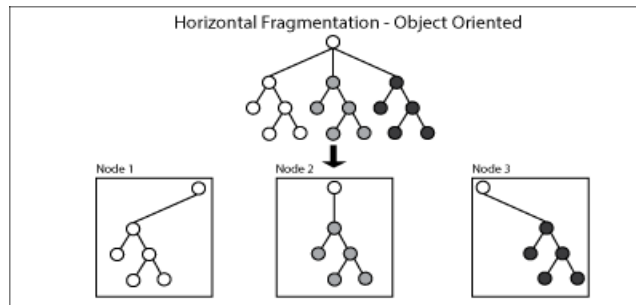


Figure 3: Horizontal Fragmentation for object oriented database

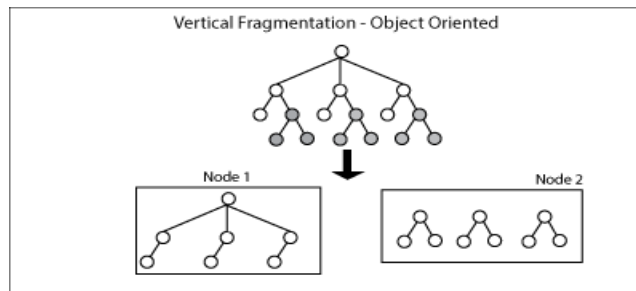


Figure 4: Vertical Fragmentation for object oriented database

3.2. Fragmentation Model for XML Databases

In general, there are only three types of fragmentation models: horizontal, vertical and mixed in XML distributed design [6]. As relational database and object oriented database has started to develop distributed design earlier than XML, the fundamental concept in XML fragmentation also referencing to these two databases. Initially, XML is introduced to run in a single machine. However, as the size of the data grow substantially and it needs to be distributed form in order to achieve better performance on retrieving and storing the data.

Generally, the fragmentation models can be broadly classified into Horizontal, Vertical and Hybrid. The following subsections briefly explain each model.

3.2.1. Horizontal Fragmentation

In XML, horizontal fragmentation can be achieved by selection. Selection is based on the pre defined conditions on splitting the fragments. A horizontal fragment f_i is determined by the selector operator σ of predicates p over collection of elements E in a homogeneous XML document. It can be written so that $f_i = E(\sigma_{p_i})$. Assume we have a XML document constructed according to the relational database stated in the previous section. If the simple selection predicate of p_1 such that /person/employee/income to be income level less than or equal 5000 and p_2 to be income level more than or 5000, thus fragments will be written as $f_1 = E(\sigma_{p_1})$ and as $f_2 = E(\sigma_{p_2})$.

From Figure 5, employee elements with the of name Wong Wei Wei and Lee Jia Fong will be then split and stored as a new XML document in node 1 as first fragment and the rest of the elements of employee will stored in node 2 as a new XML document.

After the operation, node 1 and node 2 may have DTD like <!DOCTYPE person (employee*)> and <!ELEMENT employee (name, gender, contacts, income, tax_id)>.

Horizontal fragmentation is recommended when the query criteria is based on particular attribute that used as selection predicate to fragmenting the XML database. In this scenario, horizontal fragmentation may reduce the transportation cost and processing time as the data is determined in a specific distributed note. Moreover, horizontal fragmentation can easily transport data between sites to improve system performance [8].

Using the same XML database in this document as an example, we use /person/employee/income as the attribute for selection predicate to fragment the database horizontally. Assume this XML database has been fragmented into 5 nodes with income level as the selection predicate of the following categories: 0-999, 1000-1999, 2000-2999, 3000-3999, 4000 and above. If a query searching for a person detail information with income level of 3000-3999, these data can be obtain by querying the fourth distributed nodes thus the query will be able to locate the data in minimum time and retrieve the data with lease processing time.

3.2.2. Vertical Fragmentation

Vertical fragmentation can be achieved by projection. It will split the data structure into smaller parts as particular selected child elements will be split and stored as fragment in other node. A vertical fragment f_i is determined by the projection operator π by path selection p over collection of element E in a homogeneous XML document. It can be written so that $f_i = E(\pi_{p_i})$. If the path selection p_i is /employee/contact, all the children elements under this tree path will be split and stored in other node. In this case, fragment $f_1 = E(\pi_{p_1})$ represents all contact elements in the XML document will be split and stored in node 2. And, the remaining elements will be stored in node 1.

After the operation, node 1 may have an DTD like `<!DOCTYPE person (employee*)>` and `<!ELEMENT employee (name, gender, income, tax_id)>`. And, node 2 may have an DTD like `<!DOCTYPE contacts (contact*)>` and `<!ELEMENT contact(address,phone)>`. In order to create reference link between these two nodes, at least one reference attribute is required for the element that will be able to refer back to elements that resided in other node or site [9].

Vertical fragmentation is a kind of affinity-based fragmentation. As opposed to horizontal fragmentation, this type of fragmentation does not encourage transportation of data from node to node which will trade off flexibility to affinity [8].

Assume a particular employee data is needed with a provided phone contact as search criteria. First the contact elements in the node 2 that match search criteria will be searched. If this entry found, the reference attribute will be used to access the employee data in node 1.

3.2.3. Hybrid Fragmentation

Hybrid fragmentation or sometimes also referring to mixed fragmentation uses both horizontal and vertical fragmentation by taken advantages of both models. It operates in the way where a horizontal fragmentation will be implemented to split the document into horizontal fragment and then further fragmented from these fragment by implementing vertical fragmentation.

A hybrid fragment f_i is determined by the horizontal and vertical fragmentation implemented. It is depend on how you would like to implement the hybrid into the XML document. It can be split horizontally then vertically or vice versa.

Assume you would like to do it horizontally then vertically. First fragment the document horizontally and called this fragment f_a . Thus, $f_a = E(\sigma_{pi})$ and from these f_a fragments we further fragmented them vertically such that the hybrid fragment $f_{i=f_a}(\pi_{pi})$.

Assume we use income level as the selection condition in horizontal fragmentation, and vertically fragment further with the path `/employee/contact` as previous example. There will be 4 hybrid fragments generated for 4 nodes.

After the operation, node 1 and node 2 may have DTD like `<!DOCTYPE person (employee*)>` and `<!ELEMENT employee (name, gender, income, tax_id)>`. Node 3 and node 4 may have DTD like `<!DOCTYPE contacts (contact*)>` and `<!ELEMENT contact(address,phone)>`. It will look exactly like in vertical fragmentation as its final operation is using vertical fragmentation. However, each node contains only two records instead of four records using vertical fragmentation.

Hybrid fragmentation is the combination of horizontal and vertical fragmentation which getting advantages of both fragmentations. In the above scenario, the search can be limited only to particular income level. At the same time, data can be also obtained from vertical fragments by contact element and then with the reference link to the particular employee.


```
<?xml version="1.0" encoding="utf-8"?>
<person>
  <employee>
    <name>Wong Wei Wei</name>
    <gender>Female</gender>
    <contact>
      <address>Sungai Besi</address>
      <phone>03-91234567</phone>
    </contact>
    <income>3500</income>
    <tax_id>120000</tax_id>
  </employee>
  <employee>
    <name>Lee Jia Fong</name>
    <gender>Male</gender>
    <contact>
      <address>Batu Pahat</address>
      <phone>07-71234567</phone>
    </contact>
    <income>2500</income>
    <tax_id>120001</tax_id>
  </employee>
  <employee>
    <name>Tan Jung</name>
    <gender>Male</gender>
    <contact>
      <address>Genting</address>
      <phone>05-34564567</phone>
    </contact>
    <income>6000</income>
    <tax_id>120002</tax_id>
  </employee>
  <employee>
    <name>Fan Wei Tong</name>
    <gender>Female</gender>
    <contact>
      <address>Kuala Lumpur</address>
      <phone>03-71234567</phone>
    </contact>
    <income>9000</income>
    <tax_id>120003</tax_id>
  </employee>
</person>
```

Figure 5: XML sample

4. FRAGMENTATION METHODS

Fragmentation model only define the fragmentation structure in distributed design. Fragmentation method is required to determine how the data should be fragmented (horizontally,

vertically or hybrid). Fragmentation by arbitrary cutting document in to fragments horizontally, vertically, or hybrid may not necessary effectively improve the query performance. Thus, some fragmentation methods have been introduced. These proposed methods have their own advantages and disadvantages against difference scenario. These methods can be group into four categories: structure and size, query and cost, predicate and holes and fillers (for streamed data).

4.1. Structure and Size

Fragmentation of XML document can be fragmented based on structure and size of the document. The structural information about the document can be obtained from the document schemata (DTD or XML Schema). The structure information also can be obtained by transverse the XML document. There is an advantage of this fragmentation method which balanced the load of site or nodes processing power. And this will lead to more effectively uses of resource and improve query performances.

Skewed query processing problem is a well known problem in distributed design. It merely indicated an unbalance on loading on particular distributed node against other nodes. And, this method of fragmentation can resolve particular skewed query processing problem as the fragment is properly distributed according to the structure and size of the document.

To fragment document using structure and size method, first of all, the document is required to be parsed. In other words, map the XML document into a tree structure. This parser is either tree-based or event-based. A tree-based parser may consume memory resources as it transverse the whole document and save all the relationship and node of these nodes in the memory. DOM is a tree-based parser. On the other hand, event-based consumed less memory. It does not construct a large tree in memory as it only scan particular element, attribute, content sequence in an XML document [10].

In structure and size method, event-based parser is used to construct vertex/node list, structural information. After obtained this information, the document then fragmented accordingly.

A typical example using this method on horizontal fragmentation can be achieved by determining a threshold size of the each fragment. Then, transverse throughout the XML document by determine the size of a single level child node horizontally. If the size of the child node including its descendants is smaller than the threshold size then continue on the next sibling child and so on until reaching the threshold size. These child nodes then will be created as a fragment and store in a distributed node or site as illustrated in figure 6. This scenario is vulnerable to skewed query process problem if particular fragment loading is much higher than other fragments.

Angela et al. proposed a simple top-down heuristics fragmentation method called SimpleX [11]. In order to fragment document using this method, three criteria are required to determine before hand: tree-width constraint, tree-depth constraint and tree-size constraint. These criteria will restrict the size of fragment. Fragment is determined when transverse down from the root element to the leaf elements (top-down). Fragment will be decided upon sub tree size that fulfils the tree-size, tree-width and tree-depth constraint. Then, structure histogram is constructed to evaluate how efficient is the fragmentation generated.

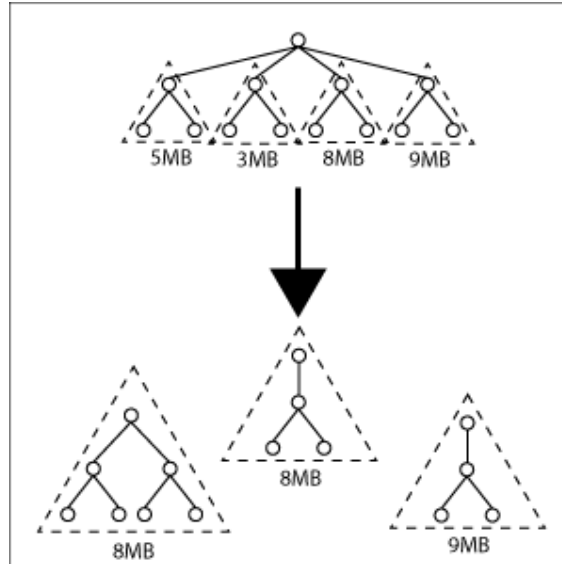


Figure 6: Fragmented by size

4.2. Query and Cost

Fragmentation of XML document can also be fragmented based on XML queries. The most common criteria to determine the fragment using this method are query frequency and the cost of query.

Leykun et al. proposed vertical fragmentation model based on queries. In their approach, two components are required to be set up for the fragmentation: Most frequently used queries with their frequencies, Element Usage Matrix (EUM) and Element Affinity Matrix (EAM) [12].

In this proposed method, it will analyze the total data access in the distributed system to determine the most frequently used queries and its frequencies. A matrix then will be constructed based on the elements access and queries. After EUM, another matrix called EAM is constructed. This matrix illustrated the relationship between elements against the queries requested. Finally, Grouping Heuristic Module is used to group elements and Splitting Heuristic Module will determine the fragment point for the fragmentation.

Ma et al., however, proposed method using heuristic to effectively fragment the XML document in horizontal fragmentation model. This method contains four steps. First of all, a horizontal fragmentation is constructed based on simple selection predicate. A query tree or query plan is build on this distribution design. From the query tree, the total execution query costs have to be determined. The query cost is the summation of storage costs and transportation costs. Storage cost is a measure of time in retrieving data from secondary storage. And, transportation cost is a measure in time for transverse time on XML document at different sites. Finally, the minimum total query cost will determine how the document should be fragmented [13, 14] Sven et al. proposed simplified cost model that work similar to previous method. The query processing cost model is based on the size estimation on the query results and query processing costs to determine the fragmentation of XML document [8].

4.3. Predicates

Predicates are commonly used in horizontal fragmentation model. There are two types of predicates: simple selection predicates and normal selection predicates.

The simple selection predicate takes the form of $path \theta v$. θ is the comparison operator which belong to the subset of $\{<, >, =, \neq, \leq, \geq, \dots\}$. $path$ is the path expression in XML and v is the value [15].

Predicate in relational database is differed from XML. In relational database, predicate indicate value of the fields. However, predicate in XML is indicated by path expression. In the previous example, predicate in relational database can be stated as $income \geq 5000$. In XML, it then express in the form of $/person/employee/income \geq 5000$.

4.4. Holes and Fillers

Holes and Fillers is a fragmentation method uses in Ad-hoc fragmentation. Ad-hoc fragmentation is fragmentation model for stream data. It does not required document schema for document fragmentation. Fragment in this model is fragmented and mark with special identifier for reconstruction later [4].

XFrag is the framework used in holes and fillers fragmentation method. In this method, the original document is break into smaller part (fillers) and one or more holes resided in filler with special tag and contains ID to corresponding filler.

5. DISCUSSION

Structure and size fragmentation method will fragment document according to the defined structure and size of XML document. The advantage of this method will distribute the content evenly across the distributed platform. However, it does not mean effectiveness in query processing response time. Skewed query processing problem is a common problem in this fragmentation method if the query processing concentrating only on particular site or distributed nodes.

The advantage of query and cost method is the most efficient method but the fragmentation cost is higher than other two methods.

Simple selection predicate is the most fundamental fragmentation method. It works fine in fragmented large XML document into smaller pieces to reduce search time and processing power on large XML document. However, it does not work efficiently compare to the query based methods.

6. CONCLUSIONS

There are pros and cons on different fragmentation models and fragmentation methods. However, heuristic is a method that can improve the query performance by study the usage of the distributed XML database. A horizontal fragmentation based on simple selection predicate method can be improved by study the query cost. According to the study to create a better fragmentation that will greatly optimize the query performance [14]. Another example of optimizing performance with its top-down heuristic is the SimpleX on structure and size fragmentation method [11].

With XML becoming the dominant standard for data exchange between various systems and databases on the Web, distributed XML is becoming crucial. In this paper, we have reviewed the types of fragmentation operations and fragmentation methods. As the result, we have also suggested the grouping for the fragmentation method.

ACKNOWLEDGEMENTS

This work is supported by funding of Fundamental Research Grant Scheme, from the Ministry of Higher Learning Education (MOHE).

REFERENCES

- [1] S.C. Haw, C.S. Lee (2011). Data storage practices and query processing in XML database: A survey. *Knowledge Based System*. 24(8). 2011, pp. 1317-1340.
- [2] M Smiljanic, H Blanken, M Keulen, W Jonker (2002). *Distributed XML Database Systems*. P1-43.
- [3] Iacob, N. (2011). Fragmentation and Data Allocation in the Distributed Environments. *Annals of the University of Craiova-Mathematics and Computer Science Series*, 38(3), 76-83.
- [4] Y.F. Huang, J.H. Chen (2001). Fragment Allocation in Distributed Database. *Journal of Information Science and Engineering* 17, pp.491-506.
- [5] V. Braganholo, M. Mattoso. (2014). A Survey on XML Fragmentation. *ACM SIGMOD Record*. 43(3).pp.24-35.
- [6] P. R. Bhuyar, A.D.Gawande, A.B.Deshmukh (2012). Horizontal Fragmentation Technique in Distributed Database. *International Journal of Scientific and Research Publication* 2(5).pp.1-6.
- [7] E. Malinowski, S. Chakravarthy (1997). Fragmentation techniques for distributing object-oriented databases.
- [8] S. Hartmann, H. Ma, K.D. Schewe (2007). Cost-based vertical fragmentation for xml. In *Advances in Web and Network Technologies, and Information Management* (pp. 12-24). Springer Berlin Heidelberg.
- [9] H. Ma, K. D. Schewe (2010). Fragmentation of XML Documents. In *Journal of Information and Data Management*, Vol. 1, No. 1, February 2010, pp. 21-33.
- [10] Sall, K. B. (2002). XML Syntax and Parsing Concepts. Retrieved from Pearson: <http://www.informit.com/articles/article.aspx?p=27006&seqNum=7>
- [11] A. Bonifati, A. Cuzzocrea, B. Zonno. (2006). Fragmenting XML Documents via Structural Constraints. *Local Proceedings of ADBIS 2006*.pp.17-29.
- [12] L. Birahnu, S. Atnafu, F. Getahun. (2010). Native XML Document Fragmentation Model. 2010 Sixth International Conference on Signal-Image Technology and Internet Based Systems, pp.233 - 240.
- [13] H. Ma, K.D. Schewe, S. Hartmann, M. Kirchberg. (2003). Distribution Design for XML documents. *Journal of Information and Data Management*, 2(1).pp. 21-33.
- [14] H. Ma, K. D. Schewe (2005). Heuristic Horizontal XML Fragmentation. In *CAiSE Short Paper Proceedings*.
- [15] H. Ma (2007). *Distribution Design for Complex Value Databases*, Ph(D) Thesis.

AUTHORS

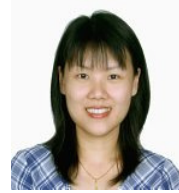
Kok-Leong Koong received his Bachelor in Computer Science and Master in Business Administration in University of Central Oklahoma, U.S.A. in 1995. He is currently lecturer of Department of Information Sciences and Computing Studies in New Era University College. His major area researches are XML Databases, E-commerce, web application and computer network.



Associate Professor Dr. Su-Cheng Haw's research interests are in XML Databases and instance storage, Query processing and optimization, Data Modeling and Design, Data Management, Data Semantic, Constraints & Dependencies, Data Warehouse, E-Commerce and Web services.



Dr. Lay-Ki Soon received her Ph.D in Engineering (Web Engineering) from Soongsil University Korea in 2009. She is currently a Senior Lecturer in Faculty of Computing and Informatics, Multimedia University. Her research interests relate to Web science, which include Web crawling, Web data mining and social network analysis. She is involved in numerous research projects funded by Malaysian government and also Japan International Cooperation Agency (JICA).



A CASE STUDY IN COMPUTER UNDERSTANDING OF PRINTED-FORMS

Davood Falahati¹, Hojat Cheraghi² and Kazem Ghalamchi³

¹Department of Electrical Engineering,
Isfahan University of Technology, Isfahan-Iran
d.falahati.1987@ieee.org

²Tehran Science and Research University, Tehran-Iran
hojat.ch@gmail.com

³Ghalamchi Foundation, Tehran, Iran.
kazemglmchi@yahoo.com

ABSTRACT

Data entry is a time consuming and erroneous procedure in its nature. In addition, validity check of submitted information is not easier than retyping it. In a mega-corporation like Kanoon Farhangi Amoozesh, there are almost no way to control the authenticity of students' educational background. By the virtue of fast computer architectures, optical character recognition, a.k.a. OCR, systems have become viable. Unfortunately, general-purpose OCR systems like Google's Tesseract are not handful because they don't have any a-priori information about what they are reading. In this paper the authors have taken a in-depth look on what has done in the field of OCR in the last 60 years. Then, a custom-made system adapted to the problem is presented which is way more accurate than general purpose OCRs. The developed system reads more than 60 digits per second. As shown in the Results section, the accuracy of the devised method is reasonable enough to be exposed in public use.

KEYWORDS

Optical character recognition, tesseract, neural networks, row finding, segmentation.

1. INTRODUCTION

Data-entry phase, is by far the most time-consuming and the deadliest line of work in data acquisition process. A remarkable portion of a company's human resource should be devoted to collect printed information in the forms. Computer-assisted data-entry process has been a human ancient dream. In the latest sixty years, there have been exerted massive efforts to implement an automatic character recognition system [1]. Template-Matching was one of the earliest methods for character recognition in which an unknown character should be compared to all of the possible candidates. Optical imaging techniques were the backbone of identification systems before 70s. Hongo and Nitta devised an optical system that processed a video signal using template matching [2].

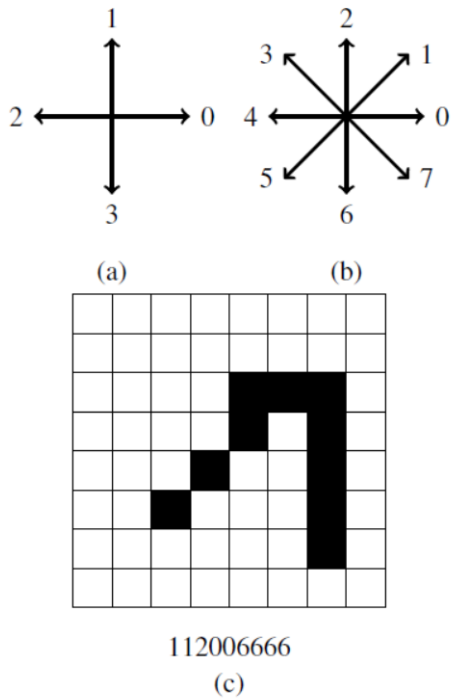


Fig. 1. Chain codes grid. (a) 4-connected grid (b) 8-connected grid. (c) a sample of coded sequence using 8-connected grid.

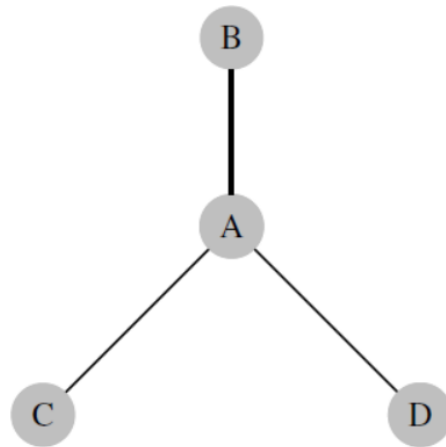


Fig. 2. Basic graphs of LAG method. Node A is a junction and nodes B,C and D are paths.

Digital computers deviated the way of template matching from optical to logical ones. Peephole method was one of the earliest logical approaches to digital template matching. Template matching technique evolved and offered using moments [1] and Fourier series later [3]. The former method made an invaluable translation-rotation-scale invariant a.k.a TRS, a tool for template matching [4]. Another set of commonplace TRS moments in the vicinity of pattern recognition are Zernike [5] and Fourier-Mellin moments that implemented in character recognition as well [6, 7].

Sakoe offered using dynamic programming (DP) technique to template matching problem [8]. Dynamic time warping (DTW) used to find the minimum distance between a given template and

its corresponding candidate [8, 9]. DP-based template matching techniques make the comparison of two non-equal length vectors possible. Therefore, character matching would be possible while aspect ratio remains intact through DP.

Template matching practice, however, remained limited to printed character recognition systems [10]. To cover handwritten characters, another useful recognition method called structural analysis, introduced [1]. Freeman introduced a novel method on encoding curvatures in [11]. The proposed method, currently called "chain codes", was proposed in order to be used in image compression however, later it found practice in character recognition efforts [10]. Chain codes were developed in encoding efforts and a more sophisticated one is presented in [12]. Chain codes are founded on quantization process and decode a curvature to a line with a certain slope. As depicted in figure 1, 4-connected and 8-connected grids are two types of commonplace chain codes being used. The higher slopes in a grid, the lower quantization error in the coded curvature. Pavlidis at Bell laboratories proposed a thinning method suited for multi-font document recognition system [13]. The before said method uses line adjacency graphs (LAGs) which tries to stack semi-linear group of segments. Pavlidis addresses two types of graph nodes as junction and paths as shown in figure 2.

Structure analysis method alleviated template matching defects in handwritten character recognition systems. The candidate with the lowest distance with the template considers to be the best match. Since characters are not connected, registration problem, however, is straightforward to deal with.

In a wider sense, trainable scoring techniques are amazingly suited to human activities like speech and handwriting recognition [14]. Hidden markov models a.k.a HMMs have long been used in speech recognition systems. Due to probabilistic features of human writing systems, HMM has also been used to serve character recognition purposes [15]. Neural networks are another useful matching asset in character recognition systems [16,17,18,19]. Neural networks are trainable and their usage is less complicated as HMMs while multi layer perceptrons (MLPs) need more training data compared with HMMs [20]. Despite HMMs' more accurate performance in [20], MLPs show a better performance in speech recognition activities [21].

Right-to-left, cursive and connected scripts, however, are the most challenging scripts in recognition systems [22]. Arabic and Farsi scripts are two well-known cursive scripts widely using. Reading these scripts requires an excessive segmentation process, say, word segmentation [23, 24]. Technically speaking, a Farsi and/or Arabic word consists of connected characters (see figure 3). Therefore, a vital step in reading Farsi/Arabic characters is to rightly segment all of the characters.



Fig. 3. Two samples of Farsi word "Sample". This word consists of 5 characters N-M-U-N-H. It is noteworthy that unlike Arabic, vowels are not being written in formal Farsi handwriting.

This paper is organized as follows. In the section 2 the proposed framework is shortly introduced. In section 3, the preprocessing methods are discussed. Section 4 is devoted to segmentation procedures. Section 5 pays attention to training process and scoring. Finally, section 6 reveals the results of our proposed method.

2. DISCUSSING THE PROBLEM

Reading and checking printed score sheets for a massive number students is a cumbersome task which is not achievable by human resources. Kanoon Farhangi Amoozesh is a test-conducting foundation in Iran holds weekly exams among 400,000 students. In order to analyze students' educational background, one needs to certify the student claimed GPA according to printed documents. An Iranian high-school score-sheet issued by ministry of education is shown in figure 4.

The image shows a standard Iranian high-school score-sheet. It features a table with multiple columns for subjects and scores. The table is filled with numerical data. There is a blue circular stamp on the left side and a logo in the top left corner. The text is in Persian.

Fig. 4. A standard Iranian high-school score-sheet

3. PREPROCESSING

The devised system works along with a web-server in which server sends uploaded score-sheets to character recognition system and its output is fed back to the web-server (see figure 5).

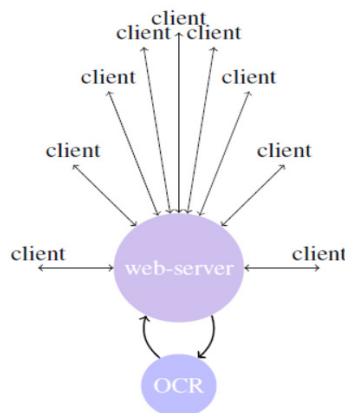


Fig. 5. Topology of the devised system

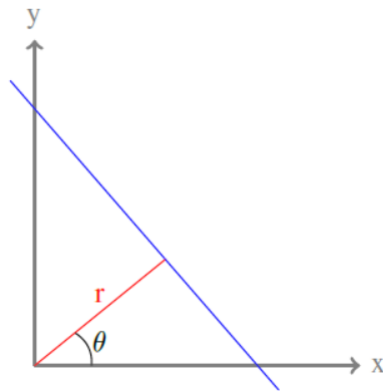


Fig. 6. Representation of a line in polar coordinates

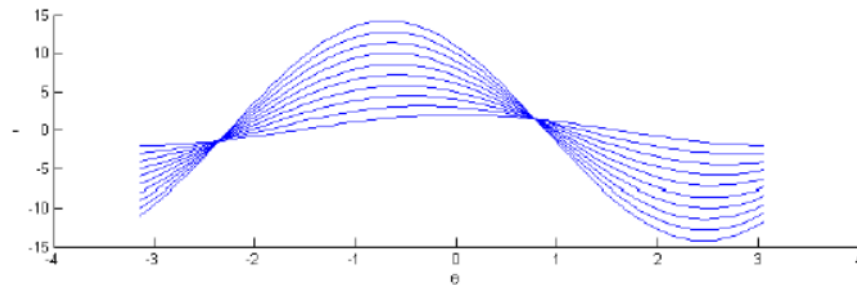


Fig.7. Sinusoid curves intersect on points located on a line. Curves are intersecting in two distinct points. It means that all of the points are located on a line.

A major problem with this case is image acquisition phase. As long as images are taken by clients, there is no control over the scanning procedure. The input images to the character recognition system are prone to rotation, scale manipulation and translation. Therefore, a preprocessing step is strictly required.

The most annoying scanning-mismatch is rotation. To remove rotation effects on reading characters, the perpendicular lines of the table are used. Hough line transform is used to extract lines and their angles [25]. As shown in figure [6], the Hough line transform first expresses a line in the polar system as below.

$$r = x \cos(\theta) + y \sin(\theta) \quad (1)$$

Each point $P=(x_0, y_0)$ identifies a Sinusoidal curve. Points located on a straight line result in Sinusoidal curves that intersect in polar coordinates (see figure 7). In a real case, there are many points and many Sinusoidal curves in result. The points with the majority of intersections identify straight lines. In this venue one looks for the longest lines and computes their angle. After that, the input image will be rotated in inverse direction. Figure 8 depicts an instance of recovered lines using Hough method.



Fig. 8. A sample of rotation correction by Hough line transform. The rotation is removed by computing the rotation angle and inversely applying it to the image. In this method the horizontal lines are removed by filtering.

4. SEGMENTATION

A top-to-bottom raster scanning is required to find the position of lines whole the score-sheet. Before that, a canny edge detector should be applied to the image to remove redundant image data. In the case of characters, canny preserves the contours of the characters. Moreover, line height is obtainable in this way. Extracted lines are depicted in figure 9. This method is scale, translation and rotation invariant since it just attempts to find pixels concentration regions.

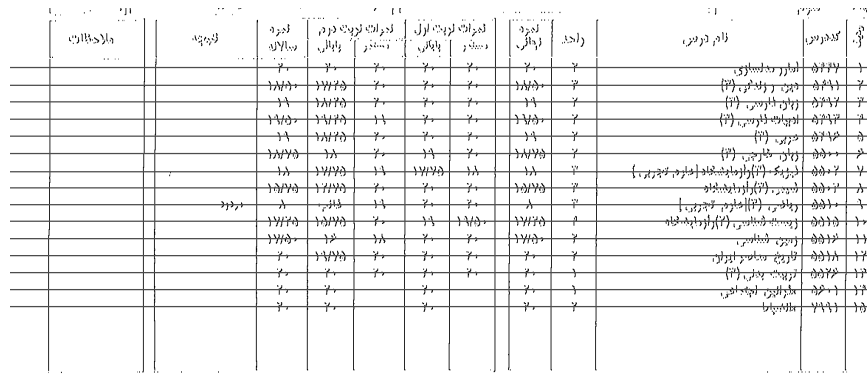


Fig. 9. Finding the rows of the score-sheets. The centers of the rows are emphasized by solid horizontal lines. The image is cropped to safeguard student's personal information.

In a similar manner, the table columns can be resolved easily. In the case the rows are not detectable, the input image would be rejected.

The last step to be paved is character segmentation. The output of row detection is fed to character parser. Character parser extracts a block containing the character. Fortunately, numerical digits are not connected in Farsi handwriting system. However, the character "point" or

"," is troublesome in Farsi. The mentioned character prints like "slash" or "/" in both Farsi and Arabic writing systems. The mentioned character sometimes connects two numerical digits specifically when scanning resolution is not fair enough. Figure 10 shows the output of character parser. To deal with the problem of parsing connected characters like figure 10.b, the authors proposed the average character width method in which the width of characters obtains for every row and the blocks wider than twice the width of average width considered as connected blocks. Average width method is life-savior in the case of ink spattering as well.

5. SCORING

The proposed multi-font character recognition system should be flexible about font change. Moreover, it should cover handwritten numerical digits. To cover this wide range of formation changes, artificial neural networks have been utilized [26]. In the proposed method multilayer perceptrons are used and back-propagation (BP) training method adjusts the weights of the perceptrons. Figure 11 illustrates an example of neural network with 4 hidden layers. Hidden layers are not accessible and their weight cannot be easily changed. Back-propagation method tries to minimize the distance between the real output, y_k , and the desired output, d_k . This distance can be formulated as below:

$$\epsilon = \frac{1}{2} \sum_{k=1}^N (d_k - y_k)^2 \quad (2)$$

Where N is the number of neurons. The effect of neurons' weights on the output error is represented by gradient of ϵ_k .

$$\Delta \epsilon_k = \frac{\partial \epsilon}{\partial w_{kj}} \quad (3)$$

Using the steepest descent gradient algorithm [27] it leads to:

$$w_{kj}(m+1) = w_{kj}(m) + \Delta w_{kj}(m) \quad (4)$$

and w_{kj} is as below:

$$\Delta w_{kj} = -\eta \frac{\partial \epsilon}{\partial w_{kj}} \quad (5)$$

η is coined as learning rate. The output of each perceptron, y_k , is the weighted sum of previous layer perceptrons. In other words:

$$z_k = \sum_j w_{kj} x_j \quad (6)$$

z_k applies to sigmoid function :

$$F_N(x) = (1 + e^{-x})^{-1}$$

Wherein:

$$y_k = F_N(z_k) \quad (7)$$

Exploiting the sigmoid function results in:

$$\frac{\partial \epsilon}{\partial w_{kj}} = \frac{\partial \epsilon}{\partial z_k} \frac{\partial z_k}{\partial w_{kj}} \quad (8)$$

Then using (6) and the fact that $y_j(p-1) = x_j(p)$ leads to:

$$\frac{\partial \epsilon}{\partial w_{kj}} = x_j(p) = y_j(p-1) \quad (9)$$

where p is the output layer. Let define a new parameter ϕ as follows:

$$\Phi_k(p) = -\frac{\partial \epsilon}{\partial z_k(p)} \quad (10)$$

Using this new parameter and obtain:

$$\frac{\partial \epsilon}{\partial w_{kj}} = -\Phi_k(p)x_j(p) = -\Phi_k y_j(p-1) \quad (11)$$

Doing some math and to get:

$$\Delta w_{kj} = \eta \phi_k(p) x_j(p) = \eta \Phi_k(p) y_j(p-1) \quad (12)$$

Invoking the chain rule in equation (10) results in:

$$\Phi_k = -\frac{\partial \epsilon}{\partial z_k} = -\frac{\partial \epsilon}{\partial y_k} \frac{\partial y_k}{\partial z_k} \quad (13)$$

The latter equation is used for the output layer. Differentiating from (2) results in:

$$\frac{\partial \epsilon}{\partial y_k} = -(d_k - y_k) = y_k - d_k \quad (14)$$

Exploiting the sigmoid property which is

$$y_k = F_N(z_k) = y_k(1 - y_k) \quad (15)$$

Therefore

$$\Phi_k = y_k(1 - y_k)(d_k - y_k) \quad (16)$$

Invoking (5) and (6) to obtain Δw_{kj} as below:

$$\Delta w_{kj}(p) = \eta \Phi_k(p) y_j(p-1) \quad (17)$$

The Δw_{ji} for the hidden layers are being computed as follows.

$$\Delta w_{ji} = -\eta \frac{\partial \epsilon}{\partial z_j} y_j(r-1) = \eta \Phi_j(r) y_i(r-1) \quad (18)$$

In order to update ϕ_j for the hidden layers, the following equation is being used.

$$\begin{aligned} \Phi_j(r) &= \frac{\partial y_j}{\partial z_j} \sum_k \Phi_k(r+1) w_{kj}(r+1) = \\ & y_j(r)[1 - y_j(r)] \sum_k \Phi_k(r+1) w_{kj}(r+1) \end{aligned} \quad (19)$$

Back-propagation method converges more quickly than Adaline and Madaline methods [26].

In order to train the neural network, the segmented digit obtained from the segmentation process feeds to neural network as a 20x20 binary image. The training set consists of 42 classes without rejection class. Each class consists of 300 samples of 300 Farsi/Arabic fonts.

6. RESULTS

The proposed character recognition approach is implemented using C language along with openCV library. It is able to read score-sheets and report the results in both plain text and HTML format. The utilization is as simple as follows:

```
$ ocr -i < path to input file > [-h]
```

Wherein "-h" comes if the HTML report is needed. The system reads every 1200x800 pixel score-sheet in less than 2 seconds on a core i5 2.4GHz personal computer. The devised system brings a high degree of precision. In average, the presented system correctly reads every numerical characters in 0.015 seconds with an error rate less than 2%. The accuracy of the Google's Tesseract is less than 40% for similar tests.

ACKNOWLEDGEMENTS

Writing and editing this paper was not possible without invaluable help of Omid Sefat, Ali Golestan and Farhad Baybordi in Kanoon Farhangi Amoozesh. Moreover, the obstacles we jumped over were shortened by Kazem Ghalamchi.

REFERENCES

- [1] S.Mori, C.Suen, and K. Yamamoto, "Historical review of ocr research and development," Proceedings of the IEEE, vol. 80, pp. 1029–1058, Jul 1992.
- [2] Y.Hongo and Y.Nitta, "Pattern recognition apparatus," Dec. 9 1986. US Patent 4,628,533.
- [3] C.T.Zahn and R.Z. Roskies, "Fourier descriptors for plane closed curves," Computers, IEEE Transactions on, vol. C-21, pp. 269–281, March 1972.
- [4] M.-K. Hu, "Visual pattern recognition by moment invariants," Information Theory, IRE Transactions on, vol. 8, pp. 179–187, February 1962.
- [5] D.Xiao and L.Yang, "Gait recognition using zernike moments and bp neural network," in Networking, Sensing and Control, 2008. ICNSC 2008. IEEE International Conference on, pp. 418–423, April 2008.
- [6] C.Kan and M.D.Srinath, "Invariant character recognition with zernike and orthogonal fouriermellin moments," Pattern Recognition, vol. 35, no. 1, pp. 143 – 154, 2002. Shape representation and similarity for image databases.
- [7] L.Torres-Mendez, J.Ruiz-Suarez, L.Sucar, and G.Gomez, "Translation, rotation, and scale-invariant object recognition," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 30, pp. 125–130, Feb 2000.
- [8] H.Sakoe and S.Chiba, "Dynamic programming algorithm optimization for spoken word recognition," Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 26, pp. 43–49, Feb 1978.
- [9] D.Falahati, M.Helforoush, H.Danyali, and M.Rashidpour, "Static signature verification for farsi and arabic signatures using dynamic time warping," in Electrical Engineering (ICEE), 2011 19th Iranian Conference on, pp. 1–1, May 2011.
- [10] A.Sinha, "An improved recognition module for the identification of handwritten digits," 1999.
- [11] H.Freeman, "On the encoding of arbitrary geometric configurations," Electronic Computers, IRE Transactions on, vol. EC-10, pp. 260–268, June 1961.
- [12] G.Schuster and A.Katsaggelos, "An optimal polygonal boundary encoding scheme in the rate distortion sense," Image Processing, IEEE Transactions on, vol. 7, pp. 13–26, Jan 1998.
- [13] T.Pavlidis, "A vectorizer and feature extractor for document recognition," Computer. Vision Graph. Image Process., vol. 35, pp. 111–127, July 1986.
- [14] L.Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," Proceedings of the IEEE, vol. 77, pp. 257–286, Feb 1989.
- [15] O.E. Agazzi and S. shiaw Kuo, "Hidden markov model based optical character recognition in the presence of deterministic transformations," Pattern Recognition, vol. 26, no. 12, pp. 1813 – 1826, 1993.

- [16] E.Alpaydin, "Optical character recognition using artificial neural networks," in Artificial Neural Networks, 1989., First IEE International Conference on (Conf. Publ. No. 313), pp. 191–195, Oct 1989.
- [17] R.Arnold and P.Miklos, "Character recognition using neural networks," in Computational Intelligence and Informatics (CINTI), 2010 11th International Symposium on, pp. 311–314, Nov 2010.
- [18] F.Yang and F.Yang, "Character recognition using parallel bp neural network," in Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on, pp. 1595–1599, July 2008.
- [19] A.Gupta, M.Srivastava, and C.Mahanta, "Offline handwritten character recognition using neural network," in Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on, pp. 102–107, Dec 2011.
- [20] E.Hatzipantelis, A.Murray, and J.Penman, "Comparing hidden markov models with artificial neural network architectures for condition monitoring applications," in Artificial Neural Networks, 1995., Fourth International Conference on, pp. 369–374, Jun 1995.
- [21] A.Waibel, T.Hanazawa, G.Hinton, K.Shikano, and K. Lang, "Phoneme recognition: neural networks vs. hidden markov models vs. hidden markov models," in Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on, pp. 107–110 vol.1, Apr 1988.
- [22] A.Cheung, M.Bennamoun, and N. Bergmann, "A recognition-based arabic optical character recognition system," in Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on, vol. 5, pp. 4189–4194 vol.5, Oct 1998.
- [23] V.Margner and M.Pechwitz, "Synthetic data for arabic ocr system development," in Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on, pp. 1159–1163, 2001.
- [24] R.Prasad, S.Saleem, M.Kamali, R.Meermeier, and P.Natarajan, "Improvements in hidden markov model based arabic ocr," in Pattern Recognition, 2008. ICPR 2008. 19th International Conference on, pp. 1–4, Dec 2008.
- [25] G.Stockman and L.G.Shapiro, Computer Vision. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1st ed., 2001.
- [26] D.Graupe, Principles of Artificial Neural Networks. River Edge, NJ, USA: World Scientific Publishing Co., Inc., 1997.
- [27] Z.Tian-liang, "Solving non-linear equation based on steepest descent method," in Information and Computing (ICIC).

UNSUPERVISED REGION OF INTEREST DETECTION USING FAST AND SURF

Abass A. Olaode¹, Golshah Naghdy¹ and Catherine A. Todd²

¹School of Electrical Computer Telecommunication Engineering,
University of Wollongong, Wollongong, Australia
Abass.Olaode808@uowmail.edu.au
golshah@uow.edu.au

²Faculty of Computer Science and Engineering,
University of Wollongong, Dubai, UAE
CatherineTodd@uowdubai.edu.au.

ABSTRACT

The determination of Region-of-Interest has been recognised as an important means by which unimportant image content can be identified and excluded during image compression or image modelling, however existing Region-of-Interest detection methods are computationally expensive thus are mostly unsuitable for managing large number of images and the compression of images especially for real-time video applications. This paper therefore proposes an unsupervised algorithm that takes advantage of the high computation speed being offered by Speeded-Up Robust Features (SURF) and Features from Accelerated Segment Test (FAST) to achieve fast and efficient Region-of-Interest detection.

KEYWORDS

Region of Interest, Image segmentation, SURF, FAST, Texture description, PLSA, BOV, K-means clustering, unsupervised image classification.

1. INTRODUCTION

Image modelling has been recognised as an essential step towards recognition [1], thus an important components in image retrieval. Many image retrieval implementations adopt global features such as colour histograms in describing image contents [2]. Although this approach of covering the entire image space has proven to be successful for images with distinct colours, Tuytelaars and Mikolajczyk [2] noted that such approach cannot distinguish between foreground and background, and it is severely challenged by image clutter and occlusions [2].

Tuytelaars and Mikolajczyk [2] explained that an approach to tackling the challenges of global image features is to segment the image into a limited number of regions or segments, where each region corresponding to a single object or part of an object. Common methods of achieving this involves exhaustively sampling different subparts of the image at each location and scale. Such approach has the tendency to become computational expensive and inefficient [2, 3]. An efficient alternative is to determine the most important region of the image, commonly regarded as Region of Interest (ROI) [4].

The identification of an image's ROI using supervised learning is often challenged by the need for prior information regarding patterns within the image collection, thus making unsupervised learning a more attractive option [4, 5]. This paper presents a novel ROI detection approach that uses fast feature detection algorithms (Speeded-Up Robust Features (SURF) and Features from Accelerated Segment Test (FAST)) to identify likely regions of interest, and then compares the texture of these regions to complete the ROI detection.

The remainder of this paper is structured as follows: Section 2 provides background information SURF detector and FAST detector. Section 3 gives a brief review of recent works on ROI detection where related approaches have been applied, while Section 4 provides a detailed description of the implementation of the proposed SURF and FAST combination in the detection of ROI. Section 5 discusses the experimentations carried out in the evaluation of the effect of the proposed ROI detection on unsupervised classification using PLSA, and Section 6 highlights the direction of future works aimed at the use of ROI in semantic labelling of images.

2. IMAGE FEATURE DETECTORS

Feature extraction algorithms use digital image processing techniques to extract low level features from the high dimensional matrix representation of images. For reliable image recognition, it is important that the features extracted from images be detectable even under changes in image scale, noise and illumination. To satisfy this need, keypoints corresponding to high-contrast locations such as object edges and corners are often sought [6, 7]. Although traditional image feature extraction algorithms consist of feature detection and feature description components, this section focuses mainly on the detection of image features.

The most popular image feature extraction algorithm is the Shift Invariant Feature Transform (SIFT). SIFT uses the Difference of Gaussian (DoG) algorithm to detect image features, and has proven to be very successful in computer vision applications due to its resistance to common image transformations [6, 7]. However, the computational requirement of SIFT is very high [6, 7], which has made algorithms such as SURF and FAST preferred choice for real-time applications [6, 7].

2.1 SURF

Like SIFT, SURF features extraction algorithms can be regarded as sparse feature extraction algorithms because they only detect and describe features at keypoint locations. Rather than using DoG and image pyramid for the detection of keypoints as in SIFT, SURF uses the Hessian matrix in which the convolution of Gaussian second order partial derivatives with a desired image are replaced with box filters applied over image integrals (sum of grayscale pixel values) [9]. Given a point $x = (x, y)$ in an image I , the Hessian matrix $H(x, \sigma)$ in x at scale σ is defined as follows (Equation 1):

$$H(x, \sigma) = \begin{pmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{pmatrix} \quad (1) [9]$$

Where $L_{xx}(x, \sigma)$ is the convolution of the Gaussian second order partial derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point x , and similarly for $L_{xy}(x, \sigma)$ and $L_{yy}(x, \sigma)$ [9]. The use of integral image representation at the keypoint detection stage of SURF ensures that the computational cost of applying box filter is independent of the size of the filter. This allows SURF to achieve much faster keypoint detection than SIFT by keeping the image size the same while varying only the

filter size [9]. Figure 1A illustrates the SURF keypoints detected on a sample image from Caltech-101 dataset.

Although, SURF's performance is mostly similar to SIFT, it is unstable to rotation and illumination changes [10]. Liu et al. [11] noted that although SURF is capable of representing most image patterns, it is not equipped to handle more complicated ones. However, Khan et al [9], implemented classification experiments on images from David Nister, Indoor, Hogween and Caltech datasets to yield results that confirms that SURF's performance is as good as that of SIFT, with both recording 97% accuracies on Caltech dataset. Therefore, this study considers SURF adequate enough to be considered for the purpose of detecting image features in the determination of ROI.

2.2 FAST

Corners are found at various types of junctions, on highly textured surfaces, and occlusion boundaries. With the aim of identifying a set of stable and repeatable features, they are typically detected using corner detectors such as the Harris corner detector, Smallest Univalued Segment Assimilating Nucleus (SUSAN) detector, and FAST detector.

The Harris detector was identified as the most stable one in many independent evaluations. Although SUSAN is more efficient than Harris detector, it is also more sensitive to noise. FAST is an improvement over the SUSAN detector with higher accuracy [2]. It fell just behind the Harris detector, but significantly faster than any other algorithm [13] making it the most appropriate for real time machine vision applications and for image retrieval purposes.

The FAST considers corners more intuitive than edges because they show a stronger two dimensional intensity change, and are therefore well distinguished from the neighbouring points [13]. FAST uses a corner response function (CRF) that gives a numerical value for the corner strength based on the image intensity in the local neighbourhood. This CRF was computed over the image and corners which were treated as local maxima of the CRF. Along with this, FAST also employs a multi-grid technique to improve the computational speed and suppress detected false corners being detected. Figure 1 demonstrates the corners points detected using the FAST algorithm.

Undoubtedly, the main contribution of FAST was the increment of the computational speed required in the detection of corners [2]. SURF and FAST have been considered two of feature detection algorithms most suitable for real-time applications due to their speeds.

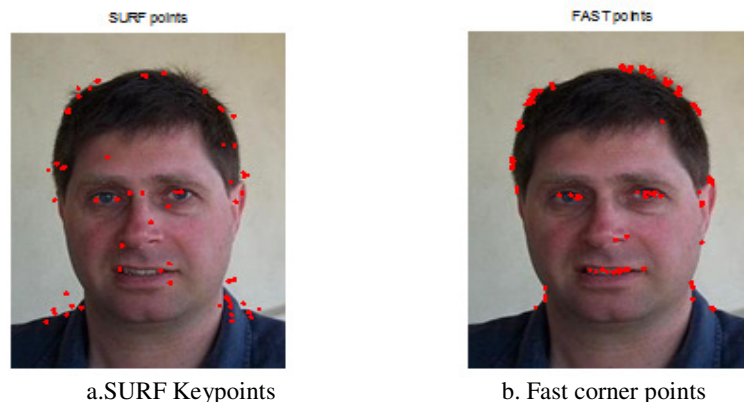


Figure 1. Feature detection on a sample image

3. RELATED WORKS

Existing unsupervised ROI determination algorithms often require extensive and computational search within an image space before the identification of the desired region [4]. In the unsupervised image categorisation framework proposed by Huang et al. [4], the authors presented an unsupervised ROI detection approach that computes dense-SURF over the unlabelled image [4]. The high computation requirement of dense-features is further increased by the comparison of the images within the image set so as to identify common feature when the process is built on unsupervised learning [4].

The unsupervised ROI determination proposed by Huang et al. [4] incurs heavy computation, and is not suitable when managing large number of images. This study considers computational speed an important requirement in image retrieval especially when handling large number of images (1000 images and above), and proposes a ROI determination algorithm that takes advantage of the high computational speed offered by SURF and FAST keypoint detection algorithms.

Although the use of keypoints in the detection of ROI has been investigated by Kapsalas et al [3], who used Harris corner detector in identifying objects present in an image, the approach proposed in this paper differs, in that it attempts to achieve effective labelling through the identification of the important object within the image. It also differs from the ROI detection approach proposed by Huang et al [4] since it does not compare the image being processed with other images in the set during the ROI detection, thus reducing the computational requirement.

4. THE PROPOSED UNSUPERVISED FRAMEWORK

The identification of interest points present within the space of an image is important in the determination of the image's ROI [3], therefore the method being proposed in this paper maximises the number of the number of interest points detected within a sample image through the use of the combination of FAST corner detector and SURF detector as shown in Figure 2A. The use of several complementary feature detectors in such manner ensures good coverage of the image surface [2].

If FAST corner points and SURF keypoints are respectively represented by the sets $F = \{f_1, f_2, f_3 \dots \dots f_L\}$ and $S = \{s_1, s_2, s_3 \dots \dots s_L\}$, then the combined FAST and SURF feature points can be represented by the set P ; where $P = F \cup S = P = \{p_1, p_2, p_3 \dots \dots p_L\}$. The two key criteria which distinguish keypoints belonging to an ROI from those that do not belong to the desired region are location and description.

The algorithm proposed uses the coordinates provided by the SURF and FAST algorithm to satisfy its need for location information. It recognises texture as the most appropriate image characteristics by which regions within an image can be distinguished from one another, and employs Law's filter [13, 14] in developing a 9 dimensional descriptor for a rectangular mask centred at the coordinates of the location of the point.

The dimension of the mask use is made to be $0.33 * (\text{height} * \text{Width})$, thus responding to image size while capturing similarities between neighbouring keypoints. The choice of 9 dimensional texture descriptor ensures the avoidance of the heavy computations associated with the popular descriptors, thereby reducing the computation overhead. The proposed algorithm relies on K-Nearest Neighbour categorisation (KNN) for the categorisation of the texture descriptions of each keypoint into either foreground or background, therefore it requires training samples.

From a reference point (\bar{x}, \bar{y}) established to be the mean of all the x and y coordinates, the horizontal and vertical distances of each point are calculated. The pair of distances for all the keypoints are placed in the sets $X = \{x_1, x_2, x_3 \dots x_l\}$ which has a mean of \bar{X} , and $Y = \{y_1, y_2, y_3 \dots y_l\}$ with a mean of \bar{Y} represents the means of the sets. A keypoint (x_i, y_i) is chosen to be a likely foreground training sample candidate if it satisfies the conditions of Equation 2.

$$|x_i - \bar{x}| < \bar{X} \quad (2a)$$

$$|y_i - \bar{y}| < \bar{Y} \quad (2b)$$

Where I_x and I_y represents the image dimensions, any keypoints that do not satisfy the above criteria is considered to be a background training sample if does not satisfy at any of Equation 3a or Equation 3b.

$$|x_i - \bar{x}| < \frac{2}{5} * (I_x) \quad (3a)$$

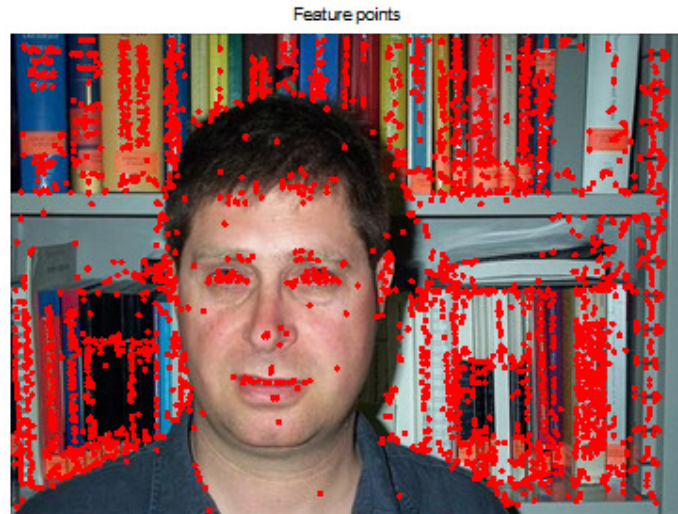
$$|y_i - \bar{y}| < \frac{2}{5} * (I_y) \quad (3b)$$

Keypoints that does not satisfy both of Equation 2, but satisfies one of Equation 3 are categorised based on their texture descriptors using KNN. Furthermore, the texture description of the “likely” foreground training samples are compared with those of the background training samples so as to achieve a set of training samples that is exclusive to the foreground. Assuming $R = \{r_1, r_2, r \dots r_L\}$ is a set of texture descriptors for points that satisfied Equation 2, and $B = \{b_1, b_2, b_3 \dots b_L\}$ is the set of texture descriptors for points that satisfied at least one of Equation 3, then a sample is confirmed to belong to the foreground training set if it satisfies the Equation 4.

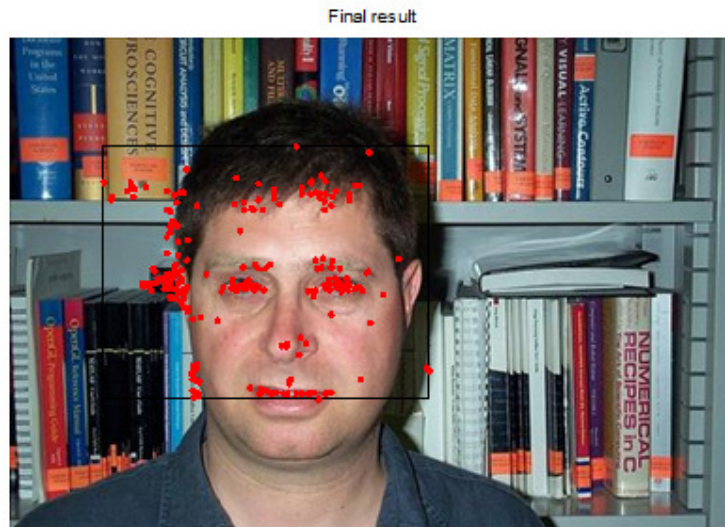
$$\frac{\sum_{j=1}^L |r_i - r_j|}{n(R)} < \frac{\sum_{j=1}^L |r_i - b_j|}{n(B)} \quad (4)$$

Equation 4 indicates that a legitimate foreground training candidate sample should record an average Euclidean distance from every other point within the “likely” foreground training sample group which is less that the average Euclidean distance recorded from the point to every point in the background training samples.

Preliminary experiments conducted as part of this study shows that although Equation 4 holds in most case, the reverse can also be the case, thus making the Equation a means of separating the “likely” foreground samples into two groups. In such case, the group which is least similar to the background training samples in term of texture description is then chosen to be the foreground training samples. In the implementation of the KNN, each feature point to be categorised is allocated the highest occurring label from the closest 5 neighbours, thus the points labelled as the foreground are grouped together to form the desired Region. In this way, the points located within the region of interest are effectively identified as shown in Figure 2B. The ROIs detected on more sample images from Caltech-101 are displayed in Figure 3.



a. The combined FAST and SURF feature points on a sample image



b. An illustration of the result of using the proposed algorithm on a sample image

Figure 2. Illustration of a.) The keypoints identified using SURF and FAST, and b.) The Region of Interest points identified using the proposed ROI detection algorithm





Figure 3. Results of application of the proposed ROI detection framework on sample images

5. EXPERIMENTS AND RESULTS

As discussed in Section I, the determination of ROI during unsupervised image categorisation is important because it ensures that most of the attention is paid to the images' foreground, thereby limiting the effect of images' background on the classification accuracy. This section examines the effect of this algorithm on the completely unsupervised combination of PLSA and K-means.

The experiments in this study used the 3 image datasets constituted from 12 Caltech-101 in [15]. While the number of images is fixed at 500, the categories are varied (4, 8 and 12 categories). These classes are: Airplanes, Motorbikes, Face, Watch, Car, Backpack (Caltech-256), Ketch, Bonsai, Butterfly, Crab, Revolver, and Sunflower. In all the experiments in this section, the Histogram of Oriented Gradients (HOG) [16] feature extraction is chosen as the image feature extraction algorithm [11].

The proposed ROI detection algorithm is implemented on the image collections and the detected ROI images are converted to the various forms PLSA models, and then quantised into semantic groups using the k-means algorithm. The PLSA/K-means classification is implemented with 25 latent topics and 25 clusters, thus allowing a one to one mapping between each of latent topics and each of the semantic groups identified during K-means clustering. In evaluating the quality of the centroids presented at the completion of each categorisation process, each of the centroids is visualised and labelled, and the label appointed to the centroid is automatically applied to all the images in the centroid's cluster. With all the available images labelled, accuracy of the unsupervised categorisation is determined through a comparison between the new labelling and the ground truth. By varying the visual codebook size under the 3 chosen image collections, this section determines the appropriate visual codebook sizes for the Bag-of-visual word modelling which precedes the PLSA classification. The graphical demonstration of their average performances over 5 runs is shown in Figure 4.

Using the data presented graphically in Figure 4, this paper identifies that the most appropriate visual codebook sizes for the implementation in the categorisation of 4, 8 and 12 categories image collections are 200, 500, and 900. Table 1 provides a comparison of the performance of PLSA classification under two scenarios; 1) without using ROI, and 2) using ROI.

An overall assessment of Table 1 reveals a general increment in categorisation accuracy when ROI is included during PLSA/K-means categorisation. This increment can be attributed to the ability of the proposed ROI algorithm to minimise the amount of image background included during image modelling.

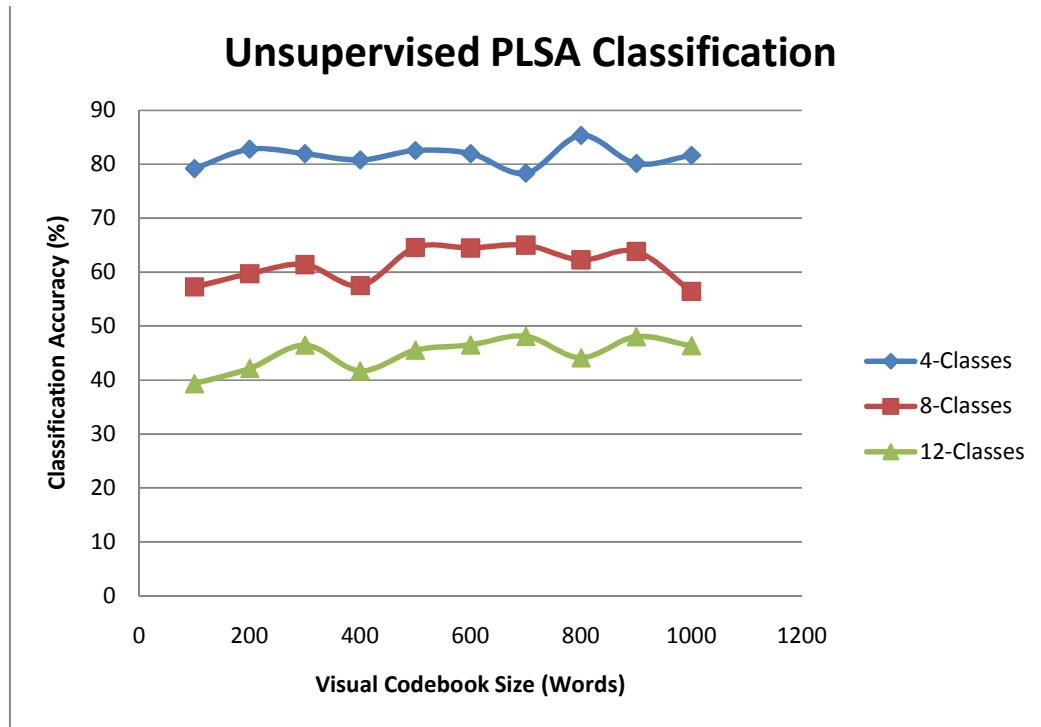


Figure 4. The unsupervised PLSA categorisations implemented with the inclusion of Region of Interest determination

Table 1. A comparison of the effects of ROI during Unsupervised image categorisation via PLSA

Number of categories	Visual Codebook sizes	Accuracies	
		Without using ROI detection	Using ROI detection
4	200	80.80%	82.77%
8	500	58.07%	64.56%
12	900	45.55%	48.07%

6. FUTURE WORKS

This paper has successfully demonstrated the use of FAST and SURF to be useful tools in the determination of an image's ROI, and the possibility of improving the accuracy of an image classification by limiting an image's modelling to the ROI of the image. However, it is important to note that the categorisation accuracies recorded with the use of ROI is lower than the accuracies obtainable under a completely supervised implementation of PLSA categorisation. The recorded performance can be further improved through the combination of the proposed ROI determination algorithm with the spatial pyramid algorithm proposed by Lazebnik et al. [17] or with semantic segmentation. These combinations will be investigated in future works.

To further improve the categorisation accuracy, there is the need to identify a more effective feature extraction algorithm especially that is able to accommodate the diverse nature of the nature dataset. A possible solution to this challenge is the combination of the HOG descriptor with another image feature extraction algorithm (such as shape or corner description algorithms). This will also be examined in future works.

7. CONCLUSION

For minimising the effect of image backgrounds on classification accuracies, this paper proposes the use determination of the ROI of each using SURF and FAST, and demonstrated the ability of the proposed algorithm to limit image modelling to relevant region within the image.

Using 3 image collections constituted from Caltech-101, this paper successfully demonstrates the effectiveness of the categorisation model in improving the unsupervised PLSA categorisation, and identified the inclusion of spatial pyramid and semantic segmentation alongside ROI determination as two approaches that may be employed in the search for higher accuracy during unsupervised PLSA categorisation.

ACKNOWLEDGEMENT

The authors of this work wish to thank the research students of their University for their support.

REFERENCES

- [1] C.M.Bishop and J.M.Winn, "Non-linear Bayesian Image Modelling," in 6th European Conference on Computer Vision, ECCV 2000, Antibes, 2000.
- [2] T.Tuytelaars and K.Mikolajczyk, "Local Invariant Feature Detectors: A Survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 3, p. 177–280, 2008.
- [3] P.apsalas, K.Rapantzikos, A.Sofou and Y.Avrithis, "Regions of interest for accurate object detection," in *International Workshop on Content-Based Multimedia Indexing, CBMI.*, London, 2008.
- [4] Y. Huang, Q.Liu, F.Lv, Y.Gong and D.Metaxas, "Unsupervised Image Categorization by Hypergraph Partition," *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 33, no. 6, June 2011.
- [5] D.-C.Lee and T.Schenk, "Image Segmentation From Texture Measurement," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. XXIX, no. 3, pp. 195-199, 1992.
- [6] M.Guerrero, "A Comparative Study of Three Image Matcing Algorithms: SIFT, SURF, and FAST," Utah state University, Utah, 2011.
- [7] R.Sukthankar and Y. Ke, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004, Pittsburg, 2004.*
- [8] E.Rublee, V.Rabaud, K. Konolige and G.Bradski, "ORB: an efficient alternative to SIFT or SURF," *IEEE International Conference on Computer Vision (ICCV)*,, Barcelona, 2011.

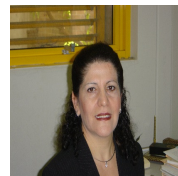
- [9] H.Bay, T.Tuytelaars and L. V. Gool, "SURF: Speeded Up Robust Features," in Computer Vision-ECCV , Zurich, Springer Berlin Heidelberg, 2006, pp. 404-417.
- [10] N.Khan, B.McCane and G. Wyvill, "SIFT and SURF Performance Evaluation against Various Image Deformations on Benchmark Dataset," in International Conference on Digital Image Computing: Techniques and Applications, Noosa, 2011.
- [11] L.J. & O.Gwun, "A Comparison of SIFT, PCA-SIFT and SURF," International Journal of Image Processing, vol. 3, no. 4, pp. 143-152, 2008.
- [12] C.Liu, J.Yang and H. Huang, "P-SURF: A Robust Local Image Descriptor," Journal Of Information Science And Engineering, vol. 27, pp. 2001-2015, January 2011.
- [13] M.Trajkovic and M. Hedley, "Fast corner detection," Image and Vision Computing, vol. 6, no. 2, pp. 75-87, 1998.
- [14] H.A. Elnemr, "Statistical Analysis of Law's Mask Texture Features for Cancer and Water Lung Detection," International Journal of Computer Science, vol. X, no. 6, pp. 196-202, 2013.
- [15] A.Olaode, N.Golshah and C.Todd, "Unsupervised Image Classification by Probabilistic Latent Semantic Analysis for the Annotation of Images," in 2014 International Conference on Digital Image Computing: Techniques & Applications (DICTA), Wollongong, 2014.
- [16] N.Dalal and B.Triggs, "Histograms of Oriented Gradients for Human Detection," INRIA, Montbonnot, 2004.
- [17] S.Lazebnik, C.Schmid and J.Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," in Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference, Illinois, 2006.

AUTHORS

Abass A. Olaode obtained a Master of Engineering in Telecommunication from the University of Wollongong, Australia in 2012, and is currently a research student at the same institution. He is currently conducting a research into the application of unsupervised machine learning in the elimination of semantic gap from image retrieval.



Golshah Naghdy is an Associate Professor at the School of Electrical, Computer and Telecommunication Engineering, University of Wollongong. She was a Senior Lecturer at Portsmouth University before joining Wollongong University in 1989. Her research interests include biological and machine vision in particular a generic vision system based on "wavelet neurons" and its application in the development of artificial retina implants, medical image processing, content based image retrieval, and robotics.



A DECISION TREE BASED PEDOMETER AND ITS IMPLEMENTATION ON THE ANDROID PLATFORM

Juanying Lin, Leanne Chan and Hong Yan

Department of Electronic Engineering,
City University of Hong Kong, Hong Kong, China
juanyilin2-c@my.cityu.edu.hk,
leanne.chan@cityu.edu.hk, h.yan@cityu.edu.hk

ABSTRACT

This paper describes a decision tree (DT) based pedometer algorithm and its implementation on Android. The DT- based pedometer can classify 3 gait patterns, including walking on level ground (WLG), up stairs (WUS) and down stairs (WDS). It can discard irrelevant motion and count user's steps accurately. The overall classification accuracy is 89.4%. Accelerometer, gyroscope and magnetic field sensors are used in the device. When user puts his/her smart phone into the pocket, the pedometer can automatically count steps of different gait patterns. Two methods are tested to map the acceleration from mobile phone's reference frame to the direction of gravity. Two significant features are employed to classify different gait patterns.

KEYWORDS

Pedometer, Decision Tree, Sensor, Gait analysis & Classification, Mobile Phone Applications.

1. INTRODUCTION

Commonly used pedometers are often built as separate products and their accuracy is often affected by random motions. In this paper, we present a new method to count steps of walking using a mobile phone. We use several sensors to extract signal features and a decision tree to perform data classification. Gyroscopes and accelerometers are widely used to detect human motions. Gyroscope sensor is used to measure the angular velocity of an object. Doheny et al. used a single gyroscope to analyze spatial gait [1]. Lim et al. proposed a gyroscope-based pedometer [2]. A gyroscope is adhered to the right shank segment to detect user's motion. The work presented here uses gyroscope to measure angular velocity of user's thigh, when the phone is in the user's pocket as shown in Figure 1.

The accelerometer can be used as a sensor to measure the acceleration of an object. Aguiar et al. used the accelerometer embedded in smart phone to detect falling of the elder [3]. Mantyjärvi et al. used accelerometers to recognize human motions [4]. The magnetic field sensor is often used in global positioning system navigation. In this work, data from this sensor are used to generate a rotation matrix. Using the matrix and the original acceleration, the vertical acceleration can be determined. Decision tree is one of the predictive modeling approaches used in statistics, data mining and machine learning. In a decision tree [5], leaves represent target values, which are also called class labels, and branches represent measurements about an item, which is also called a feature. Manap et al. used decision tree to detect parkinsonian gait motor impairment [6].

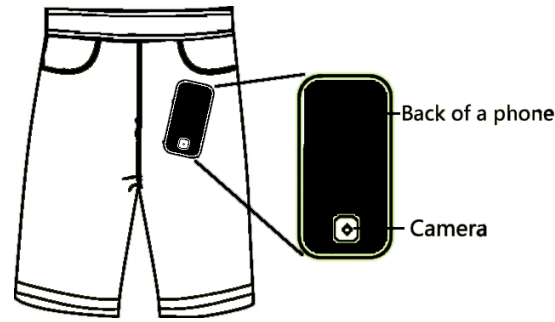


Figure 1. An example of a mobile phone in the user's pocket.

In the work by Lovell et al. [7], a sliding window with a size of 128 samples is used to segment signal of acceleration. In our work, an angular velocity based algorithm is developed to segment the signal of acceleration. Using this algorithm, classification and step counting can be done at the same time. In many pervious work of pedometer or gait classification, such as [2], [7], [8], researchers did not consider the capacity of anti-interference of their systems. In our method, daily irrelevant motions are added to the training set to improve the capacity of anti-interference of the system.

2. SYSTEM STRUCTURE

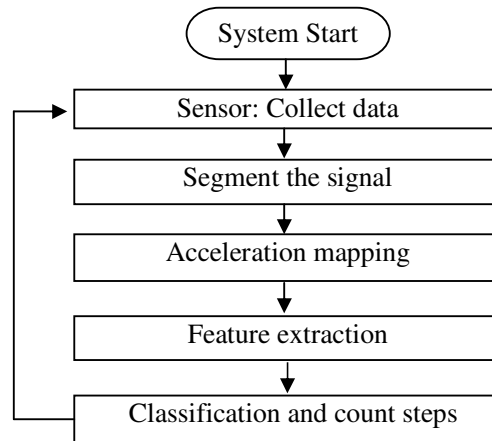


Figure 2. System follow chart of the pedometer.

The system structure of the proposed pedometer is shown in Figure 2. Signals of original acceleration, angular velocity and magnetic field are recorded with a sampling frequency of 100 Hz. Then the signals will be cut into small segments. After that, original acceleration is mapped to the direction of the gravity. The features are extracted from each segment. Finally, all features are sent to the decision tree to classify each segment.

3. SIGNAL PROCESSING ALGORITHMS

3.1. Algorithms for signal segmenting

To the thigh, a cycle of walking only contains 2 phases, forward rotation (FR) and backward rotation (BR). According to the case shown in Figure 1 and the reference frame of gyroscope shown in Figure 3, BR and FR can be detected by the x-axis of gyroscope in mobile phone.

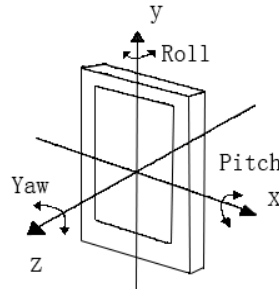


Figure 3. Three axes of the gyroscope in a mobile phone.

Most mobile phones now have a large screen and occupy most space of user's pocket. Therefore, the position of mobile phone is usually stable in the user's pocket and it is reliable to use the x-axis to detect FR and BR. FR and BR can be easily recognized in the signal of x-axis as shown in Figures 5, 6 and 7. The principle of the algorithm is to detect FR of user's thigh and use it to separate signal of each step. The system will continue to monitor the angular velocity of x-axis and detect FR.

If there are 15 consecutive data points whose values are all less than -1 rad/s, an FR is detected. The start point of a segment is the first data point with positive value after the FR. The start point is located by monitoring the first positive point after detecting the 15 consecutive negative points. The end point of a segment is the last peak whose value is larger than 1 rad/s, before the FR of the next step. A peak is located by checking whether there is a data point denoted by $x(n)$ that meets the requirement: $x(n) - x(n-1) > 0$ and $x(n+1) - x(n) < 0$, where n denotes the index of the data point. After setting the start point, if no FR is detected, the end point of the segment will be set to be the 150th data point after the start point. The signals of angular velocity and vertical vibration are segmented according to start points and end points as shown in Figures 4, 5, 6 and 7. FR is an important element of a walk-like event. If no FR is detected, no segments will be created as illustrated by the signals after the segmentation in Figure 8. Therefore, some irrelevant motions are discarded and the reliability of the system is improved.

Using this algorithm, one segment represents a walk-like event. The system can simply count the number of segments, which are considered to be true walk events by a decision tree, and obtain the number of steps of different gait patterns as shown in Figure 8. Let S_{WLG} , S_{WUS} , and S_{WDS} denote the number of steps of the 3 gait patterns. The mobile phone only monitors one of user's thighs. Therefore, one segment represents 2 steps.

If a sliding window is used, one step might be detected with 2 consecutive windows, and the number of segments is not equal to the number of steps. Also, a step near the boundary of a segment might be missed, if the step counting algorithm is applied to a larger segment that consists of several small segments.

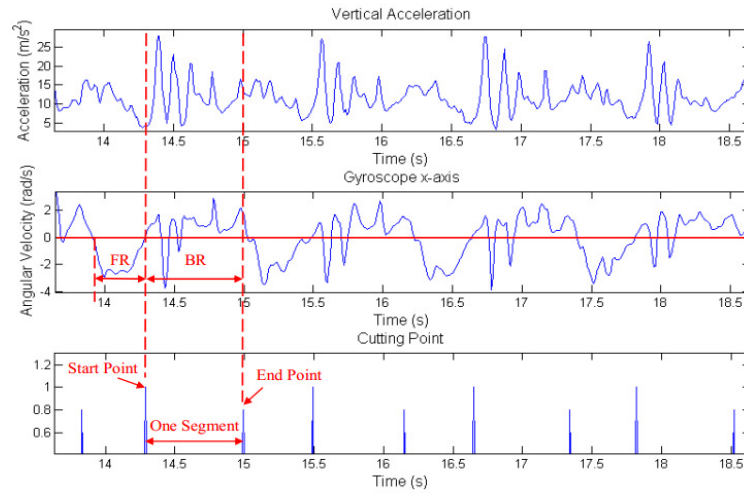


Figure 4. Signals of walking on a level ground

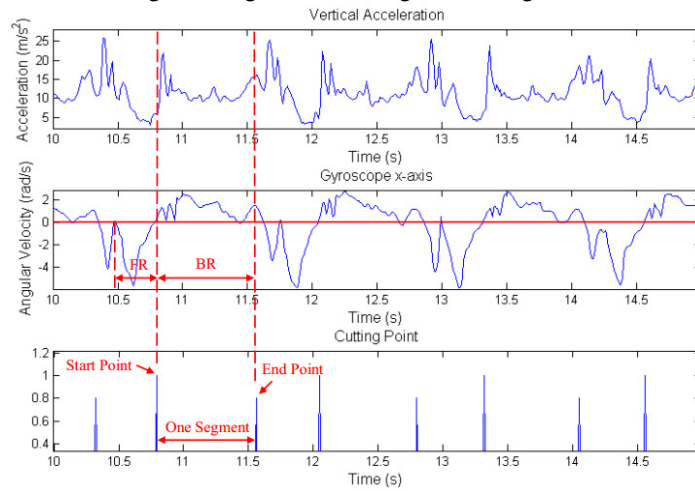


Figure 5. Signals of walking up stairs.

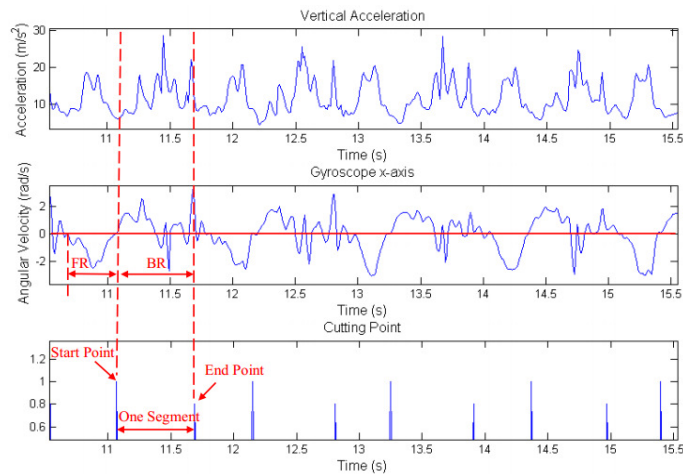


Figure 6. Signals of walking down stairs.

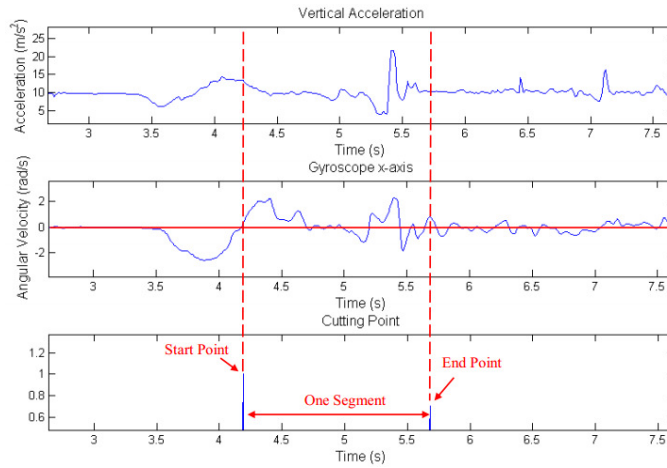


Figure 7. Signals of walking down stairs.

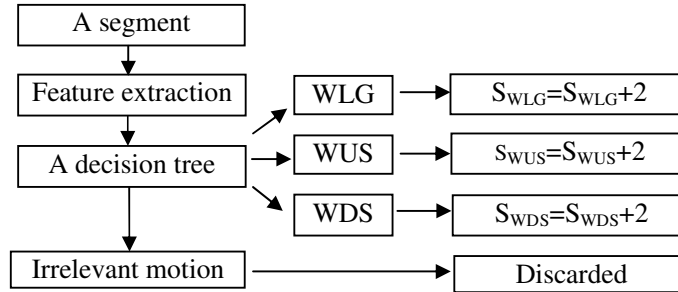


Figure 8. Gait classification and step counting.

3.2 Acceleration mapping

The acceleration given by the mobile phone is respected in the mobile phone reference frame shown in Figure 3. Vertical vibration is a significant signal induced by the walk. Therefore, original acceleration needs to be mapped to the direction of the gravity to generate the signal of vertical vibration. There are two methods to achieve it.

In the first method, we calculate the angle between vector of linear acceleration provided by linear acceleration sensor and the vector of g provided by gravity sensor, where g denotes the acceleration due to gravity, and $|g|=9.8 \text{ m/s}^2$. Let A_{linear} denote the linear acceleration, A_{GD} denote the value of the acceleration in the direction of gravity, and x_{linear} , y_{linear} and z_{linear} denote elements of the vector of A_{linear} respectively, then they can be computed as follows.

$$\left| \vec{A}_{\text{linear}} \right| = \sqrt{x_{\text{linear}}^2 + y_{\text{linear}}^2 + z_{\text{linear}}^2} \quad (1)$$

$$\cos \langle \vec{A}_{linear}, \vec{g} \rangle = \frac{\vec{A}_{linear} \cdot \vec{g}}{\left| \vec{A}_{linear} \right| \left| \vec{g} \right|} \quad (2)$$

$$A_{GD} = -\cos \langle \vec{A}_{linear}, \vec{g} \rangle \cdot \left| \vec{A}_{linear} \right| \quad (3)$$

In the second method, a rotation matrix can be generated by `getRotationMatrix`, a function provided by Android, using data from the accelerometer and the magnetic field. Then the original acceleration can be mapped to the direction of gravity. Let A_{GD} denote the value of the acceleration in the direction of gravity, $M_{rotation}$ denote the rotation matrix and $A_{original}$ denote the vector of acceleration respecting to mobile phone's reference frame, then

$$[0, 0, A_{GD}] = M_{rotation} \times A_{original} \quad (4)$$

In Android, there are two kinds of sensors, hardware-based sensors and software-based sensors. Software-based sensors need data from several hardware-based sensors to produce its own data [9]. This means they need more time to process their data. Linear acceleration sensor and gravity sensor are software-based. Method 2 is better than Method 1 because the former is based on hardware-based sensors and needs less time to generate the data.

4. SIGNIFICANT FEATURES

In Figure 9, D_{FS} denotes the distance between one's foot and the surface of ground or stairs. $Length_M$ denotes the length between the start point and the point of the maximum in a segment. $Length_S$ denotes the length of a segment. In the signal of vertical vibration, two significant features, location of peak in a segment and variance are found to classify different gait patterns.

4.1. Location of the maximum in a segment

The definition is shown as follows. $Location_M$ denotes the location of the maximum in a segment. Data of a segment is recorded in an array. Each data point has its own index. $Index_S$ denotes the index of the start point. $Index_E$ denotes the index of the end point. $Index_M$ denotes the index of the data point with maximum value. $Length_S$ denotes the length of a segment. $Length_M$ denotes the distance between the start point and the point with the maximum value.

$$Length_M = Index_M - Index_S \quad (5)$$

$$Length_S = Index_E - Index_S \quad (6)$$

$$Location_M = \frac{Length_M}{Length_S} \times 100 \quad (7)$$

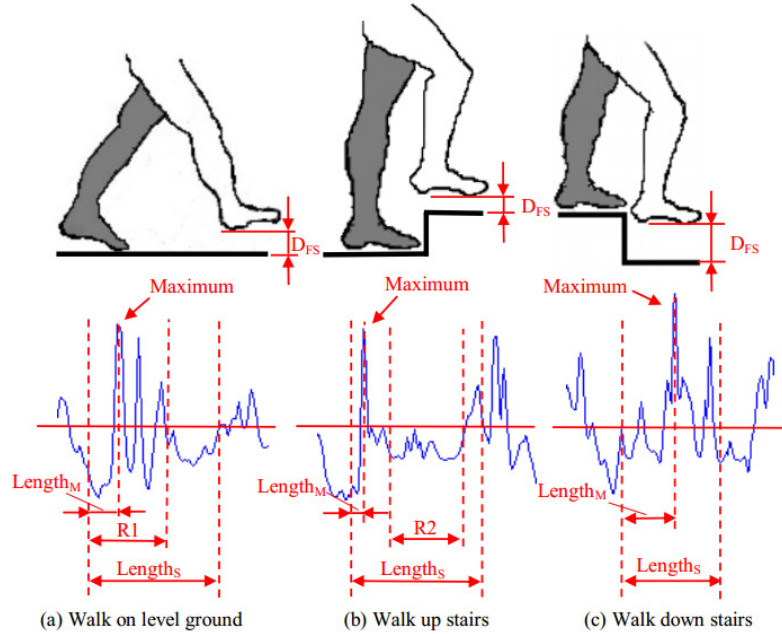


Figure 9. Gait analysis and signal of vertical vibration

Signal in a segment represents the BR of user's thigh, as shown in Figure 9. In motions of walking on level ground or walking up stairs, when the user begins to rotate his/her thigh backward, his/her foot will soon touch the surface, due to small D_{FS} . Then vibration is induced by heel strike. Therefore, the maximum is located near the start point of a segment. When walking down stairs, D_{FS} is larger. After beginning to rotate one's thigh backward, one's foot will not soon touch the surface and induce vibration. Then the maximum is not located near the start point. This feature can separate the motion of walking down stairs from the other 2 gait patterns.

4.2 Variance of a segment

Let D_a denote the average of data values in a segment, n denote the number of data points, and D_i denote a specified data point. Var denotes the variance of a segment. We calculate the variance of a segment as follows

$$D_a = \frac{1}{n} \sum_{i=1}^n D_i \quad (8)$$

$$Var = \frac{1}{n} \sum_{i=1}^n (D_i - D_a)^2 \quad (9)$$

In Figure 9, $R1$ and $R2$ denote 2 ranges in the signals of walking on level ground and down stairs respectively. $R1$ is a range of vibration. This range will increase the variance of the segment. Compared with $R1$, $R2$ is a range that is relatively flat. $R2$ will decrease the variance of the segment. Therefore, the variance of a segment of walking on level ground will be larger than that in a segment of walking up stairs. Using this feature, the motion of walking on level ground can be distinguished from the motion of walking up stairs.

5. TRAINING SET AND DECISION TREE

Five subjects, participate in an experiment to create a training set with 4 kinds of motions, including the 3 gait patterns and irrelevant motion. We choose the decision tree as the classification engine since it has a very low computational complexity and can be implemented on a mobile computing unit (MCU) efficiently [10-12]. In order to avoid imbalanced distribution of different classes in a decision tree, the amount of each class in a training set should be balanced. If one class is the majority in a training set, the decision tree created by this training set is more likely to classify an unknown instance to that class. Then C4.5 algorithm in Weka is used to identify distinct features and create a decision tree, according to the training set. Six features are selected to create the decision tree as shown in Figure 10. In this figure, “Ground” denotes walking on level ground. “Up” denotes walking up stairs. “Down” denotes walking down stairs. “Other” denotes irrelevant motion. In the signal of vertical acceleration, AcceleGMin denotes the minimum value, AcceleGMax denotes the maximum value, AcceleGAverage denotes the average, AcceleGMaxL denotes the location of the maximum and AcceleGVar denotes the variance. In the signal of angular velocity, GyroMin denotes the minimum.

```

AcceleGVar <= 16.988167
| AcceleGAverage <= 10.725424: Other
| AcceleGAverage > 10.725424
| | AcceleGMin <= 6.927609
| | | AcceleGMaxL <= 20
| | | | GyroMin <= -1.517695: Other
| | | | GyroMin > -1.517695: Up
| | | | AcceleGMaxL > 20
| | | | | AcceleGMin <= 5.061104
| | | | | | GyroMin <= -3.314555: Up
| | | | | | GyroMin > -3.314555: Other
| | | | | AcceleGMin > 5.061104
| | | | | | AcceleGMin <= 6.043906: Ground
| | | | | | AcceleGMin > 6.043906: Up
| | | | | AcceleGMin > 6.927609: Other
AcceleGVar > 16.988167
| AcceleGMaxL <= 42
| | AcceleGMaxL <= 7.619048: Other
| | AcceleGMaxL > 7.619048
| | | AcceleGMax <= 29.776773: Ground
| | | AcceleGMax > 29.776773: Other
| | AcceleGMaxL > 42
| | | AcceleGMin <= 3.332462: Other
| | | AcceleGMin > 3.332462
| | | | AcceleGMaxL <= 82: Down
| | | | AcceleGMaxL > 82
| | | | | AcceleGMin <= 4.426645: Down
| | | | | AcceleGMin > 4.426645: Other

```

Figure 10. Decision tree used in the system

While creating the classifier Weka also evaluate the performance of this predictive model. Cross validation is a common method to evaluate the accuracy of classifiers [10]. In Leave One-Out (LOO) cross validation, one subject is used for testing and the rest are used for training. The classification result is then computed and repeated until all subjects have participated in the testing dataset. The overall classification result is then computed as the average of all testing subjects [13]. Here 10-folder cross-validation is used to measure the accuracy of this classifier. In

the cross-validation, the whole training dataset is divided into 10 subsets. One subset is used for testing the rest are used for training. The classification accuracy of this decision tree is 92.3645 %. Another measure of classification algorithms performance is a confusion matrix [14]. Precision and recall are typical classification performance measures using the confusion matrix [15]. Precision and recall are defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (10)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (11)$$

To a class in training dataset, TP (True Positive) denotes the number of correctly classified positive instances, FN (false negative) denotes the number of positive instances incorrectly classified as negative; TN (true negative) denotes the number of correctly classified negative instances and FP(false positive) denotes number of negative instances incorrectly classified as positive. The precise and recall of each class is shown in table 1.

Table.1 Confusion matrix of the decision tree

Class	Classified as				Measurement				
	WLG	WUS	WDS	Other	TP	FP	FN	Precision	Recall
WLG	93	1	2	4	93	4	7	95.9%	93.0%
WUS	1	87	0	4	87	8	5	91.6%	94.6%
WDS	0	1	80	6	80	5	7	94.1%	92.0%
Other	3	6	3	115	115	14	12	89.1%	90.6%

6. EXPERIMENT RESULT

The decision tree based pedometer is tested in a walking experiment and an anti-interference experiment. Subjects were asked to wear a Samsung Gear fit [16], a wearable device, in the two experiments. Then the efficiency of the proposed system can be compared with that of the Gear fit. Four subjects participated in these two experiments. In the walking experience, each subject was asked to take 200 steps on level ground, go up 4 floors, then go down 4 floors. Each floor has 16 stairs. In the anti-interference experiment, subjects were asked to shake or swing the mobile phone and the Gear fit 10 times at the same time and to see whether the pedometer and Gear fit take those motions as steps. Samsung Gear fit cannot classify gait patterns. The accuracy of Gear fit in Table 1 only represents the accuracy of step detection. In the walking experiment, the overall classification accuracy is 89.4%. In the anti-interference experiment, the average false steps recorded by the pedometer are 2.5, while Gear fit produces 12 false steps, as shown in Table 2.

Table .1 Accuracy of step detection

Gait Pattern	Total Steps	Proposed Pedometer		Samsung Gear Fit	
		Steps Detected	Accuracy of Step Detection	Steps Detected	Accuracy of Step detection
WLG	800	776	97.0%	779	97.4%
WUS	256	230	89.8%	235	91.8%
WDS	256	238	92.9%	210	82.0%
Average	-----	-----	93.2%	-----	90.4%

Table .2 Accuracy of classification

Proposed Pedometer			
Gait Pattern	Total Steps	Steps Correctly Classified	Accuracy of Classification
WLG	800	752	94.0%
WUS	256	218	85.2%
WDS	256	228	89.1%
Average	-----	-----	89.4%

Table 3. Result of anti-interference experiment

	False Steps Recorded				
	Subject1	Subject2	Subject3	Subject4	Average
Proposed Pedometer	2	0	4	4	2.5
Gear fit	13	10	17	8	12

7. CONCLUSION

A decision-based pedometer that can count steps and classify 3 gait patterns is developed. An angular velocity based algorithm is used in this pedometer to segment signals and enable the pedometer to count steps of different gait patterns easily. The decision tree is used to improve the accuracy and reliability of the pedometer. The system has been tested in several experiments with good results. The experiment results show that the proposed pedometer produces much less false step count than a commercial product.

ACKNOWLEDGEMENTS

This work is supported by City University of Hong Kong (Project 6987027).

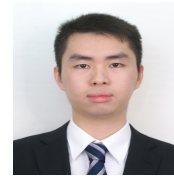
REFERENCES

- [1] E.P.Doheny, T.G.Foran and B.R.(2010) Greene, "A single gyroscope method for spatial gait analysis," in Proc. EMBC, pp. 1300-1303.
- [2] Y.P.Lim, I.T.Brown and J.C.T.Khoo,(2008) "An accurate and robust gyroscope-based pedometer," in Proc. EMBS, pp. 4587-4590.
- [3] B.Aguiar, T.Rocha, J.Silva and I.Sousa,(2014) "Accelerometer-based fall detection for smartphones," in Proc. MeMeA, pp. 1-6.
- [4] J.Mantyjarvi, J.Himberg and T.Seppanen,(2001) "Recognizing human motion with multiple acceleration sensors," in Proc. Systems, Man, and Cybernetics, vol. 2, pp.747-752.
- [5] Rokach, L., O.Maimon,(2008) Data mining with decision trees: theory and applications, World Scientific Pub Co Inc, pp vii and 71.
- [6] H.H.Manap, N.Md Tahir, R.Abdullah,(2013) "Parkinsonian Gait Motor Impairment Detection Using Decision Tree," in Proc. EMS, pp. 209-214.
- [7] N.H.Lovell, N.Wang, E.Ambikairajah, B.G. Celler,(2007) "Accelerometry Based Classification of Walking Patterns Using Time-frequency Analysis," in Proc. EMBS, pp. 4899-4902.
- [8] J.S.Wang, C.W.Lin, T.C.Yang, Y.J Ho,(2012) "Walking Pattern Classification and Walking Distance Estimation Algorithms Using Gait Phase Information," IEEE Trans. Biomedical Engineering, vol. 59, no. 10, pp. 2884-2892.
- [9] Motion Sensors in Android http://developer.android.com/guide/topics/sensors/sensors_motion.html
- [10] R.O.Duda, P.E.Hart & D. G. Stork,(2001) Pattern Classification. 2nded, New York: Wiley.

- [11] Z.Chi and H.Yan, (1996) "ID3-derived fuzzy rules and optimized defuzzification for handwritten numeral recognition," IEEE Trans. on Fuzzy Systems, vol. 4, no. 1, pp. 24-31.
- [12] S.Zhao, Z.Chi, P. Shi and H. Yan, (2003) "Tow-stage segmentation of handwritten Chinese characters based on fuzzy decision rules," Pattern Recognition, vol. 36, pp. 145-156.
- [13] A.Murad, T.Sarkodie-Gyan, H.Y.Yu, O.Fuentes, R.Brower. & A.Abdelgawad, (2011) "Automatic classification of pathological gait patterns using ground reaction forces and machine learning algorithms," EMBC, pp.453-457.
- [14] N.V.Chawla, K.W.Bowyer, L.O.Hall, and W.P.Kegelmeyer, (2002) "SMOTE: Synthetic Minority Over-sampling Technique", Journal of Artificial Intelligence Research, vol. 16, pp. 321 -357.
- [15] D.L.Olson & D.Delen (2008) Advanced Data Mining Techniques. Springer.
- [16] Information on Samsung Gear Fit:
http://www.samsung.com/global/microsite/gear/gearfit_features.html

AUTHORS

Juanying Lin received his master degree in Electronic and Information Engineering from City University of Hong Kong. He is currently a research assistant in the Department of Electronic Engineering at City University of Hong Kong. His research interests include digital image processing, motion pattern recognition and mobile computing.



Leanne Chan received her PhD degree in Biomedical Engineering from University of Southern California. She is currently Assistant Professor in the Department of Electronic Engineering at City University of Hong Kong. Her research interests include retinal prosthesis testbed development and microelectronic systems.



Hong Yan received his Ph.D. degree from Yale University. He was professor of imaging science at the University of Sydney and currently is professor of computer engineering at City University of Hong Kong. He was elected an IAPR fellow for contributions to document image analysis and an IEEE fellow for contributions to image recognition techniques and applications. His research interests include bioinformatics, image processing and pattern recognition.



INTENTIONAL BLANK

E-LEARNING SCENARIOS USING INTELLIGENT MULTIAGENT SYSTEMS

Ali M. Aseere

College of Computer Science, King Khalid University, KSA
amg@kku.edu.sa

ABSTRACT

Agent technologies could be a good approach to solving a number of problems concerned with personalised learning due to their inherent autonomy and independence. In this paper, we describe a number of e-learning scenarios that could be addressed by agent technologies. We then analyse these scenarios highlighting how specific agent features such as coalition formation and bargaining (Negotiation) could be used to solve the problem. Our aim is to show how agent systems can not only form a good framework for distributed e-learning systems, but how well they match to situations where learners are themselves autonomous and independent.

KEYWORDS

Coalition formation, personalised learning, Negotiation.

1. INTRODUCTION

E-learning systems have been used for a number of years in the delivery and management of learning content. More recently there has been a focus on how they could help personalize the learner's experience, as students begin moving from a world of VLEs (Virtual Learning Environments) into a space where students are taking more control of their learning in the form of PLEs (Personal Learning Environments).

Agent systems are a good approach to building systems where different people (represented by agents) may have orthogonal goals. As such they match very well with the world of personalized learning, where many students may have to negotiate with their peers, their tutors and their institution in order to achieve what they want within the constraints of other people's goals and objectives.

In this paper we present two scenarios showing the use of agent technologies in e-learning, the first is an independent student selecting a course, the second is personalized learning styles. We then analyse these scenarios showing how specific agent features such as coalition formation and bargaining (negotiation) might be used to solve that kind of problem

2. RELATED WORK

A number of researchers have applied agent technology to e-learning. Yang et al proposed to apply intelligent system to enhance navigation-training systems that consists of the client portion

and server portion using JADE framework [1]. However, they focus on the intelligence of the agents themselves, rather than communications between agents.

Shi et al. designed an agent system for computer science education that focuses on two courses where the learning process is student-centered, self-paced and highly interactive [2]. They use Java RMI, JavaSpace and JATLite to create a web-based system; in this case they use personal agents to manage student's data and their interactions with course material.

Agent technologies work well as a framework for building distributed systems, however it is only when some form of conversation, self-organisation or negotiation is needed that they become really valuable.

For example, Soh et al have shown a system called Intelligent Multiagent infrastructure for Distributed Systems in Education to support student in real time classroom where a buddy group is formed dynamically to support the members to achieve common goals [3].

In the next two sections we look at two scenarios that really utilize the potential of agent technologies, including a generalised version of this kind of group formation. In each case we provide a general description of the scenario, an analysis of the agent features within the scenario, and variations of the scenario that would also share the same features

3. SCENARIO 1: COURSE SELECTION

3.1. Description

Adult learners often act more independently than students straight out of school or college; they are also often enrolled in part-time courses where there is a lot of flexibility in how they receive credits. In these circumstances, the learners are free to choose which courses they wish to take, however the University is restricted on which courses it might run, due to the overheads of running each course.

Students therefore have to not only find courses which match their preferences, but also ensure that they enrol on courses in enough numbers to allow those courses to run. This might be compromising on their preferences, or changing their choices in the light of student numbers. In this kind of situation, the University is relatively passive, however it might help students reach agreement by suggesting courses that are likely to run.

3.2. Agent Features

In this scenario the students need to find other students with similar interests, and act together so that courses they are interested in will run. In Agent systems this is called a coalition formation problem, it occurs whenever agents are required to form groups in order to achieve some common goals [4].

Individual agents may need to compromise when joining a coalition, but the advantages of being inside a coalition outweigh the disadvantages, and may even be the only way for the agent to achieve their goals. In this case fulfilling the student's preferences becomes the goal for each student agent.

3.3. Variations

This scenario is all about students making choices between alternatives, but where there are some external restrictions on those choices that might make it useful to act together. As way as making module choices this would also be relevant for students choosing a course, or even making choices between Universities. In this last case the factors might also include issues such as reputation, distance from home, facilities and so forth.

4. SCENARIO 2: PERSONALIZED LEARNING STYLES

4.1. Description

Different student may have different personal preferences about the way they want to learn or be assessed. For example, students may have a preferred learning style (for example, some students may prefer information presented visually). However, an institution may have regulations about having a mixed set of assessment styles. For example, many Universities are cautious about having modules assessed by 100% course work.

This can cause conflict, as the student wants to be assessed in the way they prefer the most. In these cases there is a need for the student to negotiate with their teacher about the methods of learning or assessment that will be used.

4.2. Agent Features

In this scenario there is an institution that associates a cost with each type of learning or assessment and wants to minimize that cost (or at least prevent it from rising above an agreed level). This cost need not be financial, and could include factors such as value to external assessors, or complexity for staff to manages.

For each student we can define a *utility function* that calculates their satisfaction with the styles they have been allocated. Students can now bargain (negotiate) with their institution, exchanging items according to their cost until their utility function is maximized within the constraints of the institution's cost level

4.3. Variations

This scenario could also be applied to students with learning differences, such as dyslexia, in which case they could have different requirements about how they are assessed (for example, preferring project work to exams). An institution might apply different rules to these students, but could use the same economic framework to negotiate their assessment.

5. SUMMARY OF AGENT TECHNIQUES

In the scenarios presented above, we have found that agent technologies, which are based on economic models, such as coalition formation and bargaining can be applied in an e-learning setting.

Coalitions of agents are temporary groups that exist to solve a particular problem. There are generally two different approaches to solving the coalition formation problem. With the centralized approach a single agent (for example, the institution) gathers all the data and makes a decision[5]. It has the advantage of being simpler to manage, but doesn't really take advantage of

agent autonomy. With the decentralised approach the problem is solved by a number of different agents working together [6]. A decentralised approach could allow students to form their own coalitions independently (e.g., in Scenario 1 by negotiating amongst themselves about a strategy for course registration to maximize their satisfaction as a whole).

Bargaining (negotiation) is a technique for reaching agreements in a multi-agent system. Any negotiation setting will have four different components [7]:

- A negotiation set, which represents the space of possible proposals that agents can make.
- A protocol of the legal proposals agents can make.
- A collection of strategies (the strategy that each agent plays is private)
- A rule that determines when a deal is struck and what the agreement is.

In Scenario 2 the institution and students have a conflict of interest. Bargaining is used to resolve conflicts, and achieve agreement between them

5. CONCLUSION

In this paper we have argued that agent technologies could be a good match for personalized e-learning. We have presented two scenarios showing the use of agent technology in e-learning and identified a number of agent techniques that could be used to solve the challenges in each scenario: coalition formation and economic systems based on bargaining (negotiation).

We plan to continue this work by building a number of prototype agent systems in the JADE agent framework, and exploring whether existing e-learning standards and models of student preferences are sufficient to support an agent-based solution

REFERENCES

- [1] Yang, C., Lin, H. and Lin, F. O. Designing Multiagent-Based Education Systems for Navigation Training. 5th IEEE Int. Conf. on Cognitive Informatics (ICCI'06)2006).
- [2] Shi, H., Shang, Y. and Chen, S.-S. A multi-agent system for computer science education. SIGCSE Bull., 32, 3 2000), 1-4.
- [3] Soh, L.-K., Jiang, H. and Ansorge, C. Agent-based cooperative learning: a proof-of-concept experiment. SIGCSE Bull., 36, 1 2004), 368-372.
- [4] Horling, B. and Lesser, V. A survey of multi-agent organizational paradigms. Knowl. Eng. Rev., 19, 4 2004), 281-316.
- [5] Akinin, S. A reliable algorithm for multi-agent coalition formation. IEEE1999).
- [6] Shehory, O. and Kraus, S. Methods for task allocation via agent coalition formation. Artificial Intelligence, 101, 1-2 1998), 165-200.
- [7] Wooldridge, M. An Introduction to Multiagent Systems Wiley & Sons, Chichester, England 2002.

IMPROVEMENT WSD DICTIONARY USING ANNOTATED CORPUS AND TESTING IT WITH SIMPLIFIED LESK ALGORITHM

Ahmed H. Aliwy¹ and Ayad R. Abbas²

¹Department of Computer Science, University Of Technology, Baghdad, Iraq

ahmed_7425@yahoo.com

²Department of Computer Science, University Of Technology, Baghdad, Iraq

ayad_cs@yahoo.com

ABSTRACT

WSD is a task with a long history in computational linguistics. It is open problem in NLP. This research focuses on increasing the accuracy of Lesk algorithm with assistant of annotated corpus using Narodowy Korpus Języka Polskiego (NKJP “Polish National Corpus”). The NKJP_WSI (NKJP Word Sense Inventory) is used as senses inventory. A Lesk algorithm is firstly implemented on the whole corpus (training and test) and then getting the results. This is done with assistance of special dictionary that contains all possible senses for each ambiguous word. In this implementation, the similarity equation is applied to information retrieval using tf-idf with small modification in order to achieve the requirements. Experimental results show that the accuracy of 82.016% and 84.063% without and with deleting stop words respectively. Moreover, this paper practically solves the challenge of an execution time. Therefore, we proposed special structure for building another dictionary from the corpus in order to reduce time complicity of the training process. The new dictionary contains all the possible words (only these which help us in solving WSD) with their tf-idf from the existing dictionary with assistant of annotated corpus. Furthermore, eexperimental results show that the two tests are identical. The execution time - of the second test dropped down to 20 times compared to first test with same accuracy

KEYWORDS

Corpus-based WSD, Lesk algorithm, dictionary and corpus based WSD.

1. INTRODUCTION

A **word sense disambiguation** is the task of determining the suitable sense for the ambiguous words in the context. It has a long history in computational linguistics, and is a non-trivial task result from the nature of language semantics. All the current used algorithms did not achieved high levels of accuracy. Many of these algorithms depend on contextual similarity for selecting the proper sense [1].

The revolution of the work on WSD may be start in 1980's where the digital large-scale lexical resources became widely available [2].

Tasks of many NLP improved using WSD because word sense ambiguities results in many problems in many applications such as, **information retrieval, machine translation, question**

answering, information retrieval, and text classification therefore; WSD is an important task. WSD algorithm is varying from one application to another [1]. In this paper, we ignored the applications and focuses on the implementation and evaluation of WSD systems as a stand-alone task.

Most of the used WSD algorithms are dictionary-based or corpus-based methods. Dictionary-based methods choose the sense whose gloss or definition shares the most words with the target word's neighborhood [1]. Lesk algorithm is an example of dictionary based methods. It is the most well-studied algorithm for sense disambiguation.

Corpus-based methods require an annotated data where each ambiguous word has the correct sense in each sentence. This annotation is done by intervention of human. Most of these methods are called "supervised" because they learn from previously sense annotated data [3]. Lesk Algorithm gives better results if it used with annotated corpus than without.

2. RELATED WORK

Our approach, as we will see, is improving the WSD dictionary using annotated corpus. The tests are made by using Lesk algorithm. There are many researches in WSD field. Most of them used dictionary based, corpus based and hybrid methods.

Montoyo et al. [4] presented two WSD methods based on two main methodological approaches: a knowledge-based method and a corpus-based method. Their approach combines various sources of knowledge, through combinations of the two WSD approaches as mentioned above. They showed how to combine these methods and sources of information in order to achieve good results in the disambiguation. Little combinations are based while some of them are voting.

Ledo-Mezquita et al. [5] proposed a method of word sense disambiguation based on the combination of the original Lesk method and the simplified one with additional application of large lexical resources, like synonym dictionaries, ontologies, etc. their experimental results show that the method has better precision than the baseline Lesk-based methods. In other side, Basile et al. [6] adopted the simplified Lesk algorithm to disambiguate adjectives and adverbs, combining it with other two methods for nouns and verbs.

Ponzetto and Navigli [7] try to maximize the chances of overlap between glosses or between the gloss and the context in Lesk algorithm by extended WordNet with Wikipedia pages.

Viveros-Jiménez et al. [8] proposed simple strategies for the context window selection that improve the performance of the Simple Lesk algorithm by: (1) constructing the window only with words that have an overlap with some senses of the target word, (2) excluding the target word itself from matching, and (3) avoiding repetitions in the context window.

Schwab et al. [9] proposed GETALP: an unsupervised WSD algorithm inspired by Lesk. A local similarity are computed using the classical Lesk measure (overlap between glosses), and then these local similarity is propagated to the whole text (global similarity) using an algorithm inspired by the Ant Colony. This approach was applied to English and Italian.

Basile et al. [10] described an algorithm which extends two well-known variations of the Lesk WSD method. The main contribution of them approach relies on the use of a word similarity function defined on a distributional semantic space to compute the gloss-context overlap. They used private dataset (BabelNet API 1.1.1) provided by the authors. They used two languages, Italian and English, in the tests.

3. DICTIONARIES BASED AND THESAURI WSD METHODS

In 1980's, after the theses of Amsler's (1980) and Michiel's (1982), Machine-readable dictionaries (MRDs) became an important source of knowledge. The first attempts were to extract lexical and semantic knowledge bases from MRDs. Then, it became the basis of lexical semantic studies [2].

WordNet is an example of improved MRD. It encodes a rich semantic network of concepts therefore; it can be an important source of word senses in English language [1]]. MRDs rapidly became a staple of WSD research but they lack pragmatic information that enters into sense determination [2].

Thesauri provide information about relationships between words, like synonymy, antonymy and, possibly, further relations. Roget's International is the most widely used thesaurus in the field of WSD [11]. The most well-known algorithm deals with MRD for WSD is Lesk algorithm.

3.1. Lesk Algorithm

The Lesk algorithm uses dictionary definitions (gloss/examples) to disambiguate a polysemous word in a sentence context. The original Lesk algorithm measures overlap between sense definitions for all words in the text and identify simultaneously the correct senses for all words in the text. A word is assigned to the sense whose gloss shares the largest number of words in common with the glosses of the other words [12]. There are many varieties of Lesk algorithm, the famous are Original and simplified.

We can see the complexity of original Lesk algorithm from the following example: if we have nine open class words with following number of senses: 26, 11, 4, 8, 5, 4, 10, 8, 3 then the sense combinations is 43,929,600. It is huge number and hence practically difficult to implement. It is the main reason for using, other version of Lesk algorithm, simplified Lesk.

3.2. Simplified Lesk Algorithm

The simplified Lesk algorithm is a modified version of Lesk algorithm as shown in figure 1 [13]. It tries to solve each ambiguous word alone without regards for other senses for other words (words which have sense ambiguity in the same context). For this reason, it can be used in practice.

```

function Simplified_Lesk(word, sentence)
  best-sense ← most frequent sense for word
  max-overlap ← 0
  context ← set of words in sentence
  for each sense in senses of word do
    signature ← set of words in the gloss_examples of sense
    overlap ← Compute_overlap(signature, context)
    if overlap > max-overlap then
      max-overlap ← overlap
      best-sense ← sense
  end
  return(best-sense) { returns best sense of word }

```

Figure 1. simplified Lesk algorithm [1].

Simplified Lesk algorithm can be used with annotated corpus by:

- Additionally Use sentences in corpus to compute signature of senses.
- Computing weighted overlap by using *idf*.

4. SIMILARITY AND COMPUTING TF-IDF FROM TRAINING CORPUS

If there are n different words (types; e.g., the most frequent meaningful words in a language) then, a document j may be represented as:

$$\vec{d}_j = (tf_{1,j}, tf_{2,j}, tf_{3,j}, \dots, tf_{n,j}) \quad (1)$$

Where $tf_{i,j}$ is the *term frequency* of the word w_i in document j . Similarly, a query q may be represented as:

$$\vec{q} = (tf_{1,q}, tf_{2,q}, tf_{3,q}, \dots, tf_{n,q}) \quad (2)$$

The simplest approaches are based on the bag-of-words (multiset of words) model. In such a model the similarity between the query and a document is the cosine of appropriate vectors:

$$\text{sim}(\vec{q}, \vec{d}_j) = \frac{\vec{q} \cdot \vec{d}_j}{|\vec{q}| \times |\vec{d}_j|} = \frac{\sum_{i=1}^n tf_{i,q} \times tf_{i,j}}{\sqrt{\sum_{i=1}^n tf_{i,q}^2} \times \sqrt{\sum_{i=1}^n tf_{i,j}^2}} \quad (3)$$

The equation (3) can be used with Lesk Algorithm without annotated corpus.

Because we deal with annotated corpus we need to other factor which is Inverse document frequency (*idf*). Inverse document frequency *idf* assigns more weight to words which are more specific for the given document. For a word w_i :

$$\text{idf}_i = \log \frac{N}{n_i} \quad (4)$$

Where N is the number of documents in the collection and n_i is the number of documents containing the word w_i .

A frequently used measure of the weight of word w_i in document j is *tf.idf* (*tf-idf*, TFIDF)¹. Then, the similarity can be computed as [1]:

$$\text{sim}(\vec{q}, \vec{d}_j) = \frac{\sum_{w \in q, d} tf_{w,q} \cdot tf_{w,j} \cdot (\text{idf}_w)^2}{\sqrt{\sum_{i=1}^n (tf_{i,q} \cdot \text{idf}_i)^2} \times \sqrt{\sum_{i=1}^m (tf_{i,j} \cdot \text{idf}_i)^2}} \quad (5)$$

Where

¹ There are numerous variants of *tf.idf*.

- $tf_{w,q}$ is term frequency of the word w in query q but $tf_{i,q}$ is term frequency of word w_i in the query.
- $tf_{w,j}$ term frequency of the word w in document d_j .
- idf_w is inverse document frequency of the word w in document j .
- n and m are number of the different words in query and document j respectively.

The above equation can be used to calculate the best word sense as following: Suppose a dictionary contains orthogonal entries². Each entry has more than one senses and each sense has some gloss examples. The simple way, for selecting the best sense, is by checking the similarity between each example and the query. The ambiguous entry will take the sense where its example gives high similarity.

4.1. Using invers sense frequency (*isf*):

isf (inverse sense frequency) is equivalent to *idf* with little differences. We deals with all examples of one sense as one **block (like one document)**, lets call it s_j . Then for each word in the examples, *isf* is calculated by³:

$$isf_i = \log \frac{S}{n_i} \quad (6)$$

Where:

- S : is the number of senses for the orthogonal entry.
- n_i : is the number of senses where word w_i appear in them examples.

The similarity between query and s_j , for the current ambiguous entry, is:

$$sim(q, s_j) = \frac{\sum_{w \in q, d} tf_{w,q} tf_{w,j} (isf_w)^2}{\sqrt{\sum_{i=1}^n (tf_{i,q} isf_i)^2} \times \sqrt{\sum_{i=1}^m (tf_{i,j} isf_i)^2}} \quad (7)$$

- $tf_{w,q}, tf_{i,q}$: are same as in equation 5.
- $tf_{w,j}$: term frequency of the word w in the examples of sense j .
- $tf_{i,j}$: is a term frequency of word i in the examples of sense j for current entry.
- isf_w : is inverse sense frequency of the word w .
- n and m are number of the different words in query and the examples of sense j respectively.

Using annotated corpus with the dictionary is very simple task with equation 7. Simply, we add all the sentences have specific entry of specific sense to the examples of this entry sense. For more details see the algorithm in figure 2.

² Orthogonal entry is the dictionary word has more than one sense. We used this term for differentiate it from the other words.

³ P. Basile, etl. (2014) used IGF (Inverse Gloss Frequency)

5. IMPLEMENTATION OF LESK WITH ANNOTATED CORPUS AND RESULTS

The National Corpus of Polish project (Pol. Narodowy Korpus Języka Polskiego; NKJP) [14] was used as a dataset. The used senses inventory is NKJP_WSI (NKJP Word Sense Inventory). The whole corpus was taken with little exceptions⁴. This means, 844 annotated files were processed. The simplified Lesk algorithm is implemented on this corpus. The results, shown in table-1, were obtained by using training and testing 10 fold⁵ of 844 files. The execution time for this experiment was very long time. In this implementation, we depended on the fact that the context was parsed previously and we got from this stage the ambiguous words and *all* their bases. Then, same test was used with deleting stop words. The results of this test are shown in table 2.

Table1. Applying simplified Lesk algorithm on NKJP corpus with deleting stop words.

fold	#Files		matched senses	Total senses	Accuracy %
	test	training			
1	84	760	1428	1792	79.687
2	84	760	505	628	80.414
3	84	760	723	850	85.059
4	84	760	661	775	85.29
5	84	760	176	203	86.699
6	84	760	333	409	81.418
7	84	760	796	954	83.438
8	84	760	1183	1449	81.642
9	84	760	1015	1206	84.162
10	88	760	992	1259	78.793
Total	844		7812	9525	82.016

Table2. Applying simplified Lesk algorithm on NKJP corpus with deleting stop words.

fold	Files		matched senses	Total senses	Accuracy %
	test	training			
1	84	760	1480	1792	82.589
2	84	760	533	628	84.872
3	84	760	720	850	84.705
4	84	760	668	775	86.193
5	84	760	179	203	88.177
6	84	760	356	409	87.041
7	84	760	816	954	85.534
8	84	760	1210	1449	83.505
9	84	760	1032	1206	85.572
10	88	760	1013	1259	80.460
Total	844		8007	9525	84.063

⁴ Very little files have bags in its structure therefore we took 844 files.

⁵ We did not use 10-Fold Cross-Validation. Simply we segmented the corpus to 10 parts (fold), One part as test and the other 9 parts as training.

6. CONSTRUCTION NEW DICTIONARY FROM THE CORPUS

Clearly, that using of annotated corpus with Lesk algorithm will increase the accuracy. But, practically, training need very long time for evaluating the terms. Certainly, it is used for first time but the performance can be improved by using a new dictionary from old dictionary which have all words (assist in deciding for solving WSD problem) for all senses. The word's tf and isf are recorded in the new dictionary with the word itself. Any word have zero isf will not be recorded. The algorithm for construction the dictionary is shown in figure 2. It will construct new dictionary automatically (we don't have query here).

By applying the algorithm, each word recorded in more than one sense will have the same isf and this is logically because $isf = \log(S/n_i)$. As mentioned below, this is very useful in practice.

tf for each word were recorded without using query and, logically, it will not affect the results when we will test this dictionary.

Algorithm for construction dictionary

- 1- start from(initialize the) existing dictionary "NKJP_WSI.xml"
- 2- Take one entry (ambiguous word w) from Dictionary
- 3- Get the senses and all words of the examples for each sense. Now, each sense has words list.
- 4- for the entry w : (i)search for this entry in the corpus (ii) all words of the sentence in which this entry appear are recorded in the sense words list of the matched sense.
- 5- For this entry of dictionary now we have list of senses where each sense have a list of words taken from the dictionary examples and the corpus. Compute tf and isf for each word where the 'documents' is the senses.
- 6- Delete each word have $isf=0$.
- 7- Record the words which are not deleted in new dictionary with its tf and isf .
- 8- Repeat 2-7 for all entries in the dictionary

Figure 2: Construction Dictionary from existing dictionary with assistant of annotated corpus

7. TESTING THE MODIFIED DICTIONARY

In practice we have query, sentence which have the ambiguous word w , but we don't have isf for each word in this sentence. This done easily by looking to the word if it is exist in any sense it will take the isf of the same word in the sense otherwise it's $isf=0$ (this because each word which recorded in more than one sense, it have same isf).

Computing tf for w in the query are done by comparing all the words in query with the set of all words in query without regards to the set of the words in document.

We used the same data set in section 5 for testing the new dictionary. The same two conditions, with/without taking stop words in computation, were taken.

We got on the results identical to the results in table 1 & 2. The important gain here is the time of execution where the time of execution drop down by 20 times comparing with the first tests.

8. ANALYZING THE RESULTS AND CONCLUSION

This paper presents that how to apply Lesk algorithm using dictionary with assistant of annotated corpus. Moreover, it presents how the similarity equation of IR is modified in order to apply it on WSD. Then, the performance (especially the time of execution) was improved by constructing a new dictionary from the annotated corpus and the used dictionary. The last modification is for practical use.

Firstly, all words in annotated corpus which related to the specific sense were obtained. Then, the set of all words for this sense and the other senses in same orthogonal entry where obtained which leads to words randomly ordered (or by frequency distribution). Finally, each word and its *tf* and *isf* are also obtained. This means statistical information about words is generated which shared in all senses for specific orthogonal entry.

We induced that, from practical test, there are a key words (limited words) which can be used to solve the WSD. The other words were used as assistant or weren't used in decision. As example, most of preposition weren't used for making the decision i.e. they can be taken as stop words (deleted). As mentioned, deleting the stop words increased the average accuracy on the tests. the other thing was induced from this work, using dictionary alone for solving WSD problem not sufficient because some words depends on previous words not in the same sentence but may be in the previous sentences.

Anyone who has simple information about polish language, he/she knows that the base words in polish take much morphosyntatic forms. This will lead to collections of many words for the same base. From this point of view, this paper suggests writing dictionary depends on the bases. Furthermore, this work must be done carefully because it (maybe) will cause drop the accuracy. It need to more works and experiments for deciding if it can be used.

Because there are key words helped the algorithm in making decisions in WSD, searching about these words is important task. This work is purely related to linguistics scientists.

REFERENCES

- [1] D. Jurafsky & J. Martin 2008 "Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition" PEARSON prentice Hall Upper Saddle River, New Jersey, USA.
- [2] N. Ide and J. Véronis (1998). "Word Sense Disambiguation: The State of the Art" Computational Linguistics, Vol. 24 Issue 1, pp 2-40.
- [3] H. Ng (1997) "Exemplar-Base Word Sense Disambiguation: Some Recent Improvements". In Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP.
- [4] A. Montoyo, A. Suarez, G. Rigau and M. Palomar (2005): "Combining Knowledge- and Corpus-based Word-Sense-Disambiguation Methods". Journal of Artificial Intelligence Research Vol. 23, PP 299-330.
- [5] Y. Ledo-Mezquita, G. Sidorov and V. Cubells, (2006) "Combined Lesk-based Method for Words Senses Disambiguation," In Proceedings of the 15 th International Conference on Computing, Washington: IEEE Computer Society.
- [6] P. Basile, M. de Gemmis, A Gentile, P. Lops, and G. Semeraro. 2007. "UNIBA:JIGSAW algorithm for Word Sense Disambiguation". In Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic.
- [7] S. Ponzetto and R. Navigli. 2010. "Knowledge-rich Word Sense Disambiguation Rivaling Supervised Systems". In Proceedings of the 48th Annual Meeting of the Association for Computational linguistics, ACL '10, Stroudsburg, PA, USA.
- [8] F. Viveros-Jiménez, A. Gelbukh and G. Sidorov (2013) "Simple Window Selection Strategies for the Simplified Lesk Algorithm for Word Sense Disambiguation".. Proceeding of 12th Mexican International Conference on Artificial Intelligence, MICAI 2013, Mexico City, Mexico.

- [9] D. Schwab, A. Tchechmedjiev, J. Goulian, M. Nasiruddin, G. Serasset, and H. Blanchon 2013. "GETALP System : Propagation of a Lesk Measure through an Ant Colony Algorithm". Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, Georgia, USA.
- [10] P. Basile, A. Caputo and G. Semeraro (2014) "An Enhanced Lesk Word Sense Disambiguation Algorithm through a Distributional Semantic Model". Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics. Dublin, Ireland.
- [11] R. Nvigli (2009) "Word Sense Disambiguation: A Survey" ACM Computing. Survey. Vol. 41, No.2, pp 1-69.
- [12] S. Torres and A. Gelbukh (2009). "Comparing Similarity Measures for Original WSD Lesk Algorithm" Research in Computing Science 43, pp. 155-166
- [13] A. Kilgarriff and J. Rosenzweig (2000). "Framework and results for English SENSEVAL." Computers and the Humanities, Vol. 34, pp 15- 48.
- [14] A. Przepiórkowskiego, M Bańko, R. Górskiego B.Tomaszczyk (2012) "Narodowy Korpus Jenzka Polskiego" Wydawnictwo Naukowe PWN SA, Warsaw, Poland.

AUTHORS

Dr. Ahmed H. Aliwy is a lecturer in Computer Science Department in University of Technology, Baghdad, Iraq. His current research interests include Arabic Natural Language Processing ANLP, Machine Learning and Natural Language Processing. He has got his PhD from Warsaw University, Poland.



Dr. Ayad R. Abbas is a lecturer in Computer Science Department in University of Technology, Baghdad, Iraq. His current research interests include AI, Machine Learning and some applications of Natural Language Processing. He has got his PhD from University of Wuhan, China.



INTENTIONAL BLANK

MYANMAR WEB PAGES CRAWLER

Su Mon Khine and Yadana Thein

University of Computer Studies, Yangon

sumon5.8.1986@gmail.com, yadana@ucsy.edu.mm

ABSTRACT

Nowadays web pages are implemented in various kinds of languages on Web and web crawlers are important for search engine. Language specific crawlers are crawlers that traverse and collect the relative web pages using the successive URLs of web page. There is very little research area in crawling for Myanmar Language web sites. Most of the language specific crawlers are based on n-gram character sequences which require training documents, the proposed crawler differ from those crawlers. The proposed system focused on only part of crawler to search and retrieve Myanmar web pages for Myanmar Language search engine. The proposed crawler detects the Myanmar character and rule based syllable threshold is used to judgment the relevant of the pages. According to experimental results, the proposed crawler has better performance, achieves successful accuracy and storage space for search engines are lesser since it only crawls the relevant documents for Myanmar web sites.

KEYWORDS

Language specific crawler, Myanmar Language, rule base syllable segmentation.

1. INTRODUCTION

The Internet provides valuable resource of all types and web area is grown exponentially day by day. Web pages are added by different site holders every times. Gathering the web pages manually for language specific search engine is not possible and realistic. Therefore search engine mainly rely on crawlers to create and maintain indices for the web pages. Web crawlers are short software codes also called wanderers, automatic indexers, Web robots, Web spiders, ants, bots, Web scatters [2]. To collect the set Myanmar Web pages for search engine, crawlers, which traverses Web by following the hyperlinks and stored the download pages in a repository and used then by indexer component to index the web pages, are needed.

In comparison to general purpose crawlers which traverse all the pages on Web, language specific crawlers are collected only for specific languages on Web. Most of the language specific crawlers were implemented using n-gram character sequences to detect language, encoding schemes and scripts of training corpus, which is the basic method for text categorization and required trained documents in prior to classify language of web pages.[7] Some researchers detected language of web pages on Urls of top domain. Eda BayKan, Monka Henzinger, Ingmar Weber [5]determined the language of web pages using its URL of the country code of the top level domain by using machine learning classifiers such as Naïve Bayes, Decision Tree, Relative Entropy, Maximum Entropy and experimented English, German, French, Spanish and Italian Languages. Takayuki Tamura, Kulwadee Somboonviwat and Masaru Kitsuregawa [8] identified language of the Thai web pages by content type of HTML META tag firstly. If the content types are missed, checked then the content of web pages based on TextCat, a language guesser based on n-gram statistics.

Myanmar web pages can't detect exactly language of web pages by checking the character set of HTML META tags since most of the web sites developers are not definitely identified for Myanmar character set in META tag. Furthermore, web pages can't identified its languages by using Urls of top domain since Myanmar languages web pages are mostly distributed on other top level domain such as .com, .info, .net rather than .mm which is refer to Myanmar country. Therefore this proposed system relies on content of web pages for crawling in order to download the Myanmar web pages and the judgment of relevancy is easily determined by proposed rule based syllable percentage threshold. The crawling process in this system is based on crawler4j[1] and extends the crawler to collect only Myanmar web pages for further process of web search engine for Myanmar Language.

This paper is organized into seven sections. Literature reviews are discussed in the next section2. Section 3 describes various types of crawlers and some open source general web crawlers. Myanmar scripts, fonts and encoding on web are explained in Section 4. Section 5 describes the proposed crawler. Experimental results will discuss in section 6 and proposed system will be concluded in section 7.

2. LITERATURE REVIEWS

In this section, the topics related to this proposed crawler are discussed. AltaVista search engine introduced a crawling module named as Mercator [4], which was scalable, for searching the entire Web and extensible. Olena Medelyan, Stefan Schulz, Jan Paetzold, Michael Poprar, Kornel Marko , [6] they used n-gram model for text categorization tool based on content of web pages using standard crawler Nutch and checked the domain of web pages with training documents collections. Dr Rajender Nath and Khyati Chopra [2] discussed about the Web Crawlers: Taxonomy, Issues & Challenges. They classified the crawlers according to coverage area, mobility, topic –domain and load distribution to Unfocused and Focused Crawler , Mobility Based Crawler , Domain specific crawler and Based on Load Intra and Distributed Crawler respectively. They also discussed issues of Crawlers. Crawler used in this paper is related to Domain (Specific) crawler because it does not need to collect the entire Web, but need to retrieve and collect only Myanmar Web pages. Finally, the relevance of the web page is determined by rule based the syllable percentage threshold.

3. VARIOUS TYPES OF CRAWLERS AND SOME OPEN SOURCE CRAWLERS

General web crawlers are designed to download as many resources as possible from a particular web site. Trupti V. Udupure1, Ravindra D. Kale, Rajesh C. Dharmik [3] are discussed four different types of web crawlers: **(1) Focused web crawler** : Focused Crawler is the crawler that tries to download the pages which are related to a specific and relevant of a topic that users interest. **(2) Incremental crawler**: In order to refresh the download pages, crawlers replaces the old documents with newly downloaded documents frequently based on the estimate of how often pages changes. **(3) Distributed crawler**: Different crawlers are working in distributed form in order to download the most coverage of the web, in which central crawler manages all other distributed crawlers. **(4) Parallel Crawler** : Many crawlers runs in parallel and a parallel crawler consists of multiple crawling process and it may be local or distributed at geographically distant location. In addition to another, some of the general open source web crawlers that are widely used today are also listed in table 1.

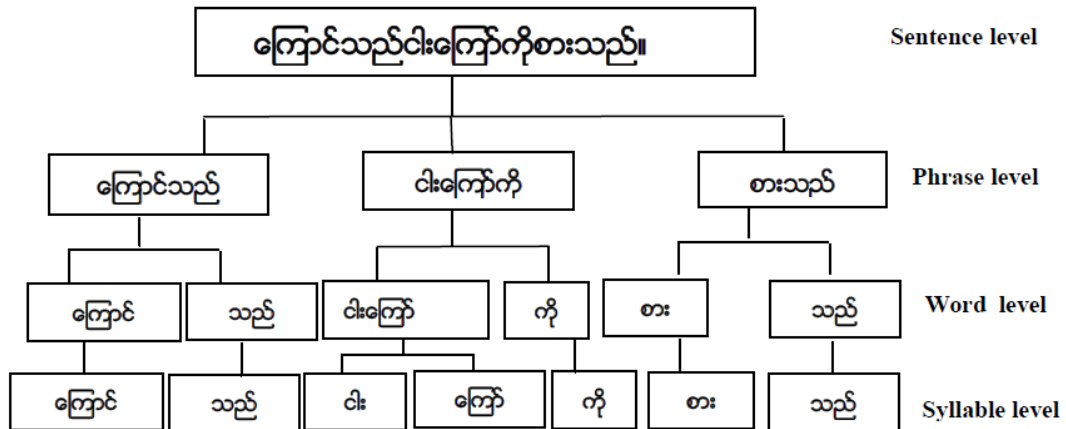


Figure1. Structure of Myanmar Sentence

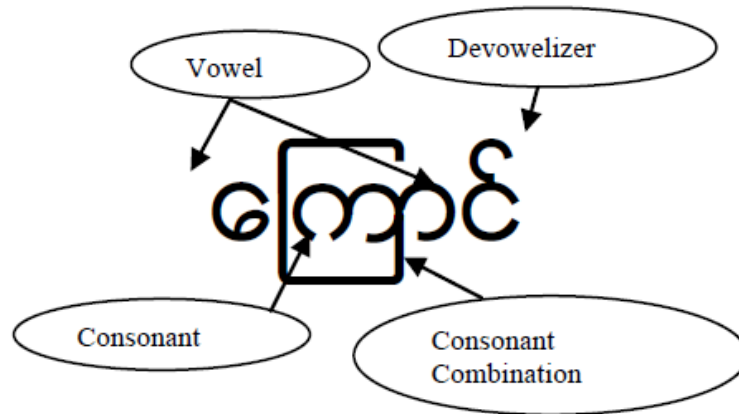


Figure2. Structure of Myanmar Syllable

4.1 Different fonts and encoding system for Myanmar Web sites.

The first generation of Myanmar encoding systems were ASCII code in which Latin English glyphs were replaced by the Myanmar script glyphs to render the Myanmar scripts which was no standardization of encoding characters. Firstly, Myanmar script was added to Unicode Consortium in 1999 as version 3.0 and improved Unicode5.1 in 2008 and Myanmar3, Padauk and Parabaik are in the range of U+1000 to U+109F. And then, various fonts such as Myazedi, Zawgyi_One have been created. Although Zawgyi_One is not Unicode standard, over 90% of Web sites use Zawgyi_One font, which are Although Unicode stores text in only one order and render correctly and Zawgyi_One can store text in several ways but superficially appear correct. Therefore, the proposed crawler converts all fonts to Zawgyi_One fonts and normalizes various writing style to one standard style. For example, user can write ' ' or ' ' after writing consonant ' ' for syllable ' ' that is equivalent to 'Ko' in English. Table 3 shows different encoding sequences of Unicode and Zawgyi_One and Table 4 shows some examples of normalization of Zawgyi_One character.

Table 3. Sequence style of using Unicode and Zawgyi_One for Myanmar Syllable

Fonts	Sequence Style
Unicode	က + ဝိ + ျ = ကိ
	က + ျ + ဝိ = ကိ
Zawgyi-One	က + ° + ျ = ကိ
	က + ျ + ° = ကိ

Table 4. Normalization of Zawgyi_One character sequences.

Various forms of writing sequence	Normalize sequence
ကိ, ကိ	ကိ
ကိ, ကိ	ကိ
ကိ, ကိ	ကိ
ကိ, ကိ, ကိ	ကိ
ကိ, ကိ, ကိ	ကိ
ကိ, ကိ, ကိ, ကိ	ကိ
.....
ကိ, ကိ, ကိ, ကိ	ကိ

5. SYSTEM ARCHITECTURE FOR PROPOSED CRAWLER

The proposed crawler traverses identified famous Myanmar web sites seeds URLs systematically, it identifies all Urls containing in that page and adds them to the frontier, which contains the list of unvisited URLs. URLs from the frontier are visited one by one, fetch the web pages and parse the pages to parser to remove HTML tags in order to check Myanmar character. The proposed crawler normalizes various fonts to Zawgyi_One font since Zawgyi_One is mainly dominant font on Web pages. After normalization, the proposed crawler calculates the syllable threshold based on rule base syllable identification in order to judgment the relevant of the pages. If the web pages are relevant, store them in the pages repository in order to ready for indexer to extract the keywords of web pages. The process is repeated until the crawling process reach the specified depth of the crawler after starting from the specified seeds URLs .Figure3 shows the design of proposed crawler and the process flow of proposed crawler can be summarized in figure4.

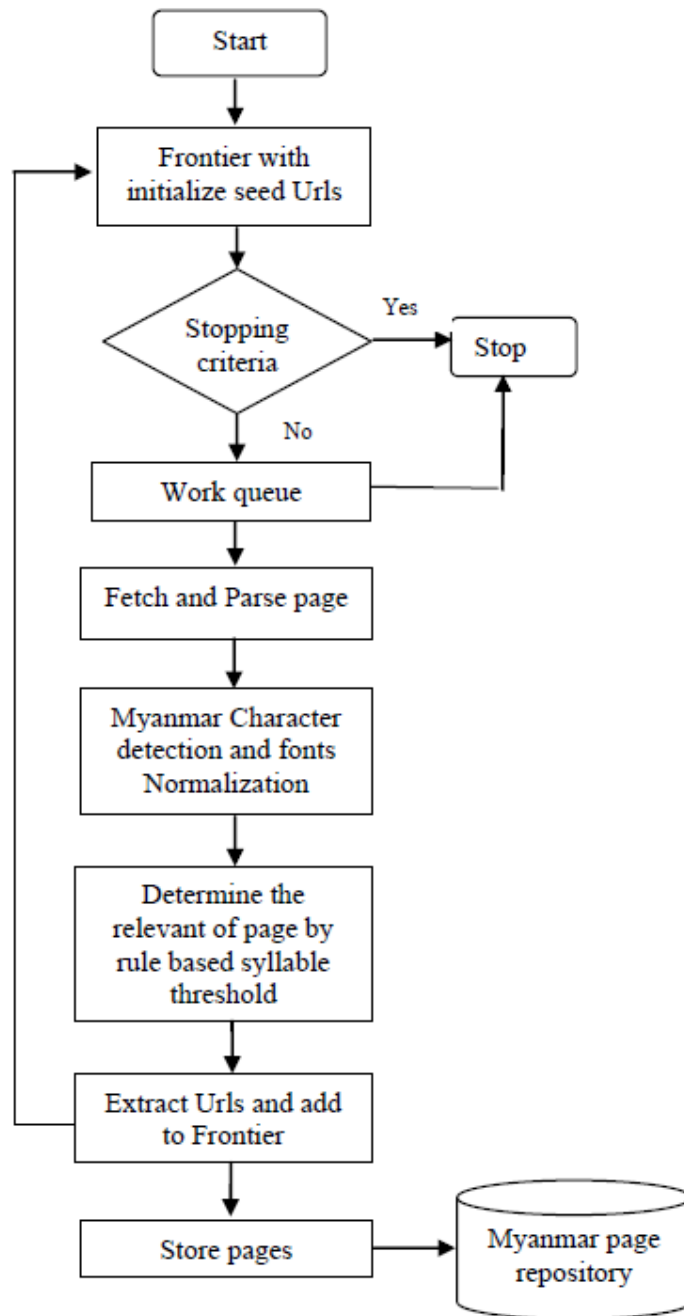


Figure3. The design of proposed crawler

1. Myanmar Languages web sites urls are put to the crawler as seed URLs.
 2. The system check for stopping criteria.
 3. If not reach the specified criteria, add URLs to the Work Queue.
 4. Pick the URL up from the Work Queue.
 5. Web pages are fetched and passed to parser in order to extract the content.
 6. Myanmar characters are detected in the range between the decimal values of 4096 to 4255 defined by Unicode Consortium.
 7. Various fonts are normalized to Zawgyi_One font.
 8. The relevant of Myanmar Web pages are identified by proposed rule base syllable threshold.
 9. Extract the Urls, add them to Frontier and pages are stored in repository.
 7. Otherwise, discard the web pages.
- Go to Step 2 and repeat when the specified depth is reached.

Figure4. Process flow of proposed crawler.

5.1. Proposed rule based syllable segmentation

After detecting Myanmar characters and normalization to one standard font, the system segmented Myanmar sentences into syllable by proposed rule based syllable segmentation methods and calculate thresholds in order to identify the relevancy of Myanmar Pages since Myanmar Web pages are missed other languages .Proposed rule based syllable segmentation method is shown in figure 5. In here, the crawler are not considered the spelling checking of syllable since the proposed crawler only segmented content of web pages.

1. If we found one consonant and next character is not '၀' or any consonant then take one syllable by combining the rest of characters until we found any consonants or '၀' or '၂'
2. If starting character is '၀' or '၂' and next character is consonants, take one syllable by combining the rest of character until we found another consonants or '၀' or '၂'
3. If first character is '၀' and second character is '၂' and next character is consonant , take one syllable by combining the rest of characters until we found another consonant or '၀' or '၂'

Figure5. Proposed rule based syllable segmentation method

Most of the Myanmar Web sites are not written only Myanmar Language and they are missed to other languages. For the combination of Myanmar and other language documents, Myanmar content exceed the predefined syllable threshold will be considered as relevant of Myanmar Web pages and stored them into page repository in order to further study of word segmentation and below the threshold will be discard as a non relevant pages to save storage space on disk. Threshold percentage is calculated the ratio of Myanmar Syllable count to the total numbers of

Myanmar Syllable and other characters contained in that web pages. Figure 6 and 7 shows some example of web pages combined with other languages such as English Languages.



Figure 6 Greater threshold of Myanmar Syllable to other language

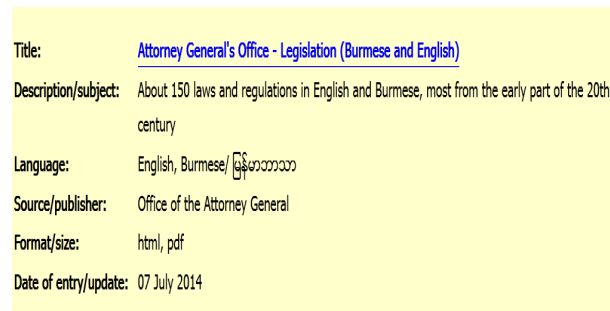


Figure7. Fewer threshold of Myanmar Syllable to other language

According to figure 6, the proposed crawler obtain 82% of Myanmar syllable percentage threshold to other language character and figure 7 obtain 2.5% of Myanmar syllable to other languages such as English language as shown in figure. The proposed crawler will regard figure 6 as relevant pages and stored in pages repository and figure 7 will be discard in order to reduce storage space in repository when syllable threshold is set to 3%. In this proposed crawler, users can easily define syllable percentage threshold depends on how much percentage of Myanmar language web pages to other language they want. It can easily to define and scalable.

6. EXPERIMENTAL RESULTS

6.1 Performance Evaluation

The evaluation methodology commonly and widely used in information retrieval is to calculate the precision. In the language specific crawling prospective, precision represents the ratio of the number of language relevant documents to the total number downloaded documents. Precision also called “harvest rate” in equation 1 is used for major performance metric for language specific crawler community.

$$\text{Precision (Harvest rate)} = \frac{\text{Language relevant pages}}{\text{Total download pages}} \tag{1}$$

6.2 Crawling experiment

In this section, the proposed crawler presents the result of experiment of crawler. The proposed crawler started with 11 Myanmar web site seeds URLs shown in Table 5, which are popular Myanmar Web sites. The crawler runs two times with 32 bit operating system, 4GB memory with different internet downloads speed at day and night respectively. The first run of the crawler at 9: AM to 2: PM can download a set of only 8960 Html Myanmar documents with the depth of crawler is set to 7 and Myanmar syllable threshold to 4% .The second run of the crawler at 1: AM to 5: AM resulted in 12582 HTML documents with the depth of crawler is set to 10 and Myanmar syllable threshold to 3%. In total, 21812 documents were collected in this system and the results are shown in table 6. The result shown that fewer percent of syllable threshold can download more documents and greater percent of syllable threshold can download fewer documents.

Table 5. Myanmar web site seeds Urls

No	Urls	Description
1	http://www.president-office.gov.mm/	Information
2	http://my.wikipedia.org/wiki/.mm	Information
3	http://www.thithtolwin.com	News
4	http://www.7days.com	News
5	http://www.myanmarwebdesigner.com/blog/	Technology
6	http://winthumon.blogspot.com/2010/03/valueable-words.html	Literature
7	http://www.rfa.org/burmese/	News
8	http://hivinfo4mm.org/category/myanmar/	Health
9	http://www.myanmar-network.net/	Education
10	http://www.oppositeeyes.info/	Politics
11	http://burmese.dvb.no/dvblive	News

Table 6. Different runs of crawler

	depth of crawler	syllable threshold	no of page collected
First run	7	4	8960
Second run	10	3	12582
Total			21542

It is a little difficult to calculate the precision of all download documents manually, the proposed crawler only calculates for first 1300 pages of each run. For the first run of crawler, by manually checking the relevancy of Myanmar pages ,1289 pages of 1300 are correctly download as Myanmar web pages and only 24 pages are download incorrectly and the precision was 98.15% . For the second run of the crawler, 1294 pages of 1300 are correctly download as Myanmar web pages and only 15 pages are download incorrectly and the precision was 98.84%. The experiments also evaluated that the proposed crawler outperformed n-gram based language identification which require sufficient training corpus for different fonts and encoding. The proposed crawler is not necessary training corpus and easily identify as Myanmar Language web

site. Table 7 shows the average percentage of precision for proposed crawler and ngram-based crawler were 98.49 % and 96.6% respectively.

Table 7. Precision of the proposed crawler and n-gram based crawler

	Proposed crawler		N-gram Based Crawler	
	First run	Second run	First run	Second run
Correctly download as Myanmar pages	1289	1294	1192	1268
Incorrectly download as Myanmar pages	11	6	108	32
No of pages	1300	1300	1300	1300
Accuracy	99.15%	99.53%	91.69	97.54
Average Accuracy	99.34%		94.6%	

The crawler analyzed what kinds of top level domains are influenced on Myanmar Web sites. The average percentage of top level domain for Myanmar web sites in which the crawler downloaded are.com 83%, .mm 7% .org 5.2%, .net 3.24%, .info 0.92% and other for 0.56 respectively are shown in figure 8.

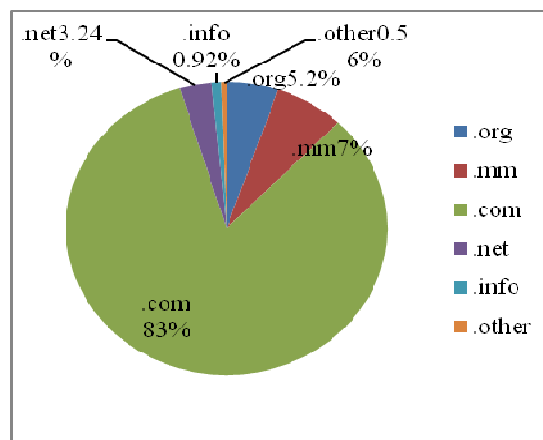


Figure8. Influence of different domains on Myanmar web sites.

The crawler also analyzed which fonts are mostly used for Myanmar web sites for each domain. Among them, Zawgyi_One is the widely used by web developer and Myanmar3 is the secondly used on Myanmar web site especially on governmental sites. Win Innwa is the thirdly used and the most rarely fonts is Padauk on Myanmar web sites according to result. Table 8 shows the fonts usage for each domain and Figure 9 shows bar chat representation for each font on each domain.

Table 8. Various fonts for each domain

	Zawgyi_One	Win Innwa	Myanmar3	Padauk	Total
.com	82.3	7.0	9.0	1.7	100%
.mm	76.5	2.0	20.0	1.5	100%
.org	87.4	4.0	7.6	1.0	100%
.net	86.0	4.0	8.3	0.7	100%
.info	92.7	4	3	0.3	100%
other	92.9	2	5	0.1	100%

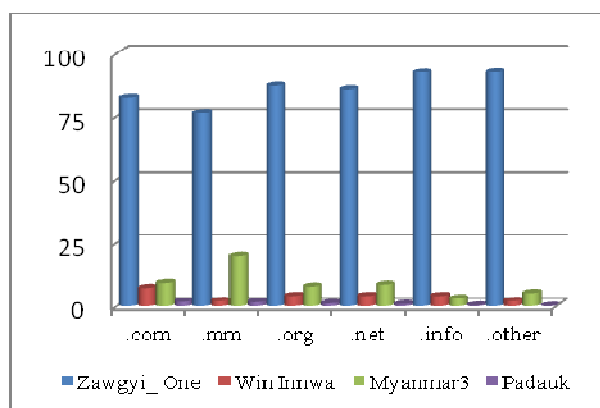


Figure.9 various fonts for each domain.

7. CONCLUSION

This system proposed language specific crawler in order to retrieve and download the Myanmar web pages for the supporting of web search engine for Myanmar Language. Myanmar characters of Web pages are detected and the relevance judgment of the web pages is determined by the proposed rule based syllable percentage threshold. This crawler can easily adjust the Myanmar syllable threshold in order to judgment the relevant of the pages. The proposed crawler can download various fonts written in web pages. This crawler also analyzes the various kinds of domain in Myanmar Language web sites and different fonts types for each domain. According to statistic, Zawgyi_One is the mostly influence in web pages and other fonts are fewer used on web pages. The proposed system is implemented in java language that is easy to install, develop and crawling speed is very high. The proposed crawler will improve the efficiency of language specific crawling for Myanmar Language in the future.

REFERENCES

- [1] <http://code.google.com/p/crawler4j/>
- [2] Dr Rajender Nath and Khyati Chopra, (2013)"Web Crawlers: Taxonomy, Issues & Challenges", International Journal of Advanced Research in Computer Science and Software Engineering. Volume 3, Issue 4.
- [3] Trupti V. Udupure, Ravindra D. Kale2, Rajesh C.Dharmik3, (2014)"Study of Web Crawler and its Different Types ", IOSR Journal of Computer Engineering. Volume 16, Issue 1, Ver VI ,PP01-05.
- [4] Allan Heydon and Marc Najork , "Mercator: A Scalable, Extensible Web Crawler".
- [5] Eda BayKan, Monka Henzinger, Ingmar Weber, (2008) "Web Pages Language identification based on URLs" .PVLDB '08 , Auchkand, New Zealand.

- [6] Olena Medelyan, Stefan Schulz, Jan Paetzold, Michael Poprar, Kornel Marko "Language Specific and Topic Focused Web Crawling".
- [7] Tomas OLVECKY (2005) "N-gram Based Statistics Aimed at Language Identification", M.Bielikova (Ed). IIT.SRC 2005, pp. 1-7.
- [8] Takayuki Tamura, Kulwadee Somboonviwat and Masaru Kitsuregawa , (2007)"A Method for Language – Specific Web Crawling and Its Evaluation" , Systems and Computers in Japan , Vol.38, No.2.
- [9] Myanmar –English dictionary Department of the Myanmar Language Commission (2011).

AUTHORS

Su Mon Khine received M.C.Sc and B.C. Sc, in Computer Science, from University of Computer Studies, Yangon. She is now PhD candidate in Information and Technology and currently doing research at University of Computer Studies, Yangon. Her research interest includes web crawling, information retrieval, natural language processing and data mining.



Dr. Yadana Thein is now working as an Associate Professor in University of Computer Studies, Yangon (UCSY) under Ministry of Science and Technology, Myanmar. She is particularly interested in Optical Character Recognition, Speech Processing and Networking. She published about 30 papers in workshops, conferences and journals. Currently, she teaches networking subject to under-graduate and post-graduate students. She supervises Master thesis and PhD research candidates in the areas of Image Processing.



IMPROVING A JAPANESE-SPANISH MACHINE TRANSLATION SYSTEM USING WIKIPEDIA MEDICAL ARTICLES

Jessica C. Ramírez^{1,2}, Yuji Matsumoto² and Darwin Muñoz¹

¹Universidad Iberoamericana, UNIBE, Santo Domingo, Dominican Republic
j.ramirez1@unibe.edu.do, d.munoz@unibe.edu.do

²Information Science, Nara Institute of Science and Technology, Nara, Japan
matsu@is.naist.jp

ABSTRACT

The quality, length and coverage of a parallel corpus are fundamental features in the performance of a Statistical Machine Translation System (SMT). For some pair of languages there is a considerable lack of resources suitable for Natural Language Processing tasks. This paper introduces a technique for extracting medical information from the Wikipedia page. Using a medical ontological dictionary and then we evaluate on a Japanese-Spanish SMT system. The study shows an increment in the BLEU score.

KEYWORDS

Comparable Corpora, Dictionary, Ontology, Machine Translation

1. INTRODUCTION

The quality, length and coverage of a parallel corpus are fundamental features in the performance of any Statistical Machine Translation (SMT) System. For some pair of languages there are a lack of aligned resources suitable for Natural Language Processing (NLP) tasks.

The use of automatic and semi-automatic methods for constructing resources along with manual resources help to reduce both the cost and time of any NLP project. For this reason many approaches have been published for constructing resources such as dictionaries, thesauri and ontologies, in order to facilitate NLP tasks such as word sense disambiguation, machine translation and other tasks [4]. [1] explore the multilingual features of Wikipedia for automatically extract sentences across multiple languages and [2] use Wikipedia for extracting Name Entities.

This study we use Wikipedia for extracting medical information from the health related articles in Japanese and Spanish, to construct a Medical Ontological dictionary, aligned the sentences in those articles and then we evaluate it impact on a Japanese-Spanish SMT system.

2. RESOURCES

Wikipedia is an online multilingual encyclopedia with articles on a wide range of topics, in which the texts are aligned across different languages. Wikipedia have articles aligned in Spanish and Japanese. Wikipedia has some features that make it suitable for research such as:

Each article has a title, with a unique ID. “Redirect pages” handle synonyms, and “disambiguation pages” are used when a word has several senses. “Category pages” contain a list of words that share the same semantic category. For example the category page for “Birds” contains links to articles like “parrot”, “penguin”, etc. Categories are assigned manually by users and therefore not all pages have a category label.

The information in redirect pages, disambiguation pages and Category pages combines to form a kind of Wikipedia taxonomy, where entries are identified by semantic category and word sense.

3. GENERAL DESCRIPTION

The general goal is to extract useful in domain data from Wikipedia to improve the performance of a Japanese-Spanish SMT system. The study is divided 3 phases: The first one the construction of the Japanese-Spanish ontological dictionary, then Japanese-Spanish parallel and then evaluate the corpus in a SMT system.

Phase 1. Ontology Medical Dictionary

The goal is the creation of a Spanish-Japanese ontology, in which, we align each medical article in Spanish and Japanese and then, we extract all the terms related to the article title. And then by using Pattern Recognition techniques. We extract information associated to the given word, for example: Kidney Stone, disease related to kidney, symptoms, causes, etc.

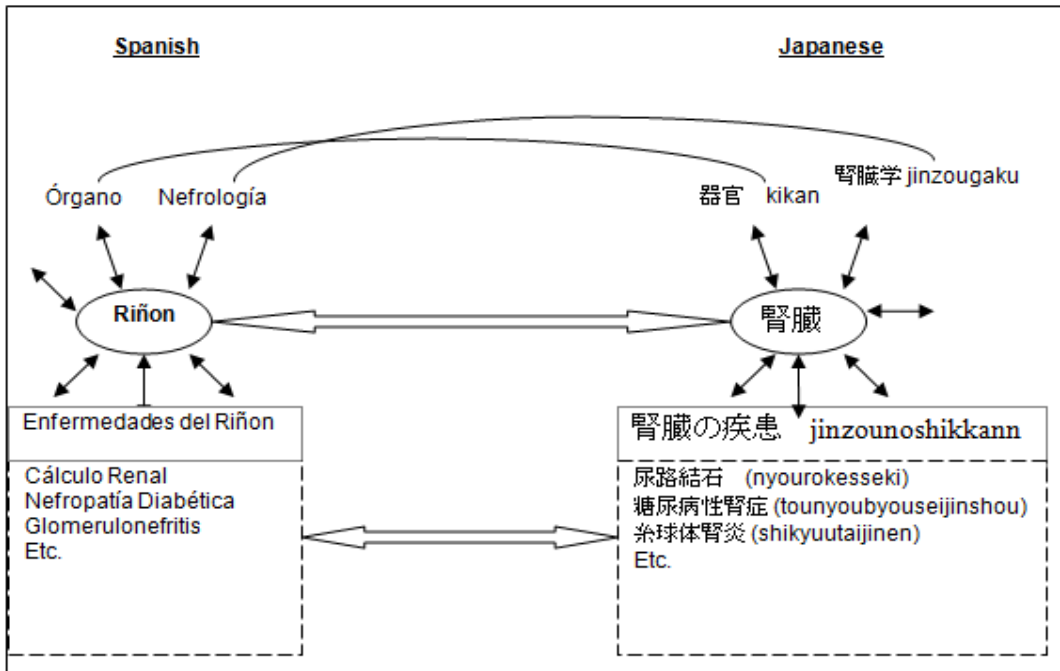


Figure 1 : General Structure of the Spanish-Japanese alignment.

Ex. In figure 1 [6] shows the structure of the system, for example, the word ‘Kidney’, Spanish is “Riñon”, is translated to Japanese a 腎臓, ‘yinzō’, which is associated with all the diseases related to kidney such as ‘Kidney disease and a list of the disease such as: ‘Kidney stone’, ‘Glomerulonephritis’, etc.

Methodology

Extracting The Medical articles from Wikipedia

The goal is acquisition of Spanish-Japanese medical domain of Wikipedia’s article titles. Each Wikipedia article provides links to corresponding articles in different languages.

Every article page in Wikipedia has on the left hand side some boxes labelled: ‘navigation’, ‘search’, ‘toolbox’ and finally ‘in other languages’. This has a list of all the languages available for that article, although the articles in different languages do not all have exactly the same contents.

To extract the medical articles we extract them by mean of their categories, mining all articles that belong to categories such as: “medicine”, “disease”, “organ”, etc.

Pre-processing Procedure

We eliminated all irrelevant information of each article in Spanish such as tables, special characters, menus, etc.

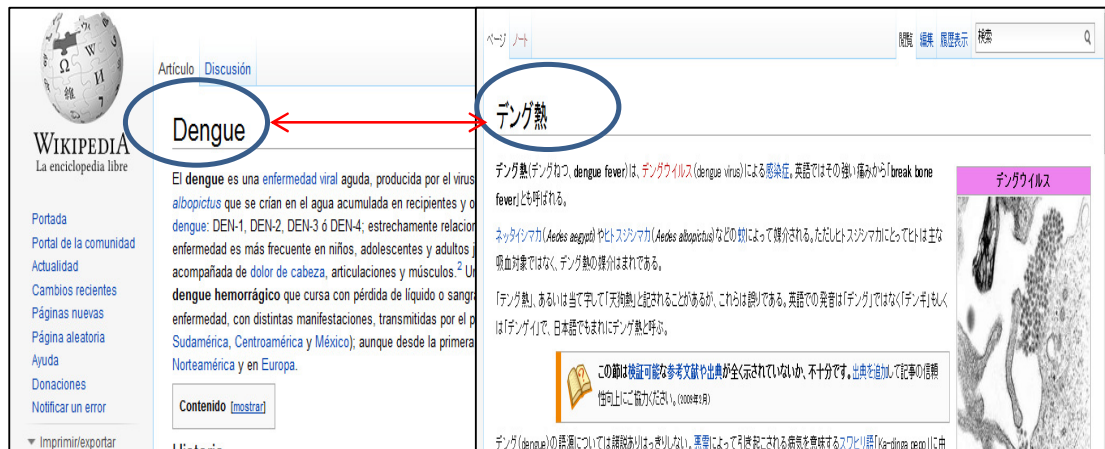


Figure 2 :Wikipedia links.

Dictionary Spanish-Japanese

Take all article titles that are nouns or named entities and look in the articles’ contents for the box called ‘In other languages’. Verify that it has at least one link. If the box exists, it links to the same article in other languages. Extract the titles in these other languages and align them with the original article title.

For instance Figure 2 shows the Spanish article titled “Dengue” (Dengue fever), which is translated into Japanese as “デング熱”(Dengu Netsu). When we click Spanish or Japanese ‘in

other languages' box we obtain an article about the same topic in the other language. This gives us the translation and we proceed to extract it.

Ontological Relation between terms

The goal is extract all features related to a given disease such as: symptoms, causes, organs, etc. By using Pattern Recognition we extract the sentences associated with the inner titles in Wikipedia. Ex. The inner title symptom in Spanish and Japanese, and extract the phrases and translated the nouns if there are hyperlinks and belong to the extracted Wikipedia dictionary. And we proceed to aligned them in both languages.

The long term goal pursue when we type a symptom like "headache", display all the diseases that that contain headaches in the list of symptoms, in case there add another symptoms like "fatigue" continue pruning the list with possible disease.

Phase 2. Constructing a Parallel Corpus

The goal is the creation of a parallel corpus by aligning the sentences of the medical articles. We use a extended form of a ruled-based approach similar to [5]. We extended the amount of rules and eliminate some rules that were redundant or cause ambiguity between rules.

Methodology

We eliminate the irrelevant information from Wikipedia articles, to make processing easy and faster.

The steps are as follows.

1. Remove from the pages all irrelevant information, such as images, menus, characters such as: "()", """, "*", etc...
2. Verify if a link is a redirected article and extract the original article
3. Remove all stopwords -general words that do not give information about a specific topic such as "the", "between", "on", et

For splitting the sentences in the Spanish articles we used NLTK toolkit¹, which is a well-known platform for building Python scripts.

For tag Spanish sentences, we used FreeLing², which an open source suit for language analyzer, specialized in Spanish language.

For Splitting into sentences, in to words and add a word category, we used MeCab³, which is a Part-of-Speech and Morphological Analyser for Japanese.

	Rule Description
Rules	Japanese=> Spanish
Noun	Noun+desu => noun

¹ <http://nltk.org/>

² <http://nlp.lsi.upc.edu/freeling/>

³ <http://cl.naist.jp/~eric-n/ubuntu-nlp/dists/hardy/japanese/>

Name Entity	NE=>NE (Capital letter)
Adjective	Adj (fe/male) =>Adj (NA/I)
Question	(sentence+?)=>(¿ + sentence +?)
Pronouns	Pron =>Pron

Table 1. shows some of the rules applied to this work. Those rules are created taking in account the morphological and syntactic characteristic of each language.

Phase 3. Using Medical Corpus in to SMT System

The main goal is to measure feasibility of using an in-domain corpus (in this case health related) in a SMT system.

We used the aligned parallel sentences extracted in phase 2 to measure its impact in a Statistical Japanese-Spanish MT system.

Experiments

We use a random sample of 500 parallel sentences extracted from Wikipedia and we add to 50k Japanese-Spanish parallel corpus from Europarl. We used human translators to translate into Japanese the 50k sentences because the Europarl corpus just contains the proceedings of the European Parliament for countries that belongs to the European Union.

We train a baseline SMT system with the 50k sentences. Then we performed experiments adding the 500 sentences extracted on phase 2 to the baseline. In both cases we used as a language model Wikipedia Spanish articles, 10k for development set, 10k for test set and 30k for training.

4. RESULTS AND DISCUSSION

Table 2 shows the results using the Europarl data and the result by adding the the medical corpus. Using the Medical corpus increase the BLEU⁴ score. However, If in the training set there is not health related sentences the BLEU score do not increase.

Corpus	BLEU
EuroParl	27.87%
EuroPal + Medical corpus	28.15%

Table 2. Results

5. CONCLUSIONS

This paper focuses on extracting medical information from Wikipedia and the creation of an ontology in Spanish and Japanese. In domain corpus can be used to improve the performance of a SMT system.

We will extend this work by using several corpus of other field, like economy, sociology and so on.

ACKNOWLEDGEMENTS

⁴ Bilingual Evaluation Understudy

This research was supported by is « Fondo Nacional de Innovación y Desarrollo Científico y Tecnológico » **FONDOCyT#2012-2013-3A2-59**, Santo Domingo, Dominican Republic.

We would like to thanks to Yuya R.

REFERENCES

- [1] Adafre, Sisay F. & De Rijke, Maarten, (2006) “Finding Similar Sentences across Multiple Languages in Wikipedia”, In *Proceeding of EACL-06*, pages 62-69.
- [2] Bunescu, Razvan & Pasca, Marius (2006) “Using Encyclopedic Knowledge for Named Entity Disambiguation”, In *Proceeding of EACL-06*, pages 9-16.
- [3] Fung, Pascale & Cheung Percy, (2004) “ Multi-level Bootstrapping for extracting Parallel Sentences from a quasi-Comparable Corpus”, In *Proceeding of the 20th International Conference on Computational Linguistics*. Pages 350
- [4] Ramírez, Jessica, Asahara, Masayuki & Matsumoto, Yuji , (2008) “Japanese-Spanish Thesaurus Construction Using English as a Pivot”, In *Proceeding of The Third International Joint Conference on Natural Language Processing (IJCNLP)*, Hyderabad, India. pages 473-480.
- [5] Ramírez, Jessica & Matsumoto, Yuji (2012) “A Ruled-Based Approach for Aligning Japanese-Spanish Sentences from a Comparable Corpora”, *International Journal of Natural Language Computing (IJNLC)*.
- [6] Ramírez, Jessica & Matsumoto, Yuji (2013) “Extracción Automática de Diccionario Médico Japonés-Español. *Actualizaciones en Comunicación Social*, Santiago, Cuba. Vol.II.
- [7] Ramírez, Jessica, Matsumoto, Yuji, Muñoz, Darwin & Joyanes, Luís (2013) “Construcción Automática de Corpus Paralelo Japonés-Español en el Área de la Salud. *Memorias de VIII Conferencia Internacional de Lingüística*, Habana, Cuba.

AUTHORS

Jessica C. Ramírez

She received his M.S. degree from Nara Institute of Science and Technology (NAIST) in 2007. She is currently pursuing a Ph.D. degree. Her research interest Include machine translation and word sense disambiguation.



Yuji Matsumoto

He received his M.S. and Ph.D. degrees in information science from Kyoto University in 1979 and in 1989. He is currently a Professor at the Graduate School of Information Science, Nara Institute of Science and Technology. His main research interests are natural language understanding and machine learning.

Darwin Muñoz

He received his M.S. degree in Business from Quebec University, Canada in 2004 and Ph.D. degree in information science from Universidad Pontificia de Salamanca, Spain in 2014. He is currently a Professor at Universidad Iberoamericana.

RECENT APPROACHES TO ARABIC DIALOGUE ACTS CLASSIFICATIONS

AbdelRahim A. Elmadany¹, Sherif M. Abdou² and Mervat Gheith¹

¹Institute of Statistical Studies and Research (ISSR), Cairo University
ar_elmadany@hotmail.com, mervat_gheith@yahoo.com

²Faculty of Computers and Information, Cairo University
sh.ma.abdou@gmail.com

ABSTRACT

Building Arabic dialogue systems (Spoken or Written) has gained an increasing interest in the last few. For this reasons, there are more interest for Arabic dialogue acts classification task because it a key player in Arabic language understanding to building this systems. This paper describes the results of the recent approaches of Arabic dialogue acts classifications and covers Arabic dialogue acts corpora, annotation schema, utterance segmentation, and classification tasks.

KEYWORDS

Arabic Dialogue Acts, Spoken Dialogue Acts, Arabic Dialogue Language Understanding.

1. INTRODUCTION

Arabic is one of a class of languages where the intended pronunciation of a written word cannot be completely determining by its standard orthographic representation; rather, a set of special diacritics are needed to indicate the intended pronunciation. Different diacritics for the same spelling form produce different words with maybe different meanings. These diacritics, however, are typically omitted in most genres of written Arabic, resulting in widespread ambiguities in pronunciation and (in some cases) meaning. While native speakers are able to disambiguate the intended meaning and pronunciation from the surrounding context with minimal difficulty, automatic processing of Arabic is often hampered by the lack of diacritics. Text-to-speech (TTS), Part-Of-Speech (POS) tagging, Word Sense Disambiguation, and Machine Translation (ML) can be enumerated among a longer list of applications that vitally benefit from automatic discretization (Al-Badrashiny, 2009). Moreover, there are three categories of Arabic language: Classic Arabic “The old written form”, Modern Standard Arabic (MSA) “The famous written form today”, and dialectal Arabic “Native spoken languages of Arabic speakers” (Diab and Habash, 2007). Since, the written form of the Arabic language - MSA- is differs from dialectal Arabic. However, MSA used primarily for written form but the regional dialects is prevalence in spoken communications or day-to-day dealings. Unlike MSA, the dialects does not have a set of written grammars rules and have different characteristics e.g. morphology, syntax and phonetics. Moreover, Dialectal Arabic can mainly divided into six dialects groups: Maghrebi, Egyptian, Levantine, Gulf, Iraqi and other. Those regional dialects of Arabic are differ quite a bit from each other. Egyptian dialect commonly known as Egyptian colloquial language is the most widely understood Arabic dialect (Zaidan and Callison-Burch, 2012).

In this paper, we focus on language understanding component for Arabic dialogues system. However, there are few works have developed for Arabic spoken dialogue system either MSA or dialect as the best of our knowledge; this is mainly due to the lack of tools and resources that are necessary for the development of such systems (Zaghouani, 2014; Lhioui *et al.*, 2013). Therefore, building language-understanding component for dialogue system is requiring four parts: (1) Dialogue Acts Annotation Schema (2) Dialogue corpus (3) Segmentation Classification (4) Dialogue Acts Classification; consequently, this paper present a survey for these parts.

This paper is organized as follows: section 2 present the concepts and terminology that's used in the paper, section 3 present Arabic language understanding components (dialogue acts annotation schema, dialogue corpus, segmentation classification, and dialogue acts classification); and finally the conclusion and future works are reported in section 4.

2. CONCEPTS AND TERMINOLOGIES

This section present the concepts that related to language understanding and used in this paper.

2.1. Dialogue Act

The terminology of speech acts has been addressed by Searle (1969) based on Austin work (1962) as (Webb, 2010):

- Assertive commit the speaker to the truth of some proposition (e.g. stating, claiming, reporting, announcing)
- Directives attempts to bring about some effect through the action of the Hearer (e.g. ordering, requesting, demanding, begging)
- Commissures commit speaker to some future action (e.g. promising, offering, swearing to do something)
- Expressive are the expression of some psychological state (e.g. thanking, apologizing, congratulating)
- Declarations are speech acts whose successful performance brings about the correspondence between the propositional content and reality (e.g. resigning, sentencing, dismissing, and christening).

Dialogue act is approximately the equivalent of the speech act of Searle (1969). Dialog acts are different in different dialog systems. So, Major dialogue theories treat dialogue acts (DAs) as a central notion, the conceptual granularity of the dialogue act labels used varies considerably among alternative analyses, depending on the application or domain (Webb and Hardy, 2005). Hence, within the field of computational linguistics - recent work - closely linked to the development and deployment of spoken language dialogue systems, has focused on the some of the more conversational roles such acts can perform. Dialogue act (DA) recognition is an important component of most spoken language systems. A dialog act is a specialized speech act. DAs are different in different dialog systems. The research on DAs has increased since 1999, after spoken dialog systems became commercial reality (Stolcke *et al.*, 2000). So, (Webb, 2010) define the DAs as the labelling task of dialogue utterance that serve in short words a speaker's intention in producing a particular utterance.

2.2. Turn vs Utterance

In natural human conversation, turn refer to the speaker talking time and turn-taking refer to the skill of knowing when we start and finish the turn in the conversion. The turn boundary contains

one or more sentences moreover, the “turn-taking” is generally fixed to the expression of a single sentences. In the spoken dialogue system the term of utterance is refer to the one speech act. (Traum and Heeman, 1997) has defines the utterance unit by one or more of the following factors:

1. Speech by a single speaker, speaking without interruption by speech of the other, constituting a single Turn.
2. Has syntactic and/or semantic completion.
3. Defines a single speech act.
4. Is an intonational phrase.
5. Separated by a pause.

Consequently, this paper refers to an utterance as a small unit of speech that corresponds to a single act (Webb, 2010; Traum and Heeman, 1997). In speech research community, utterance definition is a slightly different; it refers to a complete unit of speech bounded by the speaker's silence while, we refer to the complete unit of speech as a turn. Thus, a single turn can be composed of many utterances. Moreover, turn and utterance can be the same definition when the turn contains one utterance as used in (Graja *et al.*, 2013) . Here an example of a long user *turn* from Arabic dialogues corpus that contains many utterances (Elmadany *et al.*, 2014):

Arabic	كنت عايزة افتح دفتر توفير عايزة اسأل على الإجراءات كنت عايزة اسألك لو سمحت
Buckwalter	lw smHt knt EAYzp As>lk knt EAYzp AftH dftr twfyr EAYzp As>l Ely Al<jrA'At
English	Excuse me I want to ask you I want open an account I need to know the proceeds

This *turn* contains four *utterances* as:

1. [لو سمحت] [lw smHt] [excuse me]
2. [كنت عايزة اسألك] [knt EAYzp As>lk] [I want to ask you]
3. [كنت عايزة افتح دفتر توفير] [knt EAYzp AftH dftr twfyr] [I want open an account]
4. [عايزة اسأل على الإجراءات] [EAYzp As>l Ely Al<jrA'At] [I need to know the proceeds]

3. LANGUAGE UNDERSTANDING COMPONENT

In this section, we present the recent researches for the four parts of building language-understanding component for Arabic dialogue systems, these parts are (1) Dialogue Acts Annotation Schema (2) Dialogue corpus (3) Segmentation Classification (4) Dialogue Acts Classification.

3.1. Dialogue Acts Annotation Schema

The idea of dialogue act plays a key role in studies of dialogue, especially in communicative behaviour understanding of dialogue participants, in building annotated dialogue corpora and in the design of dialogue management systems for spoken human-computer dialogue. Consequently, to build annotated dialogues corpus we need annotation schema that contains a list of predefined categories, semantic labels, or dialogue acts; schema is considering the key player to build the annotated corpus and dialogue acts classification task.

Searle (1969) has addressed the history of dialogue acts schema (see section 2.1). Moreover, the research on dialogue acts is increasing since 1999 after spoken dialogue systems become a commercial (Stolcke *et al.*, 2000). Many dialogue acts schema applied in non-Arabic dialogues such as English and Germany such as SWITCHBOARD-DAMSL schema.

As the best of our knowledge, all of the previous dialogue acts annotation schemas applied to mark-up dialogue corpora based on non-Arabic languages such as English, German and Spanish. Moreover, there are few efforts were done to propose dialogue acts annotation schemas for Arabic such as

— So, the first attempt was by (Shala *et al.*, 2010) that proposed dialogue acts schema contains 10 DAs:

- Assertion
- Response to Question
- Command
- Short Response
- Declaration
- Greetings
- Promise/Denial
- Expressive Evaluation
- Question
- Indirect
- Request

— (Dbabis *et al.*, 2012) has been improved (Shala *et al.*, 2010) schema; the reported schema based on multi-dimension “6th categories” 13 DAs:

- **Social Obligation Management**
 - Opening
 - Closing
 - Greeting
 - Polite Formula
 - Introduce
 - Thanking
 - Apology
 - Regret
- **Turn Management**
 - Acknowledgement
 - Calm
 - Clarify
- Clarify
- Feedback
- Out of topic
- Non understanding signal
- **Request**
 - Question
 - Order
 - Promise
 - Hope
 - Wish
 - Invocation
 - Warning
- **Argumentation**
- Opinion
- Appreciation
- Disapproval
- Accept
- Conclusion
- Partial Accept Reject
- Partial Reject
- Argument
- Justification
- Explanation
- Confirmation
- **Answer**
- **Statement**

These schemas have applied to mark-up dialogues corpora based on a general conversation discussion like TV talk-show programs.

— (Graja *et al.*, 2013) reported a words semantic labelling schema to mark-up dialogue utterance word-by-word for inquiry-answer dialogues specially train railway stations; this schema contains about 33 semantic labels for word annotation within five dimensions:

- **Domain concepts**
 - Train
 - Train_Type
 - Departure_hour
 - Arrival_hour
 - Day
 - Origin
 - Destination
 - Fare
 - Class
 - Ticket_Numbers
 - Ticket
 - Hour_Cpt
- Departure_Cpt
- Arrival_Cpt
- Price_Cpt
- Class_Cpt
- Trip_time
- Ticket_type
- **Requests concepts**
 - Path_Req
 - Hour_Req
 - Booking_Req
 - Price_Req
 - Existence_Req
 - Trip_timeReq
- Clarification_Req
- **Dialogue concepts**
 - Rejection
 - Acceptance
 - Politeness
 - Salutation (Begin)
 - Salutation (End)
- **Link concepts**
 - Choice
 - Coordination
- **Out of vocabulary**
 - Out

— Recently, (Elmadany *et al.*, 2014) reported a schema based request and response dimensions for inquiry-answer dialogues such as flights, mobile service operators, and banks; this schema contains DAs:

- Request Acts
 - Taking-Request
 - Service-Question
 - Confirm-Question
 - YesNo-Question
 - Choice-Question
 - Other-Question
 - Turn-Assign
- Response Acts
 - Service-Answer
- Other-Answer
 - Agree
 - Disagree
 - Greeting
 - Inform
 - Thanking
 - Apology
 - MissUnderstandingSign
 - Correct
 - Pausing
- Suggest
 - Promise
 - Warning
 - Offer
- Other Acts
 - Opening
 - Closing
 - Self-Introduce

3.2 Arabic Dialogue Acts Corpora

The use of corpora has been a key player in the recent advance in NLP research. However, the high costs of licensing corpora could be a difficult for many young researchers. Therefore, find freely available corpora is clearly a desirable goal, unfortunately; the freely available corpora are mostly not easily found and the most resources available from language data providers are expenses paid or exclusively reserved for subscribers. As the best of our knowledge, Arabic dialogue segmentation processing is considered hard due to the special nature of the Arabic language and the lack of Arabic dialogues segmentation corpora (Zaghouani, 2014). However, there are many annotated dialogued acts corpora for non-Arabic languages, these are the most annotated corpora used in DAs classifications tasks listed in (Webb, 2010) for non-Arabic languages such as:

- **MAPTASK**¹: consist of 128 English dialogues, containing 150,000 words.
- **VERBMOBIL**²: consist of 168 English dialogues, containing 3117 utterances. This corpus has annotated with 43 distinct Dialogue Acts.
- **SWITCHBOARD**³: consist of 1155 telephone conversations, containing 205,000 utterances and 1.4 million words.
- **AMITIES**⁴: consist of 1000 English human-human dialogues from GE call centres in the United Kingdom. These dialogues containing 24,000 utterances and a vocabulary size of around 8,000 words.
- **AMI**⁵: Contains 100 hours of meeting.

Unfortunately, to found fully Annotated Arabic dialogue acts corpus is more difficult but there are many of Arabic speech corpora prepared for Automatic Speech Recognition (ASR) research/application. Moreover, most of these corpora are available from the LDC or ELRA members with membership fees e.g. CALLHOME corpus⁶(Canavan *et al.*, 1997). Therefore, as the best of our knowledge, there are some efforts to building a fully annotated corpus for Arabic dialogues such as:

¹ Available at <http://www.hcrc.ed.ac.uk/maptask/>

² Available at http://www.phonetik.uni-muenchen.de/Bas/Bas_Korporaeng.html

³ Available at <ftp://ftp.ldc.upenn.edu/pub/ldc/public-data/swb1-dialogact-annot.tar.gz>

⁴ Available at <http://www.dcs.shef.ac.uk/nlp/amities/>

⁵ Available at <http://groups.inf.ed.ac.uk/ami/corpus/>

⁶ Available at <https://catalog.ldc.upenn.edu/LDC96S35>

- TuDiCoI⁷ (Tunisian Dialect Corpus Interlocutor): Corpus consists of Railway Information from the National Company of Railway in Tunisia (SNCFT) which a transcribed spoken Arabic dialogues; these dialogues are between the SNCFT staff and clients who request information about the train time, price, booking...etc. Moreover, the initial corpus of TuDiCoI has reported by (Graja *et al.*, 2010) containing 434 transcribed dialogues with 3080 utterances includes 1465 staff utterances and 1615 client utterances. So, TuDiCoI corpus has enriched by (Graja *et al.*, 2013) to contain 1825 transcribed dialogues with 12182 utterances includes 5649 staff utterances and 6533 client utterances. In addition, each dialogue consist of three utterances for clients and three utterances for staff; client turn is composed of average 3.3 words. The low words per clients utterances and dialogues length is due to the words used by clients to request for information about railway services. Moreover, the corpus turns are not segmented into utterances because it is sort and they considered the utterance is equal to the turn. Unfortunately, TuDiCoI are not annotated using DAs schema but it is marked-up by word-by-word schema (see section 3.1).
- (Elmadany *et al.*, 2014) is reported a manually annotated Arabic dialogue acts corpus and manually segmented turns into utterances for Arabic dialogues language understanding tasks. It has contains an 83 Arabic dialogues for inquiries-answers domains which are collected from call-centers. Moreover, this corpus contains two parts:
 - Spoken dialogues, which contains 52 phone calls recorded from Egyptian's banks and Egypt Air Company call-centers with an average duration of two hours of talking time after removing ads from recorded calls, and It consists of human-human discussions about providing services e.g. Create new bank account, service request, balance check and flight reservation. Moreover, these phone calls have transcribed using Transcriber^{®8}, a tool that is frequently used for segmenting, labeling and transcribing speech corpora.
 - Written 'Chat' dialogues, which contain 31 chat dialogues, collected from mobile network operator's online-support 'KSA Zain, KSA Mobily, and KSA STC'.

Building an annotated DAs corpus need four process recoding (for spoken)/ collecting (for chat) dialogues process, transcription process (for spoken only), segmentation process, and annotation process. Moreover, these processes are expensive.

3.3 Arabic Dialogue Segmentation

A segmentation process generally means dividing the long unit into meaningful pieces or small units "non-overlapping units" and it is considering one of the important solutions to solve Natural Language Processing (NLP) problems. Definition of segmentation will differ according to the NLP problem such as:

1. When dividing the text into topics, paragraphs, or sentences, properly named Text Segmentation e.g. (Tourir *et al.*, 2008; El-Shayeb *et al.*, 2007).
2. When dividing the sentences into a group of words, properly named Phrase Segmentation.
3. When dividing words into its clitics/affix (prefix, stem, and suffix), properly named tokenization e.g. (Diab *et al.*, 2004).

⁷ Available at <https://sites.google.com/site/anlprg/outils-et-corpus-realises/TuDiCoIV1.xml?attredirects=0>

⁸ <http://trans.sourceforge.net/en/presentation.php>

Build a completely Human-Computer systems and the belief that will happens has long been a favourite subject in research science. So, dialogue language understanding is growing and considering the important issues today for facilitate the process of dialogue acts classification; consequently segment the long dialogue turn into meaningful units namely utterances is increasing. Moreover, Human-Computer Dialogues are divided into different types: Speech Dialogues proper name “Spoken Dialogue” which works in waves and Written Dialogues proper name “Chat” or “Instant Messaging” (IM) which works on text. The waveform in spoken dialogues is usually segment the long input into short pieces based on simple acoustic criteria namely pauses “non-speech intervals”, this type of segmentation is namely acoustic segmentation; but it’s different when working in text such as chat dialogues, here use a linguistic segmentation. Consequently, to improve the human-computer system need for understand spoken dialogue by extracting the meaning of speaker's utterances, the acoustic segmentation is inadequate in such cases that are needed for further processing based on syntactically and semantically coherent units because it is not reflecting the linguistic structure of utterances (Stolcke and Shriberg, 1996). However, segmentation process is known in dialogues language understanding by many titles such as Utterances Segmentations, Turns Segmentations, and Dialogue Acts Segmentations (see section 2.2);

There are many approaches to understanding both dialogues types (spoken and written) for non-Arabic languages e.g. English, Germany, France... etc. (Ang *et al.*, 2005; Ivanovic, 2005; Zimmermann *et al.*, 2005; Ding and Zong, 2003). Moreover, understanding Arabic dialogues have gained an increasing interest in the last few years. To the best of our knowledge; there are few works interested in Arabic dialogue acts classification (see section 3.4); these works have used the user’s *turn* as an *utterance* without any segmentation e.g. (Shala *et al.*, 2010; Bahou *et al.*, 2008; Graja *et al.*, 2013; Lhioui *et al.*, 2013; Hijjawi *et al.*, 2013; Hijjawi *et al.*, 2014). In addition, there are a few works for the Arabic discourse segmentation such as:

- (Belguith *et al.*, 2005) has proposed a rule-based approach based on 83 rules for Arabic text segmentation which extracted from contextual analysis of the punctuation marks, the coordination conjunctions and a list of particles that are considered as boundaries between sentences.
- (Tourir *et al.*, 2008) has proposed a rule-based approach based on sentences connectors without relying on punctuation based on empirical study of 100 Articles, each article have between 450 and 800 words, for analysis to extract the connectors. Consequently, they provided term “*Passive*” for connector that does not imply any cutting point e.g. “و /and /w” and term “*Active*” for connector which indicates the beginning or the end of a segment e.g. “لكن /lkn”. In addition, they concluded that *Passive* connector has useful only when comes before *active*. Hence, they are tested the approach on 10 articles, each article have 500 to 700 words.
- (Khalifa *et al.*, 2011) proposed a Machine-Learning approach using SVM based on the connector “و /and /w”. Moreover, they reported sixth types of “و /and /w” connector that divided into two classes: (1) “*Fasl*” for a connector that indicates the beginning of segments, and (2) “*Wasl*” for connector that does not have any effect on segmentation. In additional, they are built a corpus for newspapers and books which includes 293 instances of the connector “و /and /w” and added diacritization marks manually to the corpus text (training and testing) during the preparation steps. However, these approach very similar to (Tourir *et al.*, 2008) when considering the connector “و /and /w”.

- (Keskes *et al.*, 2012) proposed a rule-based approach based on three principals: (1) using punctuation indicators principal only (2) using lexical cues principal only (3) using mixed punctuation indicators and lexical cues. In addition, they used 150 news articles (737 paragraphs, 405332 words) and 250 elementary school textbooks (1095 paragraphs, 29473 words) for built the lexical cues and effective punctuation indicators. Moreover, they concluded two types of punctuation indicators: (1) “*strong*” that always identify the end or the start of the segments such as the exclamation mark (!), the question mark (?), the colon (:), and the semi-colon (;) (2) “*Weak*” that don’t always identify the begin or the begin of the segment segments such as full-stop (.), the comma (,), quotes, parenthesis, brackets, braces and underscores; They reported the mixed punctuation indicators and lexical cues principal has the best results in textbooks and newspapers.

These approaches are not testing on Arabic dialogues that completely differs for newspapers and books articles; and Arabic spontaneous dialogues is properly dialect Arabic, which is informal text.

3.4 Recent approaches to Arabic Dialogue Acts Classification

There are two ways to understand the dialogues language (Webb and Hardy, 2005):

- **Shallow understanding:** It is simple spotting keywords or having lists of, for example, every location recognized by the system. Several systems are able to decode directly from the acoustic signal into semantic concepts precisely because the speech recognizer already has access to this information.
- **Deeper analysis:** Using linguistic methods; including part-of-speech tagging, syntactic parsing and verb dependency relationships.

Using Machine Learning (ML) for solving the DA classification problem, researchers have not historically published the split of training and testing data used in their experiments, and in some cases methods to reduce the impact of the variations that can be observed when choosing data for training and testing have not been used (Webb, 2010). Moreover, DAs are practically used in many live dialogue systems such as Airline Travel Information Systems (ATIS) (Seneff *et al.*, 1991), DARPA (Pellom *et al.*, 2001), VERBMOBIL project (Wahlster, 2000), and Amities dialogue system (Hardy *et al.*, 2004). Now, we will describe in brief some of DAs approaches over annotated corpora to recognize dialogue acts:

- Several approaches have proposed for DAs classification and N-gram models can be considering the simplest method of DA prediction; predicting the upcoming, DA based on some limited sequence of previous DAs such as (Hardy *et al.*, 2004; Webb, 2010; Webb and Hardy, 2005; Webb *et al.*, 2005a, 2005b; Nagata and Morimoto, 1994; Niedermair, 1992). Moreover, (Reithinger and Klesen, 1997; Boyer *et al.*, 2010; Stolcke *et al.*, 2000) are used Hidden Markova Model (HMM) with N-gram.
- Samuel et al. (1998) used Transformation-Based Learning (TBL) (Brill, 1995) over a number of utterance features, including utterance length, speaker turn and the dialogue act tags of adjacent utterances.
- (Carberry and Lambert, 1999) used a rule-based model of DA recognition that uses three sources of knowledge, linguistic (including cue phrases), contextual and world knowledge. Moreover, the linguistic knowledge is used primarily to identify if the speaker has some belief in the evidence presented, using prior known cue phrases e.g. BUT, or the use of surface-negative question forms (Doesn't X require Y?) (Webb, 2010). Also (Prasad and Walker,

2002) are used a rule based learning method in the DARPA Communicator dialogues. More recently, (Georgila *et al.*, 2009) extended (Prasad and Walker, 2002) work to include manually constructed context rules that cover the user side of the Communicator dialogues

- Bayesian approaches have proven to be effective for DAs classification (Webb, 2010); (Grau *et al.*, 2004) used Naïve Bayesian over the WITCHBOARD corpus within a tri-gram language model.
- (Ji and Bilmes, 2005; Ji and Bilmes, 2006) are investigated the use of dynamic Bayesian networks (DBNs) using graphical models and they reported the best performing set of features is a tri-gram model of the words in the utterances combined with a bi-gram model of DA.

These approaches are tested on non-Arabic dialogues e.g. English, Germany, France... etc. which completely differs for Arabic dialogues. Moreover, understanding Arabic dialogues have gained an increasing interest in the last few years. To the best of our knowledge, there are few works interested in Arabic dialogue acts classification such as:

(Bahou *et al.*, 2008) proposed a method for the semantic representations of utterances of spontaneous Arabic speech based on the frame grammar formalism as show in

- Figure 1 and it's tested on Tunisian national railway queries (1003 queries representing 12321 words) collected using Wizard-of-Oz technology. In addition, this method consists of three major steps: a pre-treatment step that includes the normalization of the utterance and its morphological analysis; a step of semantic analysis that assigns semantic tags to each lexical unit of query; and a frame generation step that identifies and fills the semantic frames of the utterance. They reported 37% recall, 60.62% precision and 71.79% as F-Measure for classification with average execution time for the utterance is 0.279 sec.

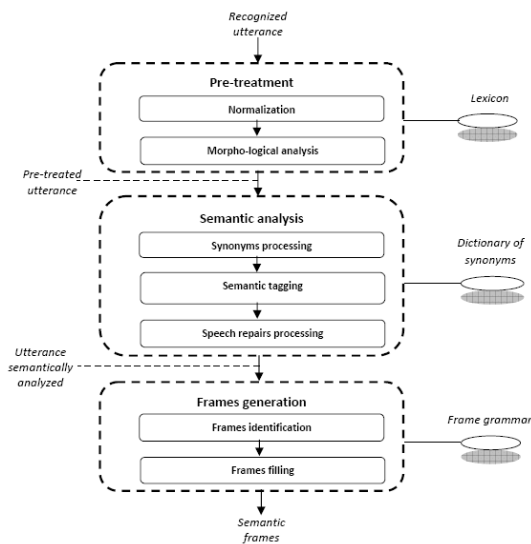


Figure 1. (Bahou *et al.*, 2008) Approach

- (Shala *et al.*, 2010) proposed a fully automated method for speech act classification for Arabic discourse based on the hypothesis that the initial words in a sentence and/or their parts-of-speech are diagnostic of the particular speech act expressed in the sentence. In addition, used the semantic categorization of these words in terms of named entities and combined this approach with Support Vector Machines (SVM) models to automatically derive the

parameters of the models they used to implement the approach as show in Figure 2. Moreover, they used two machine-learning algorithms, Naïve Bayes and Decision Trees to induce classifiers acts for Arabic texts and they reported 41.73% as accuracy scores of all models.

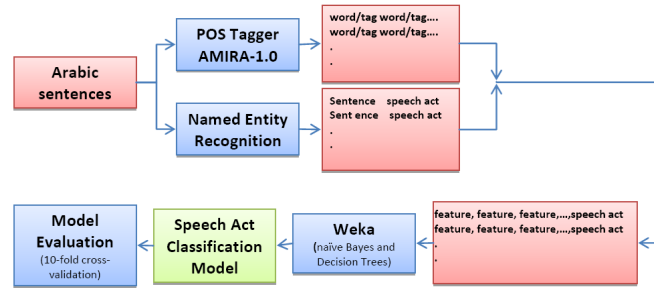


Figure 2. (Shala *et al.*, 2010) Approach

- (Lhioui *et al.*, 2013) proposed an approach based on syntactic parser for the proper treatment of utterances including certain phenomena such as ellipses and it has relies on the use of rule-base (context free grammar augmented with probabilities associated with rules) as show in Figure 3. In addition, they used HHM for creating the stochastic model (if a pretreated and transcribed sequence of words - this words are obviously the output of recognition module - and their annotated corresponding sequences was taken). Moreover, they applied their method on Tunisian touristic domain collected using Wizard-of-Oz technology which contains 140 utterances recorded from 10 speakers with 14 query types (DA) e.g. negation, affirmation, interrogation and acceptance and reported 70% recall, 71% precision and 73.79% as F-measure for classification with average execution time 0.29 seconds to process an utterance of 12 words

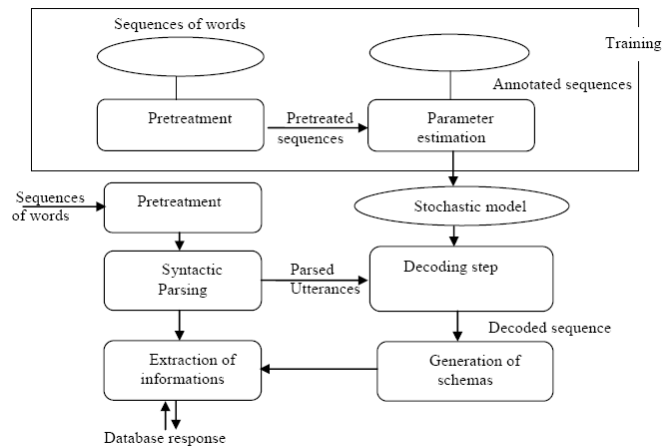


Figure 3. (Lhioui *et al.*, 2013) Approach

- (Graja *et al.*, 2013) proposed discriminative algorithm based on Conditional Random Fields (CRF)⁹ to semantically label spoken Tunisian dialect turns which are not segmented into utterances from TuDiCoI corpus (see section 3.2). Moreover, they applied some treatments to improve turn's structure: (1) lexical normalization such as replacing the word “زررفسيون”

⁹ Conditional random fields (CRF) are undirected graphical models trained to maximize a conditional probability which proposed by (Lafferty *et al.*, 2001)

“Reservation” for all its forms e.g. “رزرفسيون”, “رازرفسيون”, “رازارفسيون”, “ريزرفسيون”. (2) Morphological analysis and lemmatization such as replacing the word “خارج” “*is going*” and “يخرج” “*goes*” by the following canonical form “خرج” “*go*”. (3) Synonyms treatment, this treatment consists in replacing each word by its synonym. In addition, they applied the approach on two data sets one without the treatments and the second with the treatments; and they reported that the treatments has reduce the errors rate compared to the non-treatments data set from 12% to 11%.

— (Hijjawi *et al.*, 2013) proposed approach based on Arabic function words such as “هل” “do/does”, “كيف” “How” and it’s focused on classifying questions and non-questions utterances. Moreover, the proposed approach extracts function words features by replacing them with numeric tokens and replacing each content word with a standard numeric token; they used the Decision Tree to extract the classification rules and this approach used on Conversational Agent called ArabChat (Hijjawi *et al.*, 2014) to improve its performance by differentiating among question-based and non-question-based utterances.

4. CONCLUSIONS

We presented this survey for the recent approaches to Arabic dialogue Acts classification and the goal behind this study is to promote the development and use of Human-Computer research in Arabic dialogues. The results obtained showed that a few works that developed based on Arabic dialogues. Consequently, we hope that this initial attempt to increasing and improve this research as non-Arabic languages.

REFERENCES

- [1] Al-Badrashiny, M. 2009. Automatic Diacritizer for Arabic text. MSc in Electronics & Electrical Communications. Faculty of Engineering, Cairo University.
- [2] Ang, J., et al. 2005. Automatic Dialog Act Segmentation and Classification in Multiparty Meetings. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05).1061 - 1064.
- [3] Bahou, Y., et al. 2008. Towards a Human-Machine Spoken Dialogue in Arabic. In Proceedings of Workshop on HLT & NLP within the Arabic world: Arabic Language and local languages processing: Status Updates and Prospects, at the 6th Language Resources and Evaluation Conference (LREC'08). Marrakech, Maroc.
- [4] Belguith, L., et al. 2005. Segmentation de textes arabes basée sur l'analyse contextuelle des signes de ponctuations et de certaines particules. In Proceedings of 12th Conference on Natural Language Processing (TALN'2005).451-456.
- [5] Boyer, K., et al. 2010. Dialogue Act Modelling in a Complex Task-Oriented Domain. In Proceedings of SIGDIAL. Tokyo, Japan:297-305.
- [6] Brill, E. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. Computational Linguistics, 21(4):543-565.
- [7] Canavan, A., et al. 1997. CALLHOME Egyptian Arabic Speech. in L. D. Consortium ed Philadelphia
- [8] Carberry, S., and L. Lambert. 1999. A Process Model for Recognizing Communicative Acts and Modeling Negotiation Sub-dialogues. Computational Linguistics.
- [9] Dbabis, S. B., et al. 2012. Dialogue Acts Annotation Scheme within Arabic discussion. In Proceedings of SemDial 2012 The 16th workshop on semantics and pragmatics of dialogue Sorbonne, Paris, France.
- [10] Diab, M., and N. Habash. 2007. Arabic Dialect Processing Tutorial. In Proceedings of The Human Language Technology Conference of the North American. Rochester
- [11] Diab, M., et al. 2004. Automatic Tagging of Arabic Text: From raw text to Base Phrase Chunks. In Proceedings of HLT/NAACL. Boston.

- [12] Ding, L., and C. Zong. 2003. Utterance segmentation using combined approach based on bi-directional N-gram and maximum entropy. Proceedings of the second SIGHAN workshop on Chinese language processing. Association for Computational Linguistics, 17(
- [13] El-Shayeb, M., et al. 2007. ArabicSeg: An Arabic News story Segmentation System. In Proceedings of Third International Computer Conference Cairo University (ICENCO 2007). Cairo, Egypt.
- [14] Elmadany, A. A., et al. 2014. Arabic Inquiry-Answer Dialogue Acts Annotation Schema. IOSR Journal of Engineering (IOSRJEN), 04(12-V2):32-36.
- [15] Georgila, K., et al. 2009. Automatic Annotation of Context and Speech Acts for Dialogue Corpora. Journal of Natural Language Engineering:315-353.
- [16] Graja, M., et al. 2010. Lexical study of a spoken corpus in Tunisian dialect. The International Arab Conference on Information Technology (ACIT 2010). Benghazi, Libya.
- [17] Graja, M., et al. 2013. Discriminative Framework for Spoken Tunisian Dialect Understanding. SLSP.102-110.
- [18] Grau, S., et al. 2004. Dialogue Act Classification using a Bayesian Approach. In Proceedings of 9th Conference Speech and Computer.
- [19] Hardy, H., et al. 2004. Data-driven strategies for an automated dialogue system. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics.
- [20] Hijjawi, M., et al. 2013. User's Utterance Classification Using Machine Learning for Arabic Conversational Agents. In Proceedings of 5th International Conference on Computer Science and Information Technology (CSIT).223-232.
- [21] Hijjawi, M., et al. 2014. ArabChat: an Arabic Conversational Agent. In Proceedings of 6th International Conference on Computer Science and Information Technology (CSIT).227-237.
- [22] Ivanovic, E. 2005. Automatic utterance segmentation in instant messaging dialogue. In Proceedings of The Australasian Language Technology Workshop.241-249.
- [23] Ji, G., and J. Bilmes. 2005. Dialog Act Tagging Using Graphical Models. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05).
- [24] Ji, G., and J. Bilmes. 2006. Backo Model Training using Partially Observed Data: Application to Dialog Act Tagging. In Proceedings of the Human Language Technology/ American chapter of the Association for Computational Linguistics (HLT/NAACL'06).
- [25] Keskes, I., et al. 2012. Clause-based Discourse Segmentation of Arabic Texts. In Proceedings of The eighth international conference on Language Resources and Evaluation (LREC). Istanbul.
- [26] Khalifa, I., et al. 2011. Arabic Discourse Segmentation Based on Rhetorical Methods. International Journal of Electric & Computer Sciences IJECS-IJENS, 11(01).
- [27] Lafferty, J., et al. 2001. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In Proceedings of International Conference on Machine Learning (ICML).282-289.
- [28] Lhioui, C., et al. 2013. A Combined Method Based on Stochastic and Linguistic Paradigm for the Understanding of Arabic Spontaneous Utterances. In Proceedings of CICLing 2013, Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science. Samos, Greece, 7817(2):549-558.
- [29] Nagata, M., and T. Morimoto. 1994. First Steps Towards Statistical Modeling of Dialogue to Predict the Speech Act Type of the Next Utterance. In Proceedings of Speech Communication.
- [30] Niedermair, G. 1992. Linguistic Modeling in the Context of Oral Dialogue. In Proceedings of International Conference on Spoken Language Processing (ICSLP'92). Ban, Canada: 63-638.
- [31] Pellom, B., et al. 2001. University of Colorado Dialog Systems for Travel and Navigation. In Proceedings of HLT '01: Proceedings of the First International Conference on Human Language Technology Research. USA.
- [32] Prasad, R., and M. Walker. 2002. Training a Dialogue Act Tagger for Humna-Human and Human-Computer Travel Dialogues. In Proceedings of the 3rd SIGdial workshop on Discourse and Dialogue. Philadelphia, Pennsylvania.
- [33] Reithinger, N., and M. Klesen. 1997. Dialogue Act Classification Using Language Models. In Proceedings of EuroSpeech.
- [34] Seneff, S., et al. 1991. Interactive Problem Solving and Dialogue in the ATIS Domain. In Proceedings of HLT '91: Proceedings of the Workshop on Speech and Natural Language. USA:354-359.
- [35] Shala, L., et al. 2010. Automatic Speech Act Classification In Arabic. In Proceedings of Subjetividad y Procesos Cognitivos Conference 14(2):284-292.
- [36] Stolcke, A., et al. 2000. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. Computational Linguistics, 26(3):339-373.

- [37] Stolcke, A., and E. Shriberg. 1996. Automatic linguistic segmentation of conversational speech. In Proceedings of The Fourth International Conference on Spoken Language Processing (LCSLP 96). 2(1005-1008).
- [38] Touir, A. A., et al. 2008. Semantic-Based Segmentation of Arabic Texts. . Information Technology Journal, 7(7).
- [39] Traum, D., and P. A. Heeman. 1997. Utterance units in spoken dialogue. Dialogue processing in spoken language systems Springer. Berlin Heidelberg, 125-140.
- [40] Wahlster, W. 2000. Verbmobil: Foundations of Speech-To-Speech Translation Springer.
- [41] Webb, N. 2010. Cue-Based Dialogue Act Classification. Ph.D. dissertation. University of Sheffield, England.
- [42] Webb, N., and H. Hardy. 2005. Data-Driven Language Understanding for Spoken Language Dialogue American Association for Artificial.
- [43] Webb, N., et al. 2005a. Dialogue Act Classification Based on Intra-Utterance Features. In Proceedings of the AAAI Work-shop on Spoken Language Understanding.
- [44] Webb, N., et al. 2005b. Empirical Determination of Thresholds for Optimal Dialogue Act Classification. In Proceedings of the Ninth Workshop on the Semantics and Pragmatics of Dialogue.
- [45] Zaghouni, W. 2014. Critical Survey of the Freely Available Arabic Corpora. In Proceedings of Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools (LREC2014).
- [46] Zaidan, O. F., and C. Callison-Burch. 2012. Arabic dialect identification. Computational Linguistics, 52(1).
- [47] Zimmermann, M., et al. 2005. Toward Joint Segmentation and Classification of Dialog Acts in Multiparty Meetings. In Proceedings of Proc. Multimodal Interaction and Related Machine Learning Algorithms Workshop (MLMI-05).

INTENTIONAL BLANK

QUICK PAD TAGGER: AN EFFICIENT GRAPHICAL USER INTERFACE FOR BUILDING ANNOTATED CORPORA WITH MULTIPLE ANNOTATION LAYERS

Marc Schreiber¹, Kai Barkschat¹, Bodo Kraft¹ and Albert Zündorf²

¹FH Aachen, University of Applied Sciences, Germany, Jülich
{marc.schreiber,barkschat,kraft}@fh-aachen.de

²University of Kassel, Germany, Kassel
zuendorf@uni-kassel.de

ABSTRACT

More and more domain specific applications in the internet make use of Natural Language Processing (NLP) tools (e. g. Information Extraction systems). The output quality of these applications relies on the output quality of the used NLP tools. Often, the quality can be increased by annotating a domain specific corpus. However, annotating a corpus is a time consuming and exhaustive task. To reduce the annotation time we present a custom Graphical User Interface for different annotation layers.

KEYWORDS

Natural Language Processing, Language Resources, Annotated Corpora, Annotation Layers, Annotation Speed, Graphical User Interface

1. INTRODUCTION

Many modern applications are analyzing natural language resources. For example, companies process incoming mails automatically and they are trying to identify if customers are satisfied with the provided services or not. To facilitate such analyses these applications apply Natural Language Processing (NLP) techniques, deriving meaning from natural language. In conclusion, the quality of the analyses relies on the output quality of the NLP tools.

The output quality of NLP tools varies when the tools will be applied in different domains [1, 2]. When NLP tools are used in specific domains, the quality can be increased by developing a domain specific algorithms [3, 4, 5] or training existing NLP tools on domain specific corpora [6, 7]. In both cases an annotated corpus is required to evaluate the output quality of NLP tools in different domains.

The creation of an annotated corpus is a time-consuming and error-prone process. To support and improve the annotation process Graphical User Interface (GUI) tools have been evolved, supporting annotators to create annotated corpora [8, 9, 10]. Often these tools offer to annotate a specific annotation layer (e. g. Named Entities (NEs)) with a specific GUI for this layer. Other tools provide to annotate multiple arbitrary layers with a generic GUI for all annotation layers.

However, both types of annotation tools have their pros and cons. Annotation tools providing to annotate a specific layer are optimized to annotate this specific layer, but at the same time they are restricted in their functionality. In contrast to these tools, other annotation tools can annotate multiple layers but they do not provide an optimized GUI for any annotation layer, resulting in higher annotation effort.

In this paper we present the Quick Pad Tagger (QPT) which closes the gap between layer specific and multiple layer annotation tools. The QPT provides to annotate multiple annotation layers and for each layer the QPT provides an optimized GUI.

The constellation of multiple annotation layers and specific GUI leads to better annotation speed. Additionally, the QPT is designed as semi-automatic annotation tool, providing suggestions to the user. The semi-automatic manner of the QPT increases the annotation speed furthermore. This paper is structured as follows: Section 2 will outline related work. Following that, Section 3 will introduce the QPT followed by an evaluation in Section 4. Finally, Section 5 gives a conclusion and outlines future work.

2. RELATED WORK

Many studies show that annotation time can be reduced when an annotated corpus will be developed. In the clinical domain Lingren et al. [11] demonstrate that pre-annotation reduces the annotation time significantly for the NE annotation layer. As an additional result they revealed that pre-annotation did not influence the Inter Annotator Agreement (IAA) or annotator performance. Loftsson et al. [12] investigate to use pre-annotations for the Part-of-Speech (POS) tag annotation layer. The authors describe that using pre-annotations reduces the effort of developing an annotated corpus. Fort and Sagot [13] investigate pre-annotation methods for POS tags more deeply and they verify that those methods result in better annotation accuracy.

Lingren et al. [11], Loftsson et al. [12], and Fort and Sagot [13] investigate the annotation process with pre-annotated corpora and they show that annotation time can be reduced. We reuse the idea of pre-annotation in the semi-automatic annotation process of the QPT (c. f. Section 3.2), resulting in higher annotation speed (c. f. Section 4).

SALTO [8] is a specific GUI annotation tool. SALTO is able to annotate syntactic structures in TIGER XML corpora—Mengel and Lezius [14] describe the TIGER XML corpus format in detail. SALTO also enables to add semantic classes and roles to TIGER XML corpora. Both features are based on graph representations and use a mouse based input method.

Knowtator [9] is a generic annotation GUI tool, implemented as a Protégé plugin [15]. The annotation schema can be defined by Protégé's knowledge-based editor which enables the generic annotation manner. Knowtator's input method is a mouse based input method. Nevertheless, Knowtator is very difficult to use for unskilled annotators.

Webanno [10] is a web-based annotation tool. It also provides a generic user interface to annotate different annotation layers with a mouse based input method. The annotation configuration is hidden to the user which makes it easier to annotate documents. A new version of Webanno provides annotation suggestions to the user as well [16].

All those annotation tools [8, 9, 10] are supporting a single annotation layer or they provide a generic GUI for multiple annotation layers. However, in the first case the functionality is limited and in the second case the annotation process is not optimized regarding annotation time. The QPT closes this gap by providing a specific GUI for each annotation layer.

3. ANNOTATING CORPORA WITH THE QUICK PAD TAGGER

The Quick Pad Tagger (QPT) follows a general design principle which is embedded into its name: Annotating text corpora (refers to Tagger) should be done as quickly as possible (refers to Quick) and therefore a minimal set of input keys is used (refers to Pad/Gamepad which provides only a limited set of keys). Inspired by a video game input method, the motivation of using a keyboard based input method can be described by the following reasons:

1. **The input method should be efficient:** Both Lane et al [17] and Omanson et al. [18] show that keyboard based input methods are often more efficient than mouse based input methods.
2. **The input method prepares to embed gamification elements:** Annotating a corpus is a time-consuming and monotonous process. Gamification is an element to improve the user experience [19]. With an improved user experience the annotation process is less exhausting.

Additionally, the QPT is a semi-automatic annotation tool for multiple annotation layers. This is motivated by the fact that suggesting annotations can improve the annotation speed (c. f. Fort and Sagot [13] and Yiman et al. [16], more details will be provided by Section 3.2).

3.1. Graphical User Interface of the Quick Pad Tagger

Currently, the QPT supports to annotate the following layers:

- Text Segmentation (TS), c. f. Figure 1
- Part-of-Speech (POS), c. f. Figure 2
- Named Entities (NEs), c. f. Figure 3
- Constituency-based Parse Trees (CPTs), c. f. Figure 4

The user can switch between the annotation layers by using the corresponding buttons (c. f. top of Figure 1). The user can switch anytime among the layers, unless required information are provided. For example, the Constituency-based Parse Tree (CPT) annotation layer requires POS information. If the document does not contain any POS tags, the user is not able to select the CPT annotation layer.

Figure 1 illustrates adding Text Segmentation (TS) annotations to a document. To annotate the TS information the QPT provides a cursor based input method. Basically, the user moves the cursor with the arrow keys and marks end of sentences with enter (blue marks in Figure 1). For splitting words into multiple tokens the user can use the spacebar (green marks in Figure 1). If one of the annotations is incorrect, the user can remove the annotations with the delete key.

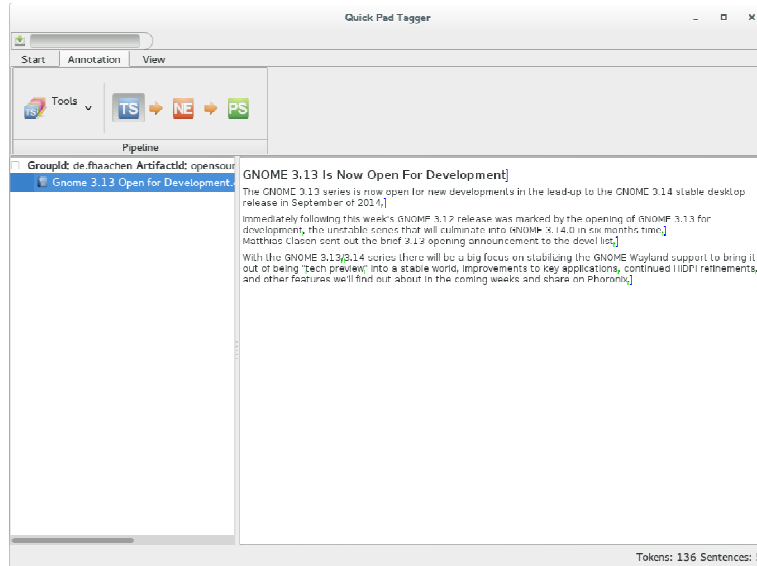


Figure 1: Text Segmentation Annotation with Quick Pad Tagger

To increase the annotation speed for the TS layer the cursor based input method moves from token to token because a character based moving slows down the annotation speed. The text displayed in Figure 1 contains of 746 characters. By using a character based input method the user would have to press the right arrow key 746 times to move the cursor from the beginning to the end of the document. After an initial whitespace tokenization the text contains of 124 tokens which reduces pressing the right arrow key to 124 times. To move the cursor from character to character the user just needs to press the control key.

Figure 2 displays the process of annotating POS tags. The QPT provides a sentence based selection and for each selected sentence the QPT shows a popup menu. By pressing left or right arrow key a token will be selected and by pressing up or down arrow key a POS tag will be assigned.

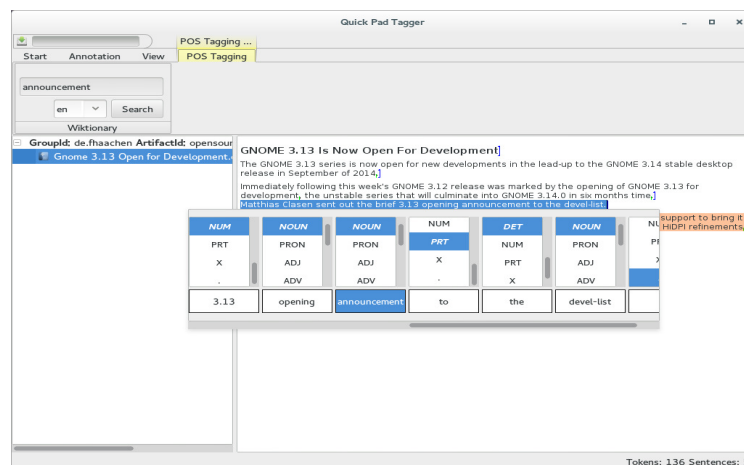


Figure 2: Part-of-Speech Tags Annotation with Quick Pad Tagger

Often annotators look up words in dictionaries like Wiktionary—Wiktionary also serves a platform for other NLP resources [20]. To reduce the effort to open Wiktionary and search for a specific word, the QPT provides an embedded search function (c. f. top left corner in Figure 2). When the user selects a token, the value of the token will be inserted into the text box. By pressing the F1 key the QPT will search on Wiktionary for the selected value which improves the annotation speed.

Figure 3 illustrates the process of annotating NEs. The QPT provides a token based selection which can be changed with the left and right arrow keys. The selection will be extended by pressing shift and arrow keys. The concepts of the NEs are displayed in a popup menu. The annotator selects another concept by pressing the up and down arrow key. By pressing the enter key the user assigns the selected concept to the selected tokens.

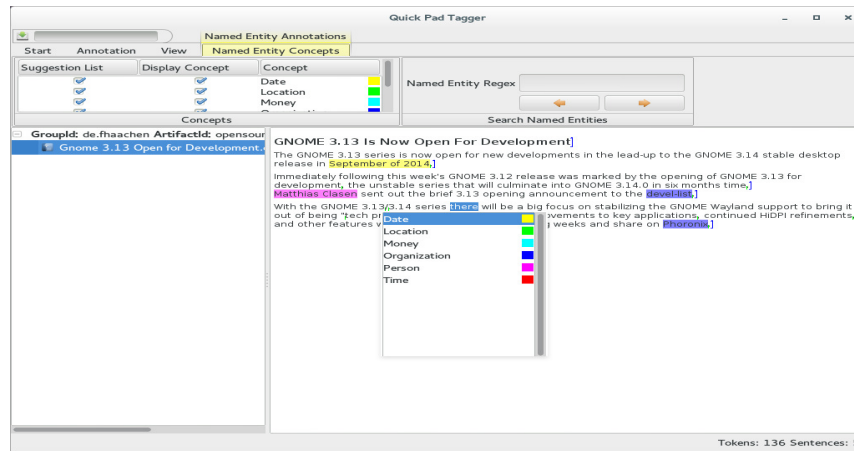


Figure 3: Named Entity Annotation with Quick Pad Tagger

Figure 4 shows the process of annotating CPTs. Similar to annotating POS tags the QPT provides a sentence based selection with an additional popup menu. The user can combine tokens to phrases and assigns each phrase a phrase tag. The left and right arrow selects a token or phrase and with the shift key the selection can be expanded. The spacebar is used to combine tokens or phrases to a new phrase. By pressing up or down arrow key the annotator selects a phrase tag.

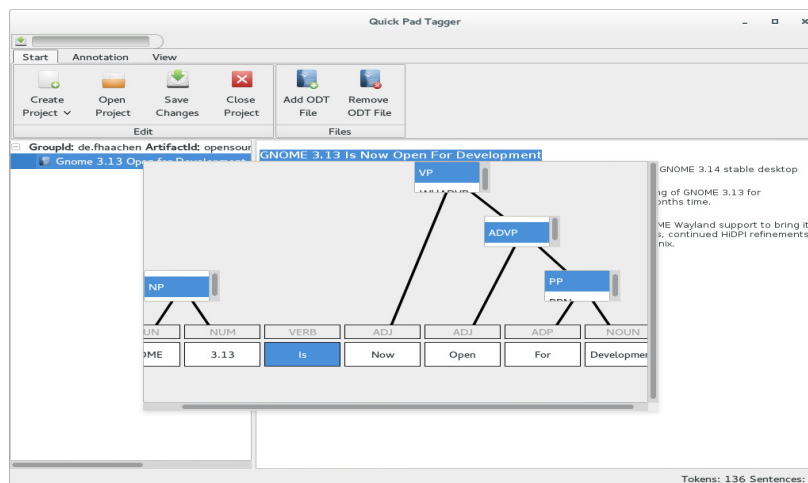


Figure 4: Constituency-based Parse Tree Annotation with Quick Pad Tagger

3.2. Semi-automatic Annotation Process

As mentioned at the beginning of Section 3 the QPT is a semi-automatic annotation tool. In this section we will provide an explanation how generating suggestions works (illustrated in Figure 5). When the first document will be added to the corpus, the annotator has to annotate the whole document without any suggestions. When the first document has been annotated, the QPT can access the previous annotation information for providing suggestions to the annotator.

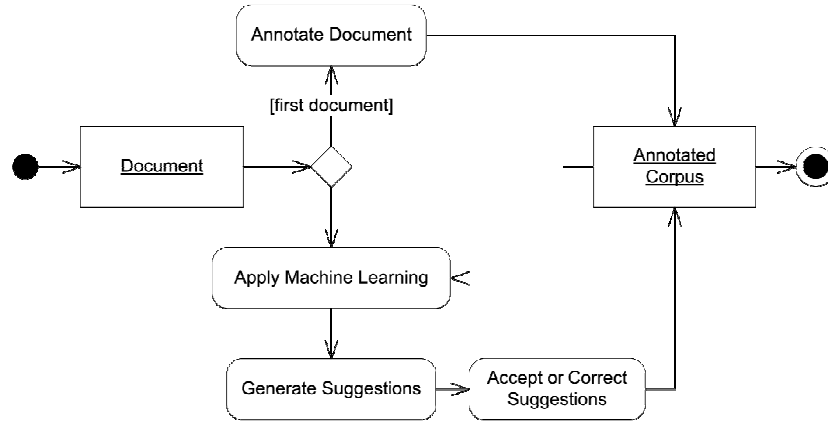


Figure 5: Semi-automatic Annotation Process

Therefore the QPT takes the information of the already existing annotated corpus and applies Machine Learning (ML) methods to generate a ML model. After that, the QPT uses the ML model to provide suggestions to the annotator. Finally, the annotator has to accept or correct the suggestions.

The ML methods for providing the annotation suggestion can be selected by the user. Through a plugin system the QPT can be extended with different NLP implementations—OpenNLP [21] and Stanford CoreNLP [22] are two example NLP implementations, providing annotation suggestions to the user.

4. EVALUATION

For the evaluation process of the conceived QPT GUI design we define the annotation speed metric as tokens per second for every annotation layer. For example, if a user is able to annotate a document consisting of 100 tokens with POS tags in 50 seconds, the annotation speed is two. This metric is independently of the used annotation tool and makes it possible to compare annotation tools regarding their effectiveness.

For the evaluation we compare the annotation speed of Webanno [10] with the QPT. OpenNLP provides the annotation suggestions for the QPT. Webanno uses a generic annotation GUI for every annotation layer and the QPT uses a specific GUI for every annotation layer. This setup was chosen to verify the hypothesis that a specific GUI for each annotation layer leads to higher annotation speed.

The annotation speed for the comparison has been measured by four different users annotating the same corpus with the following annotation layers: TS, NEs and POS tags. For annotating NEs we use the concepts Date, Location, Money, Organization and Person because these concepts are

known by the annotators. As POS tag set we use the universal POS tag set of Petrov et al. [23] because it is easy to learn for German texts.

Two users annotate the corpus with Webanno (Annotator A and B) and the other two users annotate the corpus with the QPT (Annotator C and D). None of the users annotated a corpus before. This setup of easy to understand annotation layers and non-experienced annotators ensures both: The annotation process does not take too long and none of the annotators is biased by a known annotation tool.

During the comparison we annotate a small corpus consisting of six documents. On average each document consists of 330 tokens. The documents come from different German online news portals:

1. Golem.de: <http://www.golem.de/>
2. stern.de GmbH: <http://www.stern.de/>
3. Süddeutscher Verlag: <http://www.sueddeutsche.de/>

The following three sections describe the evaluation comparing the annotation speed of Webanno and the QPT. Each section describes the results of the experiment comparing Webanno and the QPT for one annotation layer (TS, POS and NEs). Additionally, each section illustrates how the semi-automatic annotation process influences the annotation speed. Therefore, we analyze the F_1 score or accuracy of the suggestions: The F_1 score/accuracy is measured by taking the previous documents, generating the ML model, generating suggestions and comparing the users' annotations with the suggestions.

4.1. Text Segmentation Annotation Layer

Webanno provides no feature to annotate TS information. To be able to compare the QPT with Webanno we use following approach: The Annotators A and B create a plain text file with a text editor. Each line in the text file contains one sentence and the tokens are separated by whitespaces. The text files are then converted to Text Corpus Format files [24] and then imported into Webanno.

Table 1 shows the average annotation time for each document of the corpus. The QPT reduces the annotation time of TS information by 27 percent.

Table 1: Average Annotation Time for Annotating a Document with Text Segmentation

Webanno		Quick Pad Tagger	
Annotator A	Annotator B	Annotator C	Annotator D
2:28 min	2:25 min	1:45 min	1:50 min

Figure 6 displays the annotation speed for each document and annotator, annotating TS information. At the beginning the annotation speed of each annotator is almost the same. After annotating the first document the annotation speed of Annotator A and B increases slightly and stays more or less the same (on average 2.41 tokens per second). In contrast to the annotation speed of Annotator A and B the annotation speed of Annotator C and D increases constantly. At the end of the experiment the annotation speed of Annotator C and D is on average 2.3 times higher.

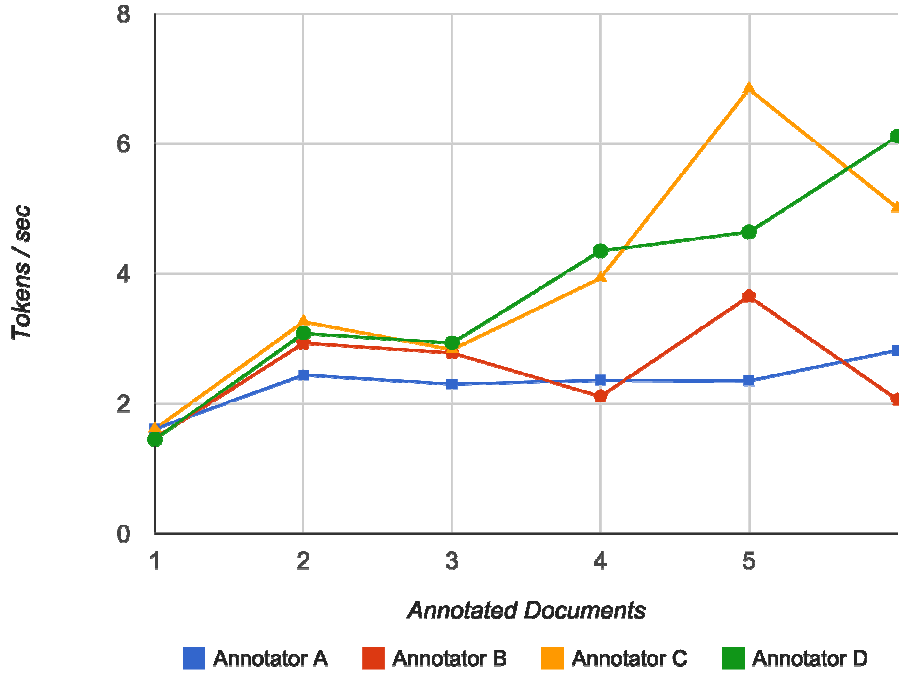


Figure 6: Annotation Speed for Text Segmentation

Figure 7 shows the F_1 score of the suggestions. On average the F_1 score is 93 percent. In conclusion, the annotation task of the Annotators C and D limits to reading the text and verifying if suggestions are correct. This limitation leads to a speedup compared to the Webanno approach.

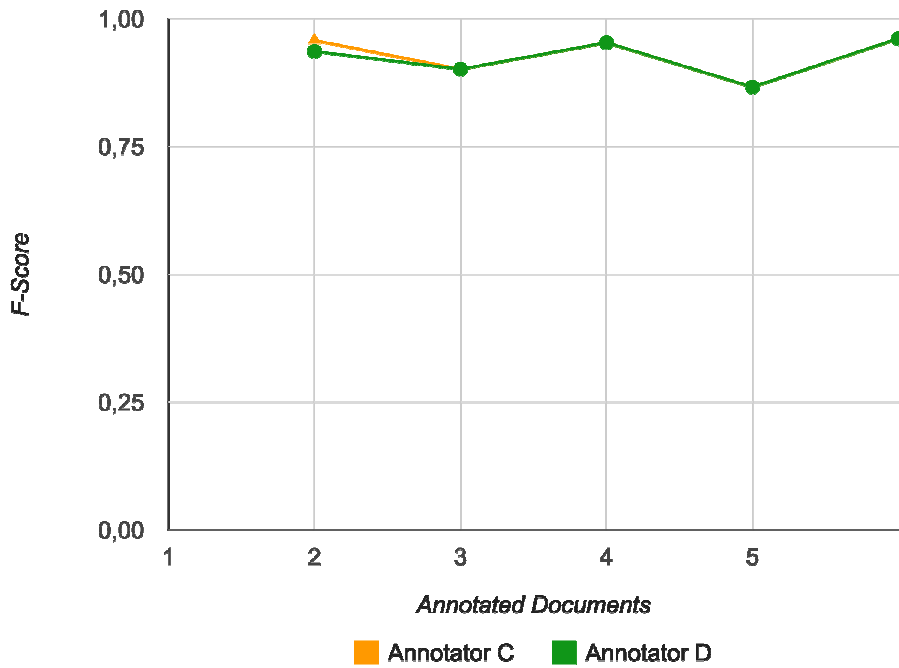


Figure 7: F_1 Score Regarding Suggestions for Text Segmentation

As the results of this experiment show, a custom GUI for annotating TS information is worthwhile. In our experiment the annotation speed could be improved by a factor of 2.3. Generating TS suggestions gives an additional speed up.

4.2. Part-of-Speech Annotation Layer

Table 2 shows the average time spent to annotate POS tags. The difference between using Webanno and the QPT are significantly: The QPT reduces the annotation time by a factor of 2.5 on average.

Table 2: Average Annotation Time for Annotating a Document with Part-of-Speech Tags

Webanno		Quick Pad Tagger	
Annotator A	Annotator B	Annotator C	Annotator D
42:19 min	51:15 min	16:59 min	20:49 min

Figure 8 displays the annotation speed for each document and annotator. On average the annotation speed is 0.2 tokens per second higher by using the QPT. Additionally, the annotation speed of Annotator C and D increases better than the annotation speed of Annotator A and B. At the end of the experiment Annotator C and D annotate faster by a factor of 2.5 on average.

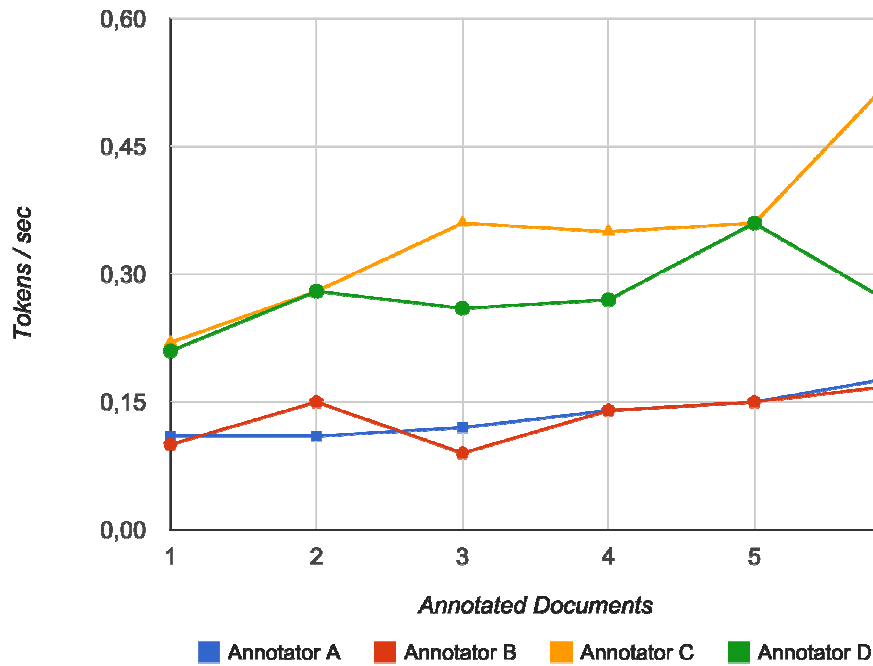


Figure 8: Annotation Speed for Part-of-Speech Tags

Figure 9 shows the accuracy of the POS tag suggestions. The accuracy of the suggestions increases from document to document. At the end of the experiment the accuracy reaches 90 percent. The increasing accuracy limits the task of annotating POS to reading and verifying suggestions. This limitation improves annotation speed further (c. f. increasing annotation speed of Annotator C and D in Figure 8).

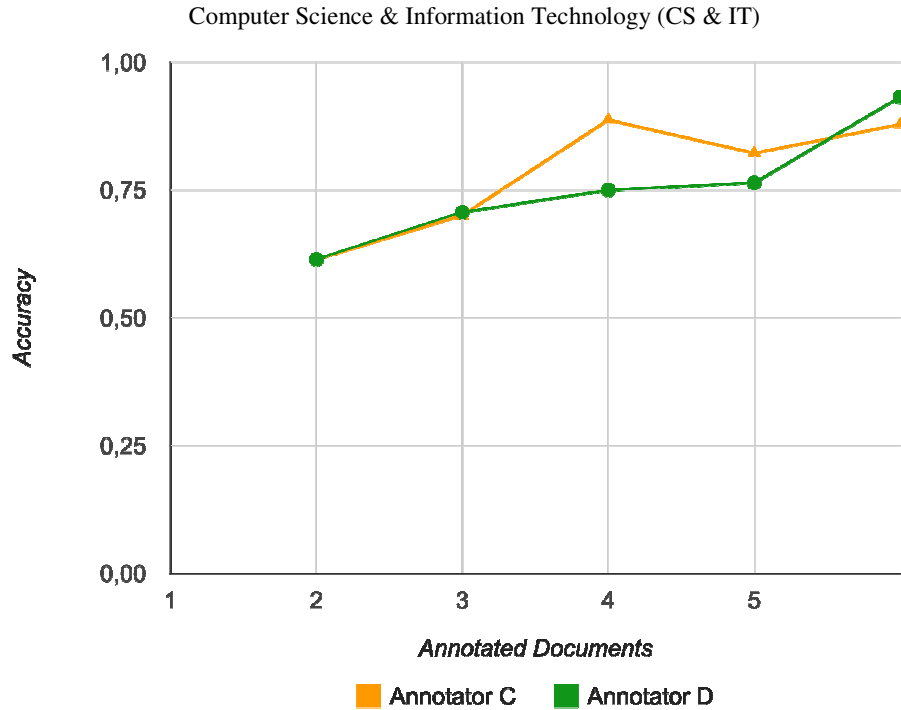


Figure 9: Accuracy Regarding Suggestions for Part-of-Speech Tags

The results of this experiment show that a custom GUI for annotating POS tags is worthwhile. Due to the custom GUI for the annotation layer the annotation speed could be improved by a factor of 2.5. The provided suggestions improve annotation speed further.

4.3. Named Entity Annotation Layer

Table 3 shows the average time spent to annotate NEs. On average the QPT reduces the annotation time by a factor of 2.4.

Table 3: Average Annotation Time for Annotating a Document with Named Entities

Webanno		Quick Pad Tagger	
Annotator A	Annotator B	Annotator C	Annotator D
7:05 min	5:41 min	2:42 min	2:42 min

Figure 10 shows the annotation speed for each document and annotator. On average the annotation speed is 0.8 tokens per second higher by using the QPT. The annotation speed for both tools stays more or less the same. When the third and fourth document are annotated, the annotation speed drops for both tools.

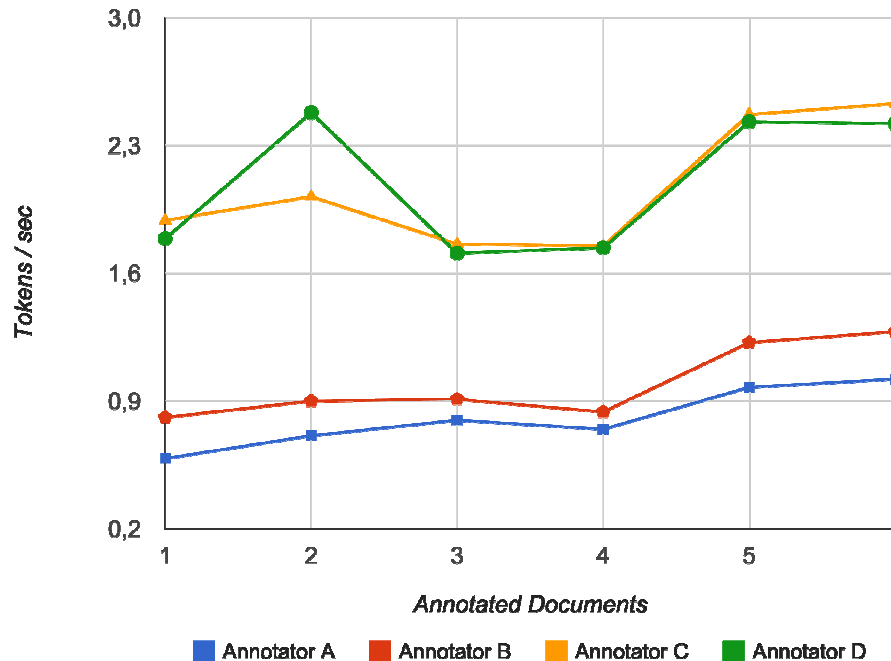


Figure 10: Annotation Speed for Named Entities

The drop of annotation speed for the third and fourth document has following reason: The annotators struggled independently to annotate some NEs and they asked for advice. The drop of annotation speed for Annotator C and D is more distinctive because on average it took not so much time to annotate the whole document (c. f. Table 3).

Additionally, in this experiment the annotation of the QPT was not influenced by the suggestions. Because of the small corpus size the F_1 score of the NE suggestions was zero. The suggestions made by the QPT were not helpful to annotators and in most cases the suggestions had to be removed.

In conclusion, the results of this experiment show a custom GUI for annotating NEs improves the annotation speed significantly (factor of 2.4). The annotation speed of the QPT outperforms the annotation speed of Webanno even with faulty suggestions. The F_1 score of NE suggestions was zero and did not boost the annotation speed further.

4. CONCLUSION AND FUTURE PROSPECTS

We presented the QPT and compared the QPT with Webanno regarding annotation speed. In our three experiments we showed that the QPT outperforms Webanno in terms of annotation speed because of two reasons:

1. The QPT provides for each annotation layer a specific GUI based on a keyboard input method. Webanno provides a generic GUI for all annotation layers based on a mouse input method. QPT's specific input methods are more efficient than Webanno's generic input method.
2. The version of Webanno used for testing does not provide suggestions to the annotator. The design of the QPT includes providing suggestions right from the beginning which enables another speed up regarding annotation speed.

Normally, users have to learn to use specialized GUIs which often takes a long time. Despite the specialized GUI, Annotator C and D were able to learn to use the QPT very quickly during our experiment. This verifies that the GUI of the QPT has a high efficiency and effectiveness [25]. Currently, the QPT is used to develop a large domain specific annotated corpus. In future work we will address Gamification elements [19] to improve the user experience furthermore. We expect that an improved user experience increases annotation speed further. Additionally, an improved user experience should keep the annotation quality high.

REFERENCES

- [1] Roman Klinger, Corinna Kolarik, Juliane Fluck, Martin Hofmann-Apitius, and Christoph M. Friedrich. Detection of IUPAC and IUPAC-like Chemical Names. *Bioinformatics*, 24(13):i268–i276, 2008.
- [2] Eugenie Giesbrecht and Stefan Evert. Is Part-of-Speech Tagging a Solved Task? An Evaluation of POS Taggers for the German Web as Corpus. In *Proceedings of the 5th Web as Corpus Workshop, WAC5*, pages 27–35, 2009.
- [3] Neil Barrett and Jens H. Weber-Jahnke. Building a biomedical tokenizer using the token lattice design pattern and the adapted Viterbi algorithm. *BMC Bioinformatics*, 12(S-3):S1, 2011.
- [4] Haibin Liu, Tom Christiansen, William A. Baumgartner, and Karin Verspoor. BioLemmatizer: a lemmatization tool for morphological processing of biomedical text. *Journal of biomedical semantics*, 3, 2012.
- [5] Jeffrey P Ferraro, Hal Daumé, Scott L DuVall, Wendy W Chapman, Henk Harkema, and Peter J Haug. Improving performance of natural language processing part-of-speech tagging on clinical narratives through domain adaptation. *Journal of the American Medical Informatics Association*, 2013.
- [6] Tim Rocktäschel, Torsten Huber, Michael Weidlich, and Ulf Leser. WBI-NER: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 356–363, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.
- [7] Melanie Neunerdt, Bianka Trevisan, Michael Reyer, and Rudolf Mathar. Part-Of-Speech Tagging for Social Media Texts. In Iryna Gurevych, Chris Biemann, and Torsten Zesch, editors, *Language Processing and Knowledge in the Web*, volume 8105 of *Lecture Notes in Computer Science*, pages 139–150. Springer Berlin Heidelberg, 2013.
- [8] Aljoscha Burchardt, Katrin Erk, Anette Frank, Andrea Kowalski, and Sebastian Pado. SALTO: A Versatile Multi-Level Annotation Tool. In *Proceedings of LREC-2006*, 2006.
- [9] Philip V. Ogren. Knowtator: A Protégé plug-in for annotated corpus construction. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, NAACL-Demonstrations '06*, pages 273–275. Association for Computational Linguistics, 2006.
- [10] Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 1–6, 2013.
- [11] Todd Lingren, Louise Deleger, Katalin Molnar, Haijun Zhai, Jareen Meinzen-Derr, Megan Kaiser, Laura Stoutenborough, Qi Li, and Imre Solti. Evaluating the impact of pre-annotation on annotation speed and potential bias: natural language processing gold standard development for clinical named entity recognition in clinical trial announcements. *Journal of the American Medical Informatics Association*, 2013.
- [12] Hrafn Loftsson, Jökull H. Yngvason, Sigrún Helgadóttiry, and Eiríkur Rögnvaldssonz. Developing a PoS-tagged corpus using existing tools. In *Proceedings of “Creation and use of basic lexical resources for less-resourced languages”*, workshop at the 7th International Conference on Language Resources and Evaluation, 2010.
- [13] Karén Fort and Benôit Sagot. Influence of Pre-annotation on POS-tagged Corpus Development. In *Proceedings of the Fourth Linguistic Annotation Workshop, LAW IV '10*, pages 56–63. Association for Computational Linguistics, 2010.

- [14] Andreas Mengel and Wolfgang Lezius. An XML-based representation format for syntactically annotated corpora. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 121–126, 2000.
- [15] N. F. Noy, M. Crubézy, R. W. Ferguson, H. Knublauch, S. W. Tu, J. Vendetti, and M. A. Musen. Protégé-2000: An Open-Source Ontology-Development and Knowledge-Acquisition Environment. *AMIA Annual Symposium Proceedings*, pages 953+, 2003.
- [16] Seid Muhie Yimam, Richard Eckart de Castilho, Iryna Gurevych, and Chris Biemann. Automatic Annotation Suggestions and Custom Annotation Layers in WebAnno. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. System Demonstrations*, page (to appear), June 2014.
- [17] David M. Lane, H. Albert Napier, S. Camille Peres, and Aniko Sandor. Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts. *Int. J. Hum. Comput. Interaction*, 18(2):133–144, 2005.
- [18] Richard C. Omanson, Craig S. Miller, Elizabeth Young, and David Schwantes. Comparison of Mouse and Keyboard Efficiency. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, pages 600–604, September 2010.
- [19] Sebastian Deterding, Miguel Sicart, Lennart Nacke, Kenton O’Hara, and Dan Dixon. Gamification. Using Game-design Elements in Non-gaming Contexts. In *CHI ’11 Extended Abstracts on Human Factors in Computing Systems, CHI EA ’11*, pages 2425–2428, New York, NY, USA, 2011. ACM.
- [20] Torsten Zesch, Christof Müller, and Iryna Gurevych. Extracting Lexical Semantic Knowledge from Wikipedia and Wiktionary. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*, electronic proceedings, May 2008.
- [21] Welcome to Apache OpenNLP. <https://opennlp.apache.org/>.
- [22] The Stanford Natural Language Processing Group . <http://nlp.stanford.edu/software/corenlp.shtml>.
- [23] Slav Petrov, Dipanjan Das, and Ryan McDonald. A Universal Part-of-Speech Tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC’12)*, may 2012.
- [24] Ulrich Heid, Helmut Schmid, Kerstin Eckart, and Erhard Hinrichs. A Corpus Representation Format for Linguistic Web Services: The D-SPIN Text Corpus Format and its Relationship with ISO Standards. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10)*. European Language Resources Association (ELRA), may 2010.
- [25] Jeffrey Rubin and Dana Chisnell. *Handbook of Usability Testing: Howto Plan, Design, and Conduct Effective Tests*. Wiley Publishing, 2nd edition, 2008.

AUTHOR INDEX

- Abass A. Olaode* 63
AbdelRahim A. Elmadany 117
Abdullah Gani 01
Ahmed H. Aliwy 89
Albert Zündorf 131
Ali M. Aseere 85
Anjum Naveed 01
Ayad R. Abbas 89
- Bakhtiar M. Amen* 27
Bodo Kraft 131
- Catherine A. Todd* 63
- Darwin Muñoz* 111
Davood Falahati 53
- Golshah Naghdy* 63
- Hojat Cheraghi* 53
Hong Yan 73
- Jessica C. Ramírez* 111
Joan Lu 27
Juanying Lin 73
- Kai Barkschat* 131
Kazem Ghalamchi 53
Kok-Leong Koong 41
- Lay-Ki Soon* 41
Leanne Chan 73
- Marc Schreiber* 131
Md Whaiduzzaman 01
Mervat Gheith 117
- Qiang Huang* 13
QiRui Huang 13
- Sardasht M. Mahmood* 27
Sherif M. Abdou 117
Su Mon Khine 99
Su-Cheng Haw 41
- XiaoMeng Zhou* 13
Yadana Theinc 99
Yongbin Bai 13
Yuji Matsumoto 111