ISSN 2602-8549

SRI LANKA ASSOCIATION FOR ARTIFICIAL INTELLIGENCE

SLAAI - International Conference on Artificial Intelligence

Volume 1

Proceedings of the 13th Annual Sessions 31st October 2017

Sri Lanka Association for Artificial Intelligence – 2017

SLAAI International Conference on Artificial Intelligence
31st October 2017
Moratuwa
Sri Lanka

© 2017 October

All rights reserved. No part of this publication may be reproduced or quoted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system, without permission in writing from the Sri Lanka Association for Artificial Intelligence.

The materials in this publication have been supplied by the respective authors, and the views expressed remain the responsibility of the named authors. The statements and opinions stated in these publications do not necessarily represent the views of the Sri Lanka Association for Artificial Intelligence.

ISSN 2602-8549

Editors:

Prof. AS Karunananda
Dr. (Mrs.) DDM Ranasinghe
Mrs. WAC Weerakoon

Sri Lanka Association for Artificial Intelligence, Department of Electrical & Computer Engineering, The Open University of Sri Lanka, Nawala, Nugegoda, Sri Lanka.

Contents Page

Message from SLAAI President
Dr. Subha Fernando ii

Message from Conference Chair
Prof. Asoka S Karunananda iii

Keynote Speech
Prof. R P C Janaka Rajapakse iv

Plenary Talk
Dr. Ruwan Weerasinghe v

Ontology Design and Development for University Admission in Sri Lankan Universities
T S D H S Wickramarathna, C Anutariya 1

Mind Uploading with BCI Technology
S Welikala, A S Karunananda 7

Deep Learning Techniques for Speech Recognition and Audio Transcription for Content Captioning
D G J B Wimalarathne, S C M de S Sirisuriya 16

Dictionary-Based Machine Translation System for Pali to Sinhala
R M M Shalini, B Hettige 23

On Optimizing Deep Convolutional Neural Networks by Evolutionary Computing
M U B Dias, D D N De Silva, S Fernando 29

Clustering-based Augmented Tabu Search for Energy Minimizing Vehicle Routing Problem (EMVRP)
G W H H P Rathnakumara, T D Rupasinghe 38

Predicting Floods in North Central Province of Sri Lanka using Machine Learning and Data Mining Methods
H Thilakarathne, K Premachandra 44

A Genetic Algorithmic Approach for Optimizing Supply Chain Network Designs for Retail Distribution Networks
I U Munasinghe, T D Rupasinghe 50

Knowledge Discovery with Data Mining for Predicting Students' Success Factors in Tertiary Education System in Sri Lanka
K T S Kasthuriarachchi, S R Liyanage 56

A Simulation-based Analytical Approach to Enhance Distribution Networks in Pharmaceutical Supply Chains
Q J D N N Rathnasiri, T D Rupasinghe 62

-Message from SLAAI President -

The SLAAI International Conference on Artificial Intelligence, organized by Sri Lanka Association for Artificial Intelligence (SLAAI), is the 13th annual gathering of researchers, practitioners, students who are interested in Artificial Intelligence (AI) in Sri Lanka. SLAAI was formally established in the year 2000 as the first AI research group in Sri Lanka, with the vision of popularizing AI among the academics, researchers, practitioners to promote AI Industry in the country.

AI has gone through huge expectations followed many disappointments. We have seen the rise and fall of expert systems, neural networks, and the use of statistical models for reasoning. Once again, we are in the era of heightened expectations regarding AI. We now have Siri, Cortana, self-driving cars and deep learning systems that promise an outstanding level of machine intelligence, where today's industry can highly benefit.

Under such a heightened environment, the role of SLAAI is very vital in establishing AI-Industry in Sri Lanka. To reach its objectives, SLAAI has organized many workshops to disseminate the AI knowledge to the society, conducted several Meet-ups, opening forums for practitioners to reach the IT industry, and several competitions inspiring students to join hands with SLAAI. SLAAI should be very proactive in disseminating the knowledge to the community and linking practitioners with IT-industry in the future as well. Align to the very purpose; this conference has been themed to "Progress in AI and Increase in its business adoption", allowing many researchers to discuss their findings applicable to the Industry.

The conference marks the final event of the year where we provide a focal forum for all our researchers, practitioners to share their knowledge, expertise, findings with the community of AI. As the president of SLAAI, I would like to thank the panel of judges, session chairs, panel of reviewers, the program committee, authors and presenters, generous sponsors, administrative staff of University of Moratuwa who wholeheartedly supported the conference, staff of Faculty of Information Technology those who organized logistics and many others who contributed to its success.

Subha Fernando, Dr.Eng
President, Sri Lanka Association for Artificial Intelligence
31.10.2017

-Message from Conference Chair -

It gives me great pleasure to send this message to SLAAI-International Conference on Artificial Intelligence of the 13th Annual Sessions of Sri Lanka Association for Artificial Intelligence (SLAAI). Formal establishment of SLAAI had taken place in early 2000 as the successor to the first AI research group, AIRLK, in Sri Lanka. SLAAI was established with the objective to promote AI Education and Research in Sri Lanka. For more than last 17 years, SLAAI has immensely contributed to promote AI in academia and industry. Among other services, Annual Sessions of SLAAI has offered the forum for presenting and publishing AI research in Sri Lanka, and dissemination of such knowledge to a wider audience comprising of academia, industry and the student community.

SLAAI – International Conference on Artificial Intelligence has expanded its horizon to attract both local and international research papers in AI. In 2017, SLAAI has noted an increase of submission of research papers covering AI research conducted locally and internationally through some collaboration with Sri Lanka. We are pleased to notice that the papers submitted to SLAAI covers a broad spectrum AI technologies including Ontology, Deep Learning, Machine Translation, Multi Agent Systems, Evolutionary Computing, and Convolutional Neural Networks, Having gone through the double-blind peer review process, SLAAI has selected 60% of papers for presentation and publication. According to the practice of SLAAI, subsequent to the presentation at the conference, improved papers will be published in the Proceedings.

I take this opportunity to express my sincere thanks to the keynote speaker, plenary speaker, session chairs, reviewers, panel of judges, authors of the papers, program committee and council members of SLAAI for their untiring effort to make this event a success. A special thank goes to Prof. Ananda Jayawardana, Vice-Chancellor of University of Moratuwa, Eng. PM Karunathne, Dean of Faculty of Information Technology, and the Staff for their support to host the Conference in this year at University of Moratuwa.

Prof. Asoka S Karunananda
Conference Chair / SLAAI
31.10.2017

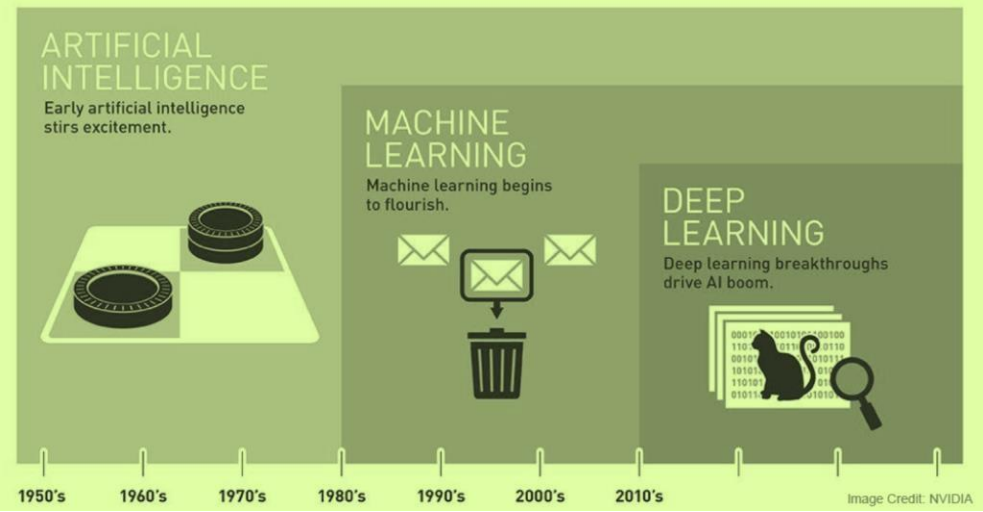
-Keynote Speech-

Artificial Intelligence in New Media and Entertainment

Prof. Janaka Rajapakse
Graduate Institute of Animation and Film Art
Tainan National University of the Arts, Tainan, Taiwan.

Abstract

Artificial Intelligence (AI) technology is revolutionizing the development process. Learn how major intelligence innovation platforms currently available from deep learning can help you solve major design/development problems and bring real value to your business. Architecture Engineering & Construction, Manufacturing, and Automotive are just some of the industries experiencing the most exciting shift in the design process in over a decade.



And the graphic hardware programing has proven to be effective at solving some of the most complex problems in the entertainment industry. It started out as an engine for simulating human imagination which produces up the realistic virtual worlds of video games and Hollywood films. Today, virtual reality (VR) is the latest hype for game developers for engaging audiences from content consumption perspective. However, the AI technology has much broader applications. As results of the latest advancements in AI technologies are becoming approachable for engineers, architects and visualization specialists. Meanwhile VR/AR and 3D printing are enabling new workflows in design, prototyping, visualizing and marketing products. Are you ready for this technology explosion? What tools and AI-ready technology can amuse you and your employees get entertainment and safely? This speech will focus on different available virtual-reality (VR) and 3D printing technology, with the purpose of implementing AI into their workflows. We will have an in-depth discussion both about our developments, as well as about future perspectives.

-Plenary Talk-

AI in Your Pocket: Myth vs Reality

Dr. Ruwan Weerasinghe
University of Colombo School of Computing, Sri Lanka.

Abstract

The once spurned technology 'artificial intelligence' has made a comeback! And with a vengeance! AI is now the darling of not just some techies, but mainstream CXOs! What caused this dramatic shift and how realistic is the expectation that society now bestows on it?

This talk will explore some of the reasons for the current resurgence of interest in AI, and try to disentangle the hype from the reality 'on the ground'. It will also open up a conversation on what as AI professionals, our own opinion on this should be, as we encounter it at business meetings and the very corridors of power.

Ontology Design and Development for University Admission in Sri Lankan Universities

T. S. D. H. S. Wickramarathna[1], C. Anutariya[2]
[1]Uva Wellassa University, Badulla, Sri Lanka.
[2]Asian Institute of Technology, Pathumthani, Thailand.

Abstract

University Admission process is a complex and time-consuming task which requires experts' advice in all the stages. Since the different stages belong to different organizations, it is rare to find experts having an overall knowledge of the process, from whom anyone can get a complete guide. Local universities in Sri Lanka are fully government funded but able to admit only around 10% of the qualified students. This leads to a high competition in admission but proper guidance in the admission process is still not established. A research has been conducted to develop a recommendation system to encapsulate the complexity of university admission process and help students to take the most suitable decision. In this paper, design and development of the Ontology, which is the knowledge base for the recommendation system is discussed. This Ontology model the undergraduate admission scenario in Sri Lanka, together with the career path mapping, so that students can far see their future opportunities before taking decisions on advanced level subject selection and university application. Ontology was developed using Protégé and was tested by running SPARQL queries. SPARQL query results confirmed that the ontology design is comprised with the real world.

Keywords: *University Admission, Ontology with Protégé, Career Path Mapping, SPARQL Queries.*

1. Introduction

University admission is a complex decision process that goes beyond simply matching test scores and admission requirements. Whether a student chooses a right university and right degree program, and conversely whether a university chooses a right student, decides the success of both institute and the students in carrying through the education. Admission and enrollment studies are generally focused on the university or collage perspective, and studies based on secondary school subject selection perspective is limited.

Higher education at local universities in Sri Lanka is fully government funded. However, due to lack of budget allocation and limited facilities, local universities are able to admit only around 10% of the qualified students [2]. This leads to a high competition in the admission, but proper guidance in admission process is still not established. Therefore, it is crucial to detect the influential factors in order to be able to reduce the failures of qualified students in admission. Data is available located in different data stores, with different names for identical entities. For this reason, the entities of the university domain and the relations between them must be defined.

There are around 100 distinct degree streams offered by Universities in Sri Lanka. Each stream has minimum subject requirements from Advanced level subjects while some degrees may have an additional requirement such as Ordinary Level results and aptitude tests. Each degree stream has planned intake for the particular academic year. Universities offer degrees under degree streams through different faculties and departments. Same time these degrees mapped to certain career paths according to the standards set by organizations and companies.

In summary, major problems and difficulties in Sri Lankan University admission system can broadly be identified as follows [3].

- Limited facilities, high competitiveness
- Limited Guidance in A-Level subject selection
- Limited awareness about specific entry requirements in some universities
- Mismatch between labor market supply and demand

- Limited awareness about other territory education options

Therefore, a recommendation system would be much helpful for students as it will encapsulate this complexity and help students to take the most suitable decision. In this study, ontology will be developed, as it will be the knowledge base for the future recommendation system. Ontology was developed using Protégé as Protégé is the most popular tool for ontology editing and developing [8]. This ontology model the undergraduate admission scenario in Sri Lanka together with the career path mapping such that the data can be published by University Grants Commission (UGC), universities and companies to be referred by students to far see their future opportunities. Though this study is limited only to Sri Lankan National university admission system, it can simply be expanded and applied to other countries with minor changes.

2. Related Work

There are several possible approaches to solve the mentioned problem. Considering the limitations and known problems in other approaches, Linked Data approach is selected and Ontology was developed as the knowledge base.

Ontology is the statement of a logical theory. The term Ontology comes from philosophy. It is an explicit specification of a conceptualization. In AI field, it believes that whatever exists can be represented. When the knowledge of a domain is represented in a declarative form, the represented set of objects is called the universe of discourse. These sets of objects along with the relationships described among them are reflected in the representational vocabulary. The knowledge-based program represents knowledge using this vocabulary [12].

In AI ontology is described by defining a set of representational terms. The definitions associate the names of entities in the universe of discourse with human- readable text by describing what the names mean, and formal axioms that constrain the interpretation and well-formed use of these terms [13]. Gruber [12] defines an ontology as explicit formal specifications of the terms in the domain and relations among them. An ontology is consists of definitions of concepts in a domain and relations among these concepts. These definitions allow developing artificially intelligent applications as they are expressed in a machine-interpretable way. Ontologies are considered as the cornerstone of the semantic web. Ontologies will capture the domain knowledge in a generic way in order to provide a clear understanding of a domain. They are referred as the shared conceptualization of a domain since they include the representation of these conceptualizations [7]. Ontologies do not depend on the application that uses them. This independence makes the software and knowledge maintenance easier and enhances the semantic interoperability between applications [8]. Therefore, it can be claimed that ontologies constitute the backbone of the Semantic Web. Ontologies can be used very effectively in different application fields and they provide a specification of a formal common representation of an application field trying to reduce the distance between the way people and machines manage information.

Several software tools are required in the ontology development process [8]. There are number of open-source as well as commercial tools are available for ontology development and they referred as Ontology editors. Some tools can be used throughout the entire ontology development process whereas some tools can be used in several stages of the ontology lifecycle like designing, implementation and maintenance of ontologies. Protégé is a free, open-source ontology and knowledge base editor and framework produced by Stanford University for building intelligent systems. Protégé is used by diverse of users for build knowledge-based solutions in several areas. Protégé provides facility to constructing domain ontologies as well as customized data entry forms to enter data. Classes, variables, class hierarchies, the relationship between classes, variable value restrictions and properties of relationships can be defined using Protégé.

Even though there is no any ontology developed to cover up the entire university admission process, there are few initiatives to model the university by creating ontology for that. Naveen et al. [10] have created a university ontology using Protégé as the tool. They

performed their task in seven steps starting with defining the classes and class hierarchy followed by defining object properties and data properties, identifying the relationships between properties, axioms of ontology and finally the reasoning of ontology. There are some other similar initiatives in a related topic of the university in education domain such as developing university ontology, by Sanjay Malik in 2010 [4]. In their ontology, they have the emphasis on the university employee detail only based on the name of the employee, date of joining , address etc. and ontology based on course, Ling Zeng, et al. 2009 [11] which focus on a particular course to reuse of course for teaching purpose. Ayesha Ameen, et al in [5] developed a university ontology mainly focused on the properties and their restrictions. They also used Protégé as the developing tool.

3. Ontology Design and Development

Protégé is chosen for implementation because it enables the construction of domain ontologies, and customized data entry forms to enter data. Protégé allows the definition of classes, class hierarchies, variables, variable-value restrictions, and the relationships between classes and properties of these relationships [4]. Moreover according to some surveys about most widely used ontology editors and most widely used domain for ontology development.

There are several design strategies introduced for ontology design [7], [9]. However design strategies presented in "Ontology Development 101: A Guide to Creating Your First Ontology" [1] was followed to develop the ontology. That strategy consists of following steps:

- Determine domain and scope of ontology
- Consider reusing existing ontologies
- List important terms in the ontology
- Define the classes and the class hierarchy
- Define properties of classes
- Define facets of the properties
- Create instances

When determining the domain and scope of the ontology, identifying the competency questions is crucial. Competency questions summarize the types of questions that the ontology should be able to provide answers and in here competency questions were designed by referring the information provided in [2],[6] and also from the official websites of University Grants Commission, Examination Department, Ministry of Higher Education.

A. Define the classes

A class definition can be done using different approaches. A top-down development process: In this approach, class definition is started by defining the most general concept in the domain. A bottom-up development process: Specific classes of the domain will start defining first in this approach. Apart from that, a combination development process, which is a combination of both top-down and bottom-up approaches, can also be used.

Thirteen classes; University, ALStream, Faculty, Subject, Department, Course Of Study, Higher Educational Institute, Score Restriction, Eligible Requirements, District, Student, Degree, and Career are identified as the top level classes in the ontology class hierarchy. Figure 1 visualizes the top-level classes of the ontology.

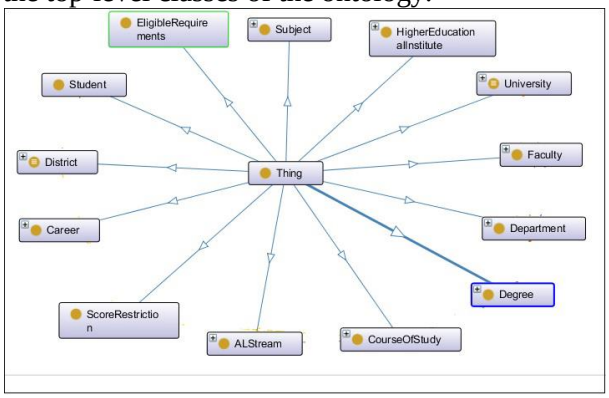


Figure 1: Top Level classes of Ontology

After identifying the top-level classes of the ontology, ontology hierarchy was developed by creating the subclasses. When defining the classes and subclasses, rules and restrictions, which needed to be embedded in the ontology, are also modeled. In the top-level class Subject, it has been identified that there are around 50 A/L subjects offered in Sri Lankan schools and they are categorized into 7 categories. In Sri Lanka, there are five major A/L streams available. The student will fall into one of the A/L Streams based on his subject selection. Not all three-subject combinations are valid. A/L streams are having restrictions in subjects, and student should select the subjects accordingly. The courses of Studies in

the Universities are categorized based on the A/L streams. However, there are some courses of studies that are not restricted only to one A/L stream.

Degree is another top-level class of the ontology. Degrees are grouped based on the type of the degree (eg: BSc, MBBS) offered by the university. There are restrictions about a valid course of studies to obtain a specific degree. For example to obtain a medical degree student must have followed biological science course of study. Career, Student, District, and ScoreRestrictions are other top-level classes. To enter into a certain career, a student must have done a specific degree. For example, if a student wanted to become a medical doctor then he must have followed medical degree. However, there are some career opportunities for students holding any type of degree.

Student is one of the important classes, which will visualize the current user of the system. In order to be eligible to apply for the university, admission student must fulfill some eligibility requirements mentioned by the UGC. Since the district of the student applying for the university take into the consideration districts in Sri Lanka and their minimum scores restrictions for the course of studies are modeled.

Higher Educational Institutes in Sri Lanka can be basically divided into two categories; Degree offering institutes and non-degree offering institutes. Universities are degree-offering institutes. Teacher training colleges, Technical Training colleges etc. categorized under non-degree offering institutes as they are not offering degrees but only diplomas. However, a student cannot enroll into any of the two higher educational institutes simultaneously, regardless of whether it is a degree offering or non-degree offering institute. Major academic division of a university is called Faculty and every university must have at least one faculty. A faculty is composed of several departments and in one faculty, there must be at least one department.

B. Define properties of classes

To construct the relationship between classes data and object property restrictions are added. Table 2 shows the object properties defined in the ontology. Properties link individuals from the domain to individuals from the range. Inverse property of the defined property and also some restrictions added into the properties are demonstrated in the figure.

Data properties are defined in order to visualize the relationship between individuals of the classes and the literal values. Table 1 shows the data properties used in the ontology.

Property restrictions are added to make sure the ontology is representing the correct existing knowledge. Following are some sample existential and cardinality restrictions.

Table1. Data Properties of the Ontology

Property	Domain	Range
hasALAttempts	EligibleRequirements	integer
hasAptitudeTest	Degree	boolean
hasDistrictRank	Student	positiveInteger
hasIslandRank	Student	positiveInteger
hasIntake	CourseOfStudy	positiveInteger
hasDuration	Degree	positiveInteger
hasMinScore	ScoreRestriction	string
hasUniversityRank	University	positiveInteger

- A University should have at least one Faculty
- Faculty should have at least one Department
- Department should offer at least one Degree
- A Degree should belong to at least one-degree stream.
- Medium of instruction of a degree should be at least one language and should have exactly one duration in years, which should be greater than 3.
- Advanced Level Stream is composed of exactly 3 Advanced Level subjects.

Table2. Object Properties of the Ontology

Property	Domain	Range
hasAdmissionRequirement	CourseOfStudy	ALStream , ScoreRestriction
hasALRequirement	CourseOfStudy	ALStream
hasScoreRequirement	CourseOfStudy	ScoreRestriction
hasDegree	Department	Degree
hasDistrict	Student	District
hasDivision	University	Faculty, Department
hasDepartment	Faculty	Department
hasFaculty	University	Faculty
hasFulfill	Student	EligibleRequirement
hasOLRequirement	Degraa	OLSubject
hasQualification	Career	Degree
hasSubject	ALStream	ALSubject
hasSubjectCategory	ALStream	ALSubject
offerDegree	University	Degree

C. Define Instances

In order to check the correctness of relationships and confirm the strength of ontology to planned Competency Questions, set of individuals under each class are defined. After creating, the instances reasoner were synchronized and tested to check whether it correctly performed according to the defined rules. All the rules were correctly reasoned. For example Chemistry individual were created under Science subject category and once the reasoner synchronized, it automatically identified that Chemistry is a subject of Physical Science, Biological Science Agri Science and Technology Science A/L streams. Moreover, it identified the degrees, which Chemistry is a subject requirement.

4. Experimental Results

Correctness of class hierarchies and the restrictions were tested by adding test classes to the ontology. All the tests were successful and inferred hierarchy correctly listed the classes. For example, to check the correctness of the A/L streams, few test classes were added and tested whether the reasoner inferred it correctly. Figure 2 shows that the reasoner correctly identified the Test class should fall in to Arts Stream (Highlighted in light orange color).

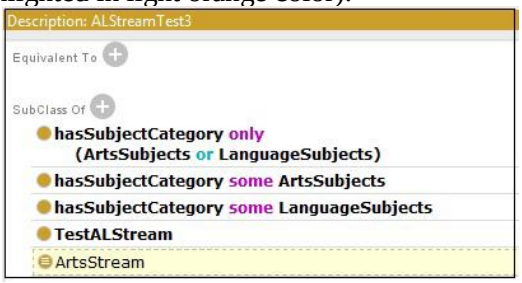


Figure 2: Inferred hierarchy test of A/L Stream

The ontology or the knowledge base can then be used to retrieve answers for several intelligent questions. Reasoners and SPARQL query engines can be used to query and grab the desired answers from the system. Among the question pool created by the questions sent by the A/L and O/L students, 50 questions were selected and tested. All the queries returned the expected answer. Some answers were not complete due to lack of data embedded to the ontology, but the answer was correct according to the available data. Following are some sample scenarios that demonstrate the knowledge retrieval from the developed knowledge base.

Scenario 1: Student wants to enroll into any course of study under Biological Science course of study. He wants to know the possible list of subjects he can follow.

```
SPARQL query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX ug:<http://www.semanticweb.org/harshani/ontologies/2015/10/universityAdmission#>

SELECT DISTINCT ?subject
WHERE{?courseofstudy rdf:type ug:BiologicalScienceCourseOfStudy.
?courseofstudy ug:hasALRequirement ?alstream.
?alstream ug:hasSubject ?subject}

subject
Chemistry
Biology
Agriculture
Physics
```

Figure 3: SPARQL Query and Output: Subject requirement for A/L streams

Scenario 2: Student wants to know the possible degrees he can follow in order to become a Software Engineer.

```
SPARQL query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX ug:<http://www.semanticweb.org/harshani/ontologies/2015/10/universityAdmission#>

SELECT ?degree
WHERE {ug:softwareEngineer ug:hasQualification ?degree}

degree
'BSc in Computer Engineering'
'BSc in Computer Science'
'BSc in Information Technology'
'BSc in Information Technology & Management'
```

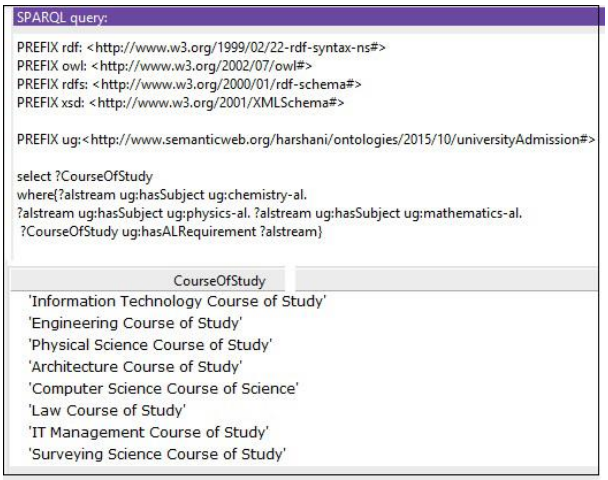
Figure 4: SPARQL Query and Output: Degree qualification for a career

Scenario 3: Student wants to know the degrees which are having Aptitude test as additional entry requirements.

```
SPARQL query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX ug:<http://www.semanticweb.org/harshani/ontologies/2015/10/universityAdmission#>

SELECT ?degree
WHERE {?degree ug:hasApptitudeTest "true"^^xsd:boolean}

degree
'BSc in Computer Engineering'
'BSc in Information Technology & Management'
'BSc in Computer Science'
```

Figure 5: SPARQL Query and Output: Degrees having Aptitude tests

Scenario 4: If a student has already done Chemistry, Physics and Combined Mathematics for his A/Ls what are the course of studies he can enroll?



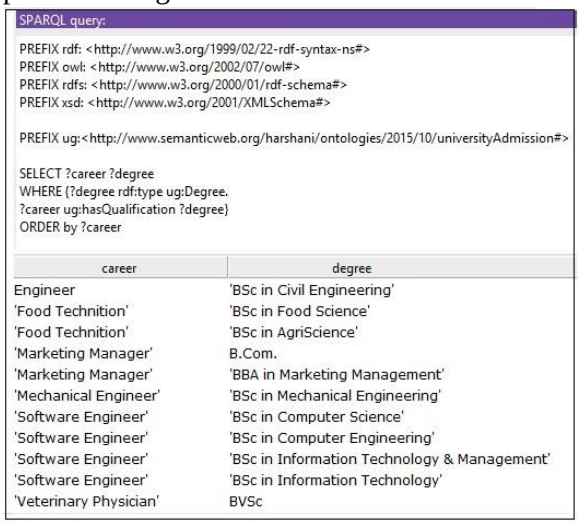
SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX ug:<http://www.semanticweb.org/harshani/ontologies/2015/10/universityAdmission#>

select ?CourseOfStudy
where{?alstream ug:hasSubject ug:chemistry-al.
?alstream ug:hasSubject ug:physics-al. ?alstream ug:hasSubject ug:mathematics-al.
 ?CourseOfStudy ug:hasALRequirement ?alstream}
```

CourseOfStudy
'Information Technology Course of Study'
'Engineering Course of Study'
'Physical Science Course of Study'
'Architecture Course of Study'
'Computer Science Course of Science'
'Law Course of Study'
'IT Management Course of Study'
'Surveying Science Course of Study'

Figure 6: SPARQL Query and Output: Course of studies for A/L subjects

Scenario 5: Student wants to know what the possible career opportunities are if he obtains a particular Degree.



SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX ug:<http://www.semanticweb.org/harshani/ontologies/2015/10/universityAdmission#>

SELECT ?career ?degree
WHERE {?degree rdf:type ug:Degree.
?career ug:hasQualification ?degree}
ORDER by ?career
```

career	degree
Engineer	'BSc in Civil Engineering'
'Food Technition'	'BSc in Food Science'
'Food Technition'	'BSc in AgriScience'
'Marketing Manager'	B.Com.
'Marketing Manager'	'BBA in Marketing Management'
'Mechanical Engineer'	'BSc in Mechanical Engineering'
'Software Engineer'	'BSc in Computer Science'
'Software Engineer'	'BSc in Computer Engineering'
'Software Engineer'	'BSc in Information Technology & Management'
'Software Engineer'	'BSc in Information Technology'
'Veterinary Physician'	BVSc

Figure 7: SPARQL Query and Output: Career opportunities for a Degree

5. Conclusion

University Admission process is a complex and time-consuming task which requires experts' advice in all the stages. All the required information are available but scattered in the web. There is no proper linking in this available information hence reduce the value of it. Not only the lack of interconnectedness but also this information is presented in a format which only human users can consume. Computer agents or applications cannot understand this information so that data is isolated.

Developing a recommendation system to guide the students in this complex process was the main objective of this research and Ontology was designed and developed as the knowledge base of the system. It was designed and implemented using the ontology editor Protégé. Correctness of the Ontology was tested by running several SPARQL queries. Around 50 queries were tested and all the queries returned the expected results and it confirmed that the ontology design is comprised with the real world.

References

[1] Noy, N. F., & Mcguinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.

[2] M.Amilajith, N. L. (2012). Advance level, sri lankan high school curriculum and university admission. Hanyang University, Institute of Educational Technology.

[3] Tharmaseelan, N. (2012). Tertiary education in sri lanka: Issues and challenges. Bulgarian Journal of Science and Education Policy, 1(1), 173–190.

[4] Malik, S. K. (2010). Developing an university ontology in education domain using protégé for semantic web. International Journal of Science and Technology, 2(9), 4673–4681

[5] Ayesha ameen, P. R., Khalel khan. (2012). Construction of university ontology. World Congress on Information and Communication Technologies.

[6] Ghanashyam Rout, S. M., Sreedhar Madichetty, & Pattnaik, S. (2012). An efficient wavelet based automation process for e-admission in universities of odisha. International Journal of Computer Applications, 47(22).

[7] Gruber, T. (1993). Towards principles for the design of ontologies used for knowledge sharing. International journal human-computer studies, 43, 907–928.

[8] Kapoor, B., & Sharma, S. (2010). A comparative study ontology building tools for semantic web applications. International journal of Web Semantic Technology (IJWesT), 1(3).

[9] K. OHara, C. B. (2007). Knowledge representation with ontologies: present challenges future possibilities. International journal human-computer studie, 65, 907–928.

[10] Naveen Malviya, S. S., Nishchol Mishra. (2011). Developing university ontology using protégé owl tool: Process and reasoning. International Journal of Scientific Engineering Research, 2(9).

[11] Zeng, L. (2009). Study on construction of university course ontology: Content, method and process. International Journal of Science and Technology, 2(9), 4673–4681.

[12] Gruber, T. (2009). Ontology. Encyclopedia of Database Systems.

[13] T. Segaran, J. T., C. Evans. (n.d.). Programming the semantic web. International journal human computer studies.

Mind Uploading with BCI Technology

S. Welikala[1], A. S. Karunananda[2]
[1,2]Faculty of Information Technology, University of Moratuwa

Abstract

Mind uploading has been an emerging field of Artificial Intelligence. This area has been primarily influenced by the research and developments in Brain Computer Interfacing (BCI). This paper presents a research into BCI and its influence on mind uploading. In this context first, a description of the essentials of brain waves, BCI technology, and analysis and processing of brain waves are presented. Secondly, it proceeds to discuss the mind uploading technology with reference to EEG based brain waves analysis through BCI. The research has revealed the recent developments in mind uploading, its applications such as data backup, immortality and uploaded astronauts. More importantly, it has discovered legal and social concerns of mind uploading as a matter, which has not been given adequate attention, despite the numerous developments in the technological landscape of mind uploading research.

Keywords: *Mind Uploading, Brain Waves Analysis, Brain Computer Interfacing, Data Backup*

1. Introduction

Mind Uploading is an emerging field, which is commonly misconcepted as a mere scientific fiction, yet the potential of the state of art was hidden for centuries. Mind uploading is the method of copying the organic brain and transferring it to a computer chip [1].When the copy is being made, the computer can run a simulation model so that the copy can respond the same way as the initial organic brain. People are born to this world had then at the very end, all those who were born, die. When they die their core values and knowledge dies along with them. Only their doings and publications remain. What about the important work they were doing before dying, what about the knowledge areas they contained in their amazing brains, which were not revealed to the world? If we somehow can preserve their knowledge values, the world indeed would be a much more intelligent place. Mind Uploading therefore provides hope to the above-mentioned issue since these amazing minds can indeed be recorded and preserved for future use with the aid of mind uploading.

This area of knowledge is an inter-disciplinary study connecting many areas relating to neuro-science. It involves brain imaging, neuroscience, computer science, AI, psychology philosophy and many more. Earlier this study area was a seemingly science-fictional subject. Later scientists implemented researches about these fictional suggestions to run experimental and theoretical research programs. This paper presents our research into the field of mind uploading, its applications and unattended dimension of mind uploading.

Rest of the paper is organized as follows. Section 2 talks about BCI technology. Section 3 is on Mind uploading. In Section 4, we describe the Blue Brain project as the most cited mind uploading initiative to date. Section 5 describes some major applications of mind uploading. Section 6 presents the challenges or concerns related to mind uploading research. Section 7 is discussion to highlight the key finding of the research. Finally, Section 8 concludes the paper.

2. Essentials of BCI

Before moving on to the study of Mind Uploading or Whole Brain emulation, it is necessary to follow up the history, which led from BCI to the development of Whole Brain Emulation. The findings are as follows.

In 1875, the British Physician Richard Caton was the first to record electrical impulses emitted due to brain activity. Hans Berger, German Psychiatrist recorded these signals from the scalp in 1929 [1]. The method of Brain Computer Interfacing was first introduced in the late 70s, but the usage of this technology was limited and set aside due to the costliness of the equipment that are needed and the complex methodology that needed to be followed in the procedure. Today due to the development of technology, there exist low cost BCI devices. Examples for such cost effective equipment are the "NeoruskyMinwave device" and the "Emotive EPOC headset"[2].

A. BCI (Brain Computer Interfacing) Definition

BCI is the systematic way that measures and uses the signals of the brain and converts them into actions by using a specially programmed computer systems and equipment [1]. It can be stated that BCI is the accepted method for a brain and a machine to communicate with each other efficiently.

Even though BCI enables interactions with the brain, the brain does not use any of its usual neural systems in order to pass brain information. or its normal output channels[1].In other words humans can control machines or synthetic equipment without and muscular activity. Simply the brain activities are translated into commands and control signals [3].

B. BCI System

The Brain Computer Interfacing methodology consists of two major controllers. They are the user and the BCI system. The user who has an organic brain makes the brain signals from their mind activities and the BCI system converts these generated signals into commands or stimulations that can control synthetic systems or machines [4].

There are two basic categories to record brain signals for BCI. They are the noninvasive record methodologies and the invasive record methodologies. These methodologies are the invasive system and the noninvasive system.

The invasive system is the hard and the difficult way. In this method the brain is reached by piercing the human scalp with a needle and sensors are implanted in the brain [5]. Examples for such invasive methods are ElectroCorticoGraphy [ECoG] and MicroElectrodeArrays [MEAs][6].

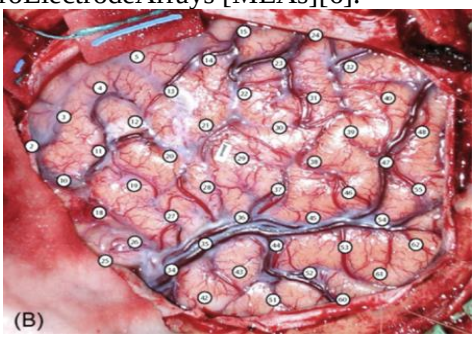


Figure 1: ECoG Sensor Placements

The noninvasive record methodologies include recording of electroencephalography [EEG], functional magnetic resonance imaging [fMRI], positron emission tomography [PET], infrared imaging [IR], near infrared spectroscopy [NIRS] , fetal magneto encephalography [fMEG] and single photon emission computed tomography [SPECT] [1].

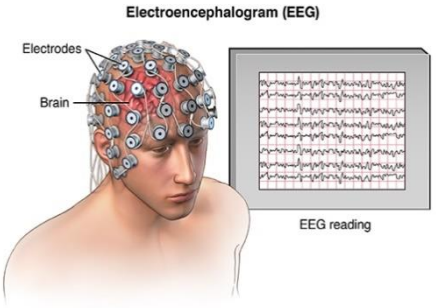


Figure 2: EEG Sensor Placements

One of the efficient devices to acquire such brain signals is the EEG Emotiv EPOC neuron headset. The headset gathers signals from the tissue of the scalp.

C. The Headset

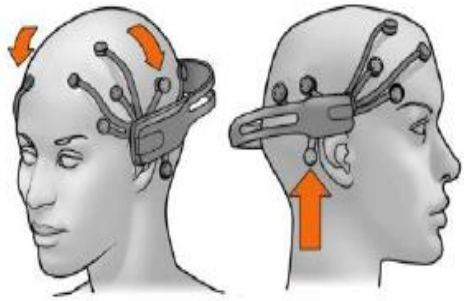


Figure 3: Outlook of the headset

The Emotive EPOC headset is one of the commercially available systems for the general public. It is cost effective and is available for both general consumers as well as researches.

The EPOC Neuron headset consists of 14 electrodes. There exists another 2 extra electrodes that are used as points to mark specific spots on the brain. The other electrodes are placed systematically as per the technology followed [4].

The above electrodes are made out from metal covered in a plastic base. It is wetted with saline water before placing on the scalp for signal acquisition[5].

The number of brain signal samples taken per second is 2048Hz. The resolution that can be obtained from this headset is 14 bits. The bandwidth of the signals that can be handled is 0.2-45HZ [8].

D. Signal Processing Method

Before any processing is done to the above acquired signals, the noise or the interruptions in the acquired brain waves are duly removed. This

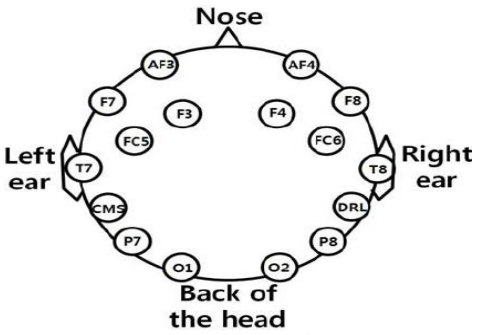


Figure 4: Sensor Placements of the Headset

noise is known as the power line noise. Then features are being extracted and then they are translated into commands by the classifier. The EEG based BCI depends on the thinking pattern of the human it is tested on, but basically the brain pattern tracking depends on the sorting algorithm used that is been used[9].

E. P300 BCI

Based on the type of signals BCI can be divided into 2 types. The first is Exogenous BCI and the second type is the Endogenous BCI. Endogenous BCI system consist those based on sensorimotor rhythms and slow cortical potentials (SCPs) which requires a period of intensive training. Exogenous BCI system uses the brain signals that are P300 stimulated. These signals are of two types, there are "related potentials" and "Steady State Visual Evoked Potential" (SSVEP) which does not require any intensive training [10]. Among all the BCI systems, P300 BCI is very popular and famous. There are various techniques to improve accuracy and flexibility of P300 based BCI [7].

3. Mind Uploading

With Whole Brain Emulation it is possible to map the exact brain functionality [9]. One-to-one mapping of the brain can be achieved via this method. With the development of Mind Uploading, the researchers are guaranteeing that it would enable software intelligence, copyble human capital, give rise to new ethical problems, immortality and a post human species.

Hans Moravec in 1988 was the first to publish a technical explanation about replacing brain cell to brain cell with software with his publication named Mind Children: The future of Robot and Human Intelligence [11].

Ralph Merkle, who is a distinguished computer scientist, was the first person to attempt to analyze the brain emulation. It was conducted in 1989 where the researcher conducted tests to review the automated analysis and tested on re-establishing methods of the brain. With the success of the tests, he further stated that, "a complete analysis of the cellular connectivity of a structure as large as the human brain is only a few decades away".

After the concept was introduced, there came several projects those scanned and reconstructed the human brain in a massive scale. An important examples for such a project is the Blue Brain project [9].

A. Data for Whole Brain Emulation

What we do in Brain Emulation is to get a one-to-one copy of the functional mind, independent of its organic structure to a biological substrate. Several types of data are obtained to model whole Brain Emulation. They can be described as follows [12].

The first data is the mind state. It is the photograph of the minds activity that depicts the internal awareness and the main factor that gives meaning to the way in which each person behaves [13].

The second data is the Mind Transition Functions. These functions showcase the real time development of the mind state. With this second type of data a person's future mind state can be predicted when a mind state of a person at a particular instance is given.

The third and the final type of data is the Transition Update Functions. This type of data explains how the second type of data or the above said Mind Transition Function change due to modifications created by the mental activity of the evaluated person. Examples of change in the biological substrate are modified neurotransmitter vesicle concentrations, modified local calcium concentrations, modified axodendritic model, synaptogenesis, neurogenesis structure etc. [14].

B. The Methodology

The typical human brain consists of approximately 85.5 billion nerve cells or in other words neurons. Each neuron is linked separately to other such neurons by axons and dendrites, which are also parts of the neuron body. The signals between the neurons are transmitted releasing and detecting biological chemicals known as neurotransmitters.

Mind uploading can be achieved by using any of the following two methods. The first method is the copy and transfer method while

the second method is the slow and steady replacement of neurons. Among the above said two methods, the copy and transfer method is the technique that is used in a larger scale. What is being done is that the basic structure of the neuron system is mapped into a simple artificial neural network. Then, the rest of the simulations and visualizations are carried out.

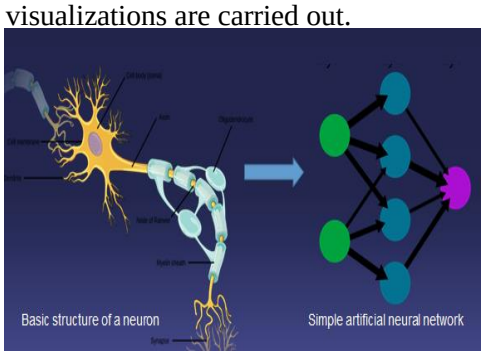


Figure 5: Copy and Transfer

4. The Blue Brain Project

Blue Brain is a project that was initiated to copy and transfer all the contents of the human brain into a virtual brain that resides in a super computer. It is like uploading a mind to a computer.

Through, this Blue Brain project researches focused on preserving a person's memory, knowledge and intelligence for forever even after the death of that person. It also focused on solving the issues people face when remembering things in their day-to-day life such as history, names and routes.

The project was led by Henry Markram, Felix Schurmann and Sean Hill and was initiated in year 2005. In 2006, the scientists were able to successfully simulate and upload a neocortex of a rat. The neocortex is the most developed part of the cortex, which is responsible for sight, conscious thinking and hearing of mammals, which is one of the complex parts of the neuron system.

The objective of the project was carried out by implementing a neo-cortical column. It is the smallest functional unit the neocortex and is the brain component, which is responsible for higher functions such as consciousness. In humans, this column is 2mm in height, 0.5 mm in diameter and is composed of about 60,000 neurons.

The researches built the first single cellular model in 2005. The first artificial neocortical column containing 10,000 neurons was built in 2008. A mesocircuit that consist of 100 such columns was developed in 2011.

A rat brain amounts to have about 100 mesocircuits; therefore, such a rat brain was built in 2014. It is believed that a human brain is equal to 1000 times the above-mentioned rat brain and it can be implemented by 2023. Comparing the neo-cortexes with regard to the number of neurons, human neocortex consists of 60,000 neurons whereas the rat neocortex consists of 10,000 neurons.

The future goal of the researcher is to stimulate a whole human neo-cortex that consists about 1 million cortical columns[15].

A. The Process

The procedure that was followed in the research was data acquisition, simulation, uploading and finally visualizing the results. The organic brain of a human was uploaded into a PC using the invasive method and the noninvasive methods that were previously discussed in this paper[16].

Data Collection is done by collecting brain portions, taking them under a microscope and gauging the shape and electrical behaviour of neurons individually.

The brain model can be extracted in two ways they are the serial sectioning and the brain imaging. In serial sectioning, the brain tissues and the other parts of the nervous system are frozen, scanned and analyzed layer by layer thus capturing the structure of the neurons and their interconnections. While in brain imaging, MRI is used as a technique to map the change in the blood flow around the nervous system and MEG to map the electrical currents. This method captures neurons by their shape, electrical and physiological activity, site within the cerebral cortex, and their population density. This information is then translated into algorithms, which describes the full function of the neurons. Then the generated algorithms are used to generate virtual neurons making them ready for the next step in the process, which is simulation.

Next, the simulation procedure is carried out. It consists of three steps and is controlled by 2 methodologies, which are the simulation speed and the simulation workflow. What happens in simulation is that first, the network skeleton is built from all the different kinds of synthesized neurons. Then, the cells are joined according to the experimentally found rules. Finally, the neurons are functionalized and the simulations are brought to life.

Then the above created neuron blueprints

are analyzed using visualization software. An example for such software is the RT Neuron. It is an ad-hoc software written specifically for neural simulations. This allows programmers and researches to view the activation potentials propagated through or between neurons.

B. The Nanobots and Software

Considering both the invasive and the noninvasive methodologies one could say the best practice to obtain brain signals is by using Nanobots. They are small types of Robots that has the capability to record the brain activities. After the robot is later inserted into a computer it could easily simulate the functionality of the organic brain [1].

A software named "Neuron" is used as the special software for the Blue Brain Researchers in order to perform simulation development. This open source software was programmed using C, C++ and FORTRAN languages.

C. The Blue Gene Supercomputer

As for the hardware necessities, the research need super computers. These computers are called Blue Genes. Many versions of Blue Gene computers were used over the past. Blue Gene/L was used from 2005 to 2009; Blue Gene/P was used until 2011. Blue Gene/Q is the version that is currently used. It is also called JuQEEN. This is known as the world's 8th fastest computer since it has a massive computing speed of 1.6 pitaflops[1].

D. Advantages and Disadvantages of Blue Brain

The blue brain was mainly implemented for the preservation of human mind after death. The next main idea was to promote decision making all by the computer itself. The blue brain has the ability to interact with other humans and animals. This application was tested using a rat brain and had been a success. Another such advantage is to use it to help deaf people by using direct nerve signal acquisition and simulation.

The main disadvantage is humans becoming slaves of the system they themselves created. It raises a chance for the computer-modeled brains to be hacked. This can also lead to a third party controlling of the brain that can be used to create destructions or to cause harm to other human beings or systems. This might eventually lead to cause a war against humans and machines [15].

5. The Applications of Mind Uploading

The application of Mind Uploading can be used for many fields of science and non-science activities. Basically, it can be used to solve brain related problems.

Researches have released that the method can be applied for the treatment of Neuron science related disorders such as Autism, Asperger syndrome, Alzheimer's disease, Parkinson's disease, Bipolar disorder, chronic diseases, dementia, epilepsy, etc. When treating above ailments the problem is to try best practices using real patients, but using mind uploading the practitioners can simulate any of the above ailments in a synthetic platform in order to test mind treatments and activities to solve the problem. Thereby the treatment can be applied to real humans without creating greater harm or difficulties to test subjects [17].

Another interesting usage of WBE is the ability to test the continuous functioning of the brain of a person even after the person has died. This can be mainly used in the case of brain donations where the brain can be analyzed to test and study signs of depression, hallucination, anxiety, stress, brain malfunction, mental retardation and other such cognitive or even genetic disorders.

Among all the applications, the most tested and the most interesting usage of this is the ability to use this technology to help people who are complete or partially paralyzed people will be able use this technology. A similar application for the above is the ability to use this technology to simulate the brain signals of deaf people who are unable to talk and t read their minds.

The discussion about Uploaded Astronauts can also be brought forward as an application of Mind Uploading. In a human spaceflight, space stations or NASA can use uploaded astronauts where the computers with the uploaded minds of the real astronauts are sent to space to eliminate harm caused by zero gravity, vacuum of space and cosmic radiation. This way we can reduce the size of the spacecraft and will be able to increase the Return on Investment in such space projects.

The issues unanswered with current engineering or scientific experimental or theoretical approaches can be sought easily with the usage of this study[18].

Another such application is for the usage of study. Students will be able to study and

practically experience the complexity of the brain, its contents such as the ion channels, receptors, neurons, synapses, role of dendrites, etc. can be well comprehended and explored.

This can also be applied to solve major social problems such as drug abuse, addicted people to intoxicants and people who have done various types of grave crimes. The technology could find as to what mental stimulants resulted the people to do crimes etc.

For the scientists and philosophers who is interested in knowing about the conscious and subconscious mind can also get advantage from this application.

In case of memory loss due to ailments or accidents, with this technology people might be able to recover their lost memories[2].

In simple terms, we will no longer forget anything that is in our mind and we need not fear about it happening. This point can also be explained as the memory backup. Preservation of the memory can be ensured. Since, this data preservation is done in a supercomputer this rise to other advantages as well. It is that computer based intelligence has much faster processing power relative to the human processing power. As a fact, the electromagnetic signals of the brain travel 150 meters per second. While the electronic signals of the brain can travel 2/3 of the speed of the light. This implies that the virtual brain model can think thousands to million times faster than the organic human brain.

With WBI, we can recreate a model of a person's brain, which paves the way to take decisions even when the actual person is present. This is more advanced than any clone since the brain is re-created in a computer and the model can itself act as a high performance super computer that has a high processing speed and a large memory capacity.

Last but not the least among all the above said applications the optimal application is that death, which is the limiting factor of all human existence would no longer effects the knowledge or the core values of a human being. The intelligence of mankind would survive death with the usage of this emerging technology.

When considering the above it is apparent that Mind Uploading can indeed be for the betterment of the society, but just as anything else in the world BCI and mind uploading have its own limitations or challenges.

Virtual mind states cannot die. At least they cannot die due to old age or sicknesses since they are not organic. Virtual beings can make themselves old or young according to their preference and can live as long as it wants until the hardware and software supports the system[13].

6. Challenges of Mind Uploading

This section discusses the challenges/concerns in the mind uploading research. In this sense, we have highlighted ethical, legal and social concerns.

A. Ethical concerns

There is no doubt that the human brain is animpressive creation that is both powerful and unique. The usage of anything in this world can be applied to create both positive and negative impacts to the world. Therefore, the human brain being the most powerful asset, people can misuse it. Therefore, people can cause massive destructions buy using it.

The core values of the human were the only thing that was safe from the adverse impacts of the machine era today. However, one can say that application of BCI is like giving the control of human core values to the hands of technology. Our brains would become dependent on computer systems.

As the next challenge, it is important to consider the impact that is created from technological malware. When you give your brain control over to computer systems, there is high chance for your brain to get effected by different types of malware.

As we all know, man is prone to be adducted to new technologies very quickly as quickly as it grasps technologies. What if people becomes addicted to BCI and become slaves of the technology. This can also be a challenge.

People can also use this ingenious technology for illegal purposes. This will do harm for the other people living in this world and then people would have fear about this subject area. Earlier, when human cloning was introduced, people started to fear the science and brought oppositions to it. BCI and mind uploading is something more strong and powerful than cloning therefore it is clears that people would scare this technology more than there feared cloning [2].

Brain emulations should be done one-nerve materials to be tested. Animals are the option used in such situations. Animal and wild life

enthusiastic brings forwards the theory that it is immoral to use animals for such testing purposes since they do not have the ability to say forth their unwillingness. In the case of using humans as testing materials, none brings out the same concept since humans do have the ability to say their objections, but if the person used for testing is a paralyzed or a so-called locked in person then the morality rule applies.

Next, the question arises about volunteering for providing human brains for testing and laboratory practices. They say that such volunteering would amount to the ending of the biological life of the volunteer. Furthermore such volunteering while alive is likely to be equivalent to assisted suicide since the outcome of such testing is unknown and there is a high probability of risk of failure [19].

Apart from the above the identity of the willing parties to be tested and the global issues such as the extinction of species of animals, which can also arise with such testing should also be taken into account before the application of mind uploading.

B. The Legal Concerns

When this technology is opened to the public, there arises a problem with regard to the legal implications. It is clear that we can program a virtual brain to have cognatic power but does it mean that having cognatic powers like humans can categorize it as human beings with rights.

Are they human replicas or are they separate individuals? That raises a question whether the synthetic brain copies should be called citizens of the country. Do they have immigration rights? Do they take the nationality of the person who was volunteered to provide the brain or do they fall into a different nationality?

Since a virtual personality can be scattered all around the world in different server computers in the globe is it possible to restrict such a person to a particular nationality considering the above stated fact. As it can be clearly seen this discussion regarding the legal implications is itself a book full of questions, people in general need to find answers for it [20].

C. Social Concerns

The virtual personnel's were created as a copy of a true and an existing human being. In other words, the mind was reproduced from a real man's mind. The problems arises from the fact that whether the virtual persons should be given the right or the opportunity to reproduce or re-generate copies of them. Due to the high processing capabilities that the machine embedded copies have, they undoubtedly exceed the powers of the human beings. Therefore, they can easily generate multiple copies of them to reproduce several copies. If that's the case the world will have multiple same personalities [20].

Mind uploading or whole brain emulation creates a one to one copy of the brain in a computer system. When it is done, one can say that the copy acts as same as the organic brain[11]. Therefore, can state that the disorder that can affect the organic brain can also affect the computerized brain. Therefore, the effect should be catered. In order to correct theses errors we need to treat it. This treatment can also be programmed solution[21].

Humans are created with the organic body and the phycology. The culture has a relatively high impact. The thought of death usually motivates as well as scares them to take actions quickly and efficiently [22].As a result of this overwhelming nature they create an anxiety in their minds. This anxiety gives meaningfulness to life as the person tries to work hard during their lifetime. Since WBE abolishes death this anxiety issue becomes erased from the minds of the people[23]. Since death anxiety enriches life it becomes as question as to the fact that WBE is disadvantages to the fruitful lifestyle of human beings[24].

If all the thoughts and mind states can be mapped to computer simulations, that implies that the brain states can be represented by bit streams in computers, which also known as digital data. This Digital data can be changed, copied as well as disrupted due to noise in various instances. Therefore, WBI is open to dangers [25].

7. Discussion

When analyzing the above details we can specifically see that Mind Uploading carries a vast scope of positive outcomes to the society at large. In the same manner, after evaluating the ethical challenges and the implications, a person can also argue that Mind Uploading is indeed a dangerous technology to adhere to. In such situations, it is the general belief to weigh both the sides in a balance to see the impact it can

make. People should realize that all, things in the society has both pros and cons, therefore it should be the responsibility of the humans to use the technology wisely for the development of the society but not to make a negative impact on the world.

When looking back at the researches and developments achieved in the past few decades it is clear that this subject area has much potential to grow and flourish for the betterment of the world. According to the above, among all animals in the world the rat was used as a test subject. It brings forward a hope that his technology therefore can also be used for other animals. As examples, ants, termites and bees are special types of insects that are known to be the most efficient and the most architectural insects of all. Therefore, if researches can find a way to control the brains of these species, the service of these important animals can also be used for the betterment of the world.

If the technology were used to create disaster to humans the species would itself extinct from the universe or else it can be used for the betterment of the society. Therefore, it is evident that the steering wheel of this technology like any other knowledge area is in the hands of its user.

8. Conclusion

Mind uploading is a challenging and a long-term prospect. It has been developed from the mere idea of being a myth into current high-end projects in just about 30 years. Today the study field is developing in a tremendous speed. Along with the daily developments, the study area of Mind Uploading has come across challenges both globally and ethically. The ethical issue of the inability to obtain the consent of the paralyzed people before using as test subjects is now on legal debate in the US. It is believed that the laws would be enforced to govern this situation in the near future. Therefore it is clear that there should be a bridge between both the aspects (Positive and negative) in order to provide a favorable outcome for the society and the world at large. Therefore, it is clear that a scale or measurement should be introduced to acknowledge the feasibility of this study field of Mind Uploading using BCI. Apart from all the technological advancements in the field of Mind Uploading using BCI technology it is apparent that the impact of law and ethics with regard to Mind Uploading is not yet been addressed sufficiently.

References

[1] M. Gidwani, A. Bhagwani, and N. Rohra, “Blue Brain - The Magic of Man,” 2015, pp. 607–611.

[2] K. Amarasinghe, D. Wijayasekara, and M. Manic, “EEG based brain activity monitoring using Artificial Neural Networks,” in *Human System Interactions (HSI), 2014 7th International Conference on*, 2014, pp. 61–66.

[3] S. Q. Xie, C. Gao, Z. L. Yang, and R. Y. Wang, “Computer-brain interface,” in *Neural Interface and Control, 2005. Proceedings. 2005 First International Conference on*, 2005, pp. 32–36.

[4] R.-B. NAGY, F. POPENTIU, and C. Tarca, *Survey of Brain Computer Interface Systems*. Jan, 2014.

[5] D. Prince, M. Edmonds, A. Sutter, M. Cusumano, W. Lu, and V. Asari, “Brain machine interface using Emotiv EPOC to control robai cyton robotic arm,” in *Aerospace and Electronics Conference (NAECON), 2015 National*, 2015, pp. 263–266.

[6] C. B. Matlack, H. J. Chizeck, and C. T. Moritz, “Empirical Movement Models for Brain Computer Interfaces,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 694–703, Jun. 2017.

[7] K. Holewa and A. Nawrocka, “Emotiv EPOC neuroheadset in brain-computer interface,” in *Control Conference (ICCC), 2014 15th International Carpathian*, 2014, pp. 149–152.

[8] D. S. Benitez, S. Toscano, and A. Silva, “On the use of the Emotiv EPOC neuroheadset as a low cost alternative for EEG signal acquisition,” in *Communications and Computing (COLCOM), 2016 IEEE Colombian Conference on*, 2016, pp. 1–6.

[9] D. Wijayasekara and M. Manic, “Human machine interaction via brain activity monitoring,” in *Human System Interaction (HSI), 2013 The 6th International Conference on*, 2013, pp. 103–109.

[10] G. S. Taylor and C. Schmidt, “Empirical Evaluation of the Emotiv EPOC BCI Headset for the Detection of Mental Actions,” *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 56, no. 1, pp. 193–197, Sep. 2012.

[11] A. Sandberg, “Feasibility of whole brain emulation,” in *Philosophy and Theory of Artificial Intelligence*, Springer, 2013, pp. 251–264.

[12] R. Koene and D. Deca, “Editorial: Whole Brain Emulation seeks to Implement a Mind and its General Intelligence through System Identification,” *J. Artif. Gen. Intell.*, vol. 4, no. 3, pp. 1–9, Jan. 2013.

[13] R. A. Koene, “Fundamentals of whole brain emulation: State, transition and update representations,” *Int. J. Mach. Conscious.*, vol. 4, no. 01, pp. 5–21, 2012.

[14] R. A. Wasu and D. Kapgate, “Improving Accuracy in Brain Computer Interface using P300 Potential,” 2015.

[15] N. Ghimire and M. F. Iqbal, “Blue Brain.”

[16] R. Vinayakumar, A. P. D. Varghese, and V. Paul, “The Blue Brain Technology Using Wetware Technology and Fuzzy Logic.”

[17] R. A. Koene, “Experimental research in whole brain emulation: the need for innovative in vivo measurement techniques,” *Int. J. Mach. Conscious.*, vol. 4, no. 01, pp. 35–65, 2012.

[18] A. Sandberg, "Ethics of brain emulations," *J. Exp. Theor. Artif. Intell.*, vol. 26, no. 3, pp. 439–457, 2014.
[19] A. Sandberg, "Monte Carlo model of brain emulation development," Working Paper 2014–1 (version 1.2), Future of Humanity Institute. http://www. aleph. se/papers/Monte% 20Carlo% 20model% 20of% 20brain% 20emulation% 20 development. pdf, 2014.
[20] B. Tonn, "Will Psychological Disorders Afflict Uploaded Personalities?," *World Future Rev.*, vol. 3, no. 4, pp. 25–34, 2011.
[21] *Singularity hypotheses*. New York: Springer, 2012.
[22] C. A. P. Linssen and P. C. Lemmens, "Embodiment in Whole-Brain Emulation and its Implications for Death Anxiety," 2016.
[23] C. Eliasmith and O. Trujillo, "The use and abuse of large-scale brain models," *Curr. Opin. Neurobiol.*, vol. 25, pp. 1–6, Apr. 2014.
[24] R. Cattell and A. Parker, "Challenges for brain emulation: why is building a brain so difficult," *Nat. Intell.*, vol. 1, no. 3, pp. 17–31, 2012.
[25] P. Eckersley and A. Sandberg, "Is Brain Emulation Dangerous?," *J. Artif. Gen. Intell.*, vol. 4, no. 3, Jan. 2013.

Deep Learning Techniques for Speech Recognition and Audio Transcription for Content Captioning

D. G. J. B. Wimalarathne[1], S. C. M. de S Sirisuriya[2]
[1,2]General Sir John Kotelawala Defence University

Abstract

Speech recognition has taken a tremendous interest in the field of Natural Language Processing. Deep Learning has been playing an increasingly large role in speech recognition and one of the most exciting things about this field is that speech recognition is at a place right now where it is becoming good enough to enable exiting applications that end up in the hands of users. Studies have shown that efficiency of devices increases 3 times, with the availably of speech recognition on daily driving devices like mobile phones, cars etc. Content captioning or generating subtitles is very useful and important since the use of videos for communication have been grown phenomenally in the past few years. However, for this, it still need human influence to do a good captioning. But, it is possible to do this with a better efficiency and accuracy with deep learning. This paper provides an overview of how traditional ASR pipeline process speech transcription and how deep learning techniques used to replace modules in traditional ASR pipeline. Finally, describes the implemented speech recognition neural network model built on Tensor flow platform. This model is capable of recognizing English spoken digits from 0 to 9 and transcript it to text. Currently model has achieved around 70.19% accuracy over different training and testing data. This prototype model is reusable for any amount of words in any language with the availability of more data and computing power.

Keywords: *Speech recognition, Feature extraction, Acoustic model, Language model, Connectionist Temporal Classification, MFCC, Tensor flow, Librosa*

1. Introduction

Speech-to-text has achieved a motivating development in the modern world. Numerous forms of end devices like Personal Computers, hand held smart devices and even trendy cars are integrated with this feature for the convenience of the end user.

There are different components that make up a complete speech applications like speech transcription, word spotting/trigger word and speaker identification/verification. Speech transcription is essential to come up with words that represent what is being said. Applications, which need to do an action or get triggered for voice, need continuous listening. In addition, some applications may use speech recognition for authentication purposes too. For a content captioning application, it need to have a good speech transcription model.

This paper presents the basic flow of how traditional Automatic Speech Recognition systems like Hidden Markove Model process audio transcriptions, then how deep learning techniques have been influenced to traditional ASR pipeline. How the audio is preprocessing and generating simple spectrogram and how CTC (Connectionist Temporal Classification) approach do speech recognition with generated spectrogram. And finally presents the speech recognition neural network that have been developed by using Google machine learning platform Tensor flow, for the system that generates subtitles for a given audio/video.

2. Traditional ASR Pipeline

Most of the traditional ASR systems are based on probabilistic theories. [1]Traditional system break the problem of taking audio and breaking it into different parts. The raw audio has represented with some kind of feature representation.

Job of the acoustic model is to learn relationship of features representation of audio and the sequence of words W, words of the language.[2] A language model contains the structural constraints available in the language to generate the probabilities of occurrence. It induces the probability of a word occurrence after a word sequence.[3]

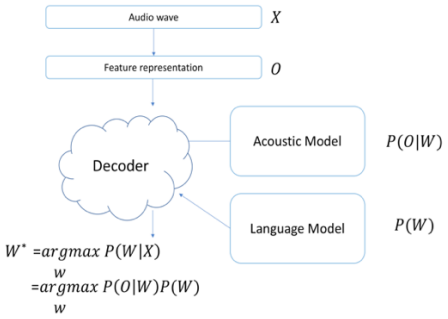


Figure 1: Modules of ASR

The job of the decoder is to find the sequence of word W that maximize the particular sequence of words W given the audio.

$$W^{*} = \underset{w}{argmax}\, P(W|X) \quad (1)$$

That is equivalent to maximizing the product of contribution from acoustic model and language model.

$$= \underset{w}{argmax}\, P(O|W)P(W) \quad (2)$$

However, it is not directly possible to transcribe in to characters especially in English language because way something is spelled is not always correspond to its sound. There for a way to get around this issue is to that replace this with the sort of intermediate called phonemes. So instead of trying to predict characters, it predicts the phonemes by the acoustic model.[4]

Example w_1 = "hello" = [HH AH OW] = [$q_1q_2q_3q_4$]

In the above example, if the word the word that need to be predicted should be hello, instead of characters, acoustic model predicts the corresponding phonemes. This in one sense makes modeling problem easier. Phonemes are the perpetually distinct units of sound that distinguish words. They are very approximate. It is not clear how fundamental this is. Therefore, this phoneme representation adds more complexity to this traditional pipeline. Because, acoustic model does not associate with words, instead it associates with another kind of transcription into phonemes. Therefor pipeline needs kind of dictionary or a lexicon to understand how it convert phonemes into words. This pipeline is highly tweakable[5], but also hard to get work well. Each part of it has its own challenges like choosing representation.

3. Deep Learning Approach

The first place that deep learning made an impact on speech recognition is to take one of the core machine learning component of the system and replace it with a deep learning algorithm. One of the fundamental problems in speech recognition is to build a neural network that map audio signal to a transcription that have a variable length. Unlike an image, which has 2D grid of pixels, audio is just 1D signal. One second of audio clip it can broke down in to small parts and each element would be a floating-point number.[6]

A. *Preprocessing*

There are two ways to start the preprocessing.

1) *Minimally pre-processing (by simple spectogram)*
2) *Training model raw audio wave*

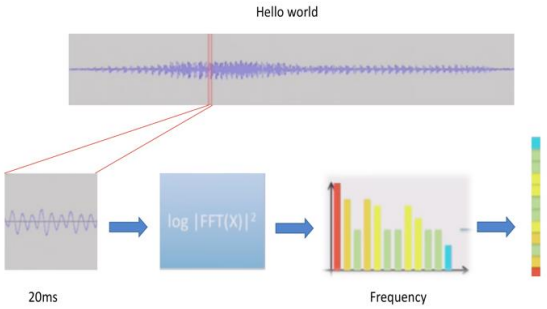


Figure 2: Small window of audio in terns on frequency

Minimally processing is some vanilla preprocessing like convert to a simple spectrogram. Mel-Frequency cepstral coefficients is a popular kind of features, and engineer complex representations. Spectrogram is sort of like a frequency domain representation. Figure 2 represent a small window in term of frequencies. [7]

The small window is about 20ms long. It can see that when it gets down to this scale, that audio waves are made up of combinations of different frequencies of sine waves. Then it has to compute Fourier transformation of the sine values and then take the log of the power of each frequency. Result of this is, that for every frequency of sin waves, what is the amount of magnitude represent by that sine wave that makes up the original signal. Therefor transformation need to apply to all the windows

of frames in audio signal. This will result a creation of spectrogram.

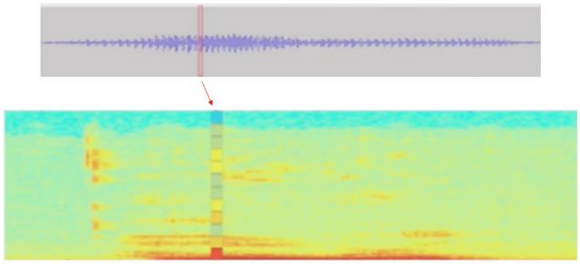


Figure 3: Spectrogram for all the window of frames

4. CTC Based End-To-End Speech Recognition

Recent years with the availability of high computing power and more data, the speech transcription has been developed and equipped with many powerful tools and libraries.[8] CTC is one of the successful approach because it explicitly uses the history of the target character without condition.

CTC is a probabilistic mode p(Y|X) where X is sequence of frames of data. Y is output tokens of data of equal length of L. X is the spectrogram.[9]

$$X = x_1, x_2, x_{31} \ldots . x_T \quad (3)$$

$$Y = y_1, y_2, y_{13} \ldots . y_L \quad (4)$$

$$T \geq L$$

The spectrogram frequencies feed into recurrent neural network as the input. The neural network is a bidirectional RNN. There, for any time step is depend on entire data, so it can compute fairly complex function over entire data. Output is bank of Softmax neurons.[10] They represent the distribution over different symbols (Softmax vocabulary), that could be hear in the audio. For example, for the first frame of audio, it will be processed by RNN and for output, it has Softmax neurons that correspond to each of the neuron.

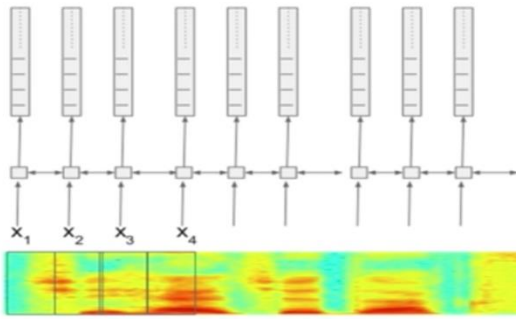


Figure 4: Spectrogram feeding sequence of frames of data

In addition, the set of Sofmax neurons represent the probability of containing each symbol in the spectrogram frame and their output is summing to 1.

For an English word transcription, the Softmax vocabulary would be {a, b, c, d, ...z,} and extra blank token. Black is a special character for identification of intervals and rest times of the speech. Softmax at step t, gives a score s(k,t) =logPr(k,t|X) at the category k in the output at time t.

Therefore, the target is to find a transcript through this individual Softmaxes over the time. It is essential that each symbol must go through a blank symbol. You can transfer either from same character to itself or from that symbol to a blank. Therefor it can end up different sequences, representing the same symbol.

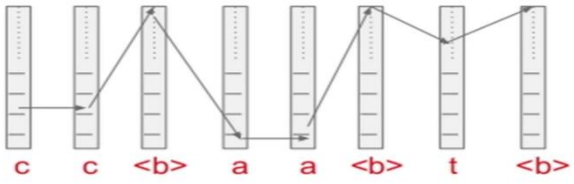


Figure 5: One possible sequence of transcription for word cat

In figure 5, it represents one possible sequence that can be obtain for the transcription for word cat.

Following are some possible sequences as well.

cc <b>aa<b>t<b>
cc<b><b>a<b>t<b>t<b>
ccccc<b>aa<b>tttt<b>

Output neurons define distribution over whole character sequences c, assuming independence:

$$P(c|x) \equiv \prod_{i=1}^{N} P(c_i|X) \quad (5)$$

This process makes a transcription of same length to the audio. Therefore, the second step is to define a mapping that crunches down this sequence into actual transcription. Given a specific sequence c, squeeze out the duplicates plus blanks that yield to transcription.

$$\beta(C) \rightarrow y$$

$$Y = \beta(C) = \beta(\text{cc_aa__t_}) = \text{cat} \quad (6)$$

The probability of transcription is the sum of the probabilities of all the possible character sequences that is correspond to the audio.

$$P(Y|X) = \sum_{c:\beta(c)=y} P(c|x) \quad (7)$$

Because of the dynamic programing, it is possible to compute both the log probability p(Y|X) and its gradient exactly. This gradient can be propagated to neural network by backpropagation and update network parameter θ to maximize likelihood of correct label y*.[11]

$$\theta^* = argmax \sum_i logP(y^{*(i)}|x^{(i)}) \quad (8)$$

Today the above process is at the level of technology and implemented that efficiently calculate CTC functions cost function that can calculate the likelihood and also can give back the gradient.

Some of the common issue of CTC based speech recognition are "Tchikovky" problem, relationships in between words, and identifying proper nouns. "Tchikovky" problem is there are certain word which you cannot spell unless you have seen the exact spelling for that problem (knife, Tsunami), since P(Y|X) molded directly from audio. Therefor with the CTC implantation it, fuse a language model by taking small step back from end to end transcription system.

5. Speech Recognition Model

The following system has been implemented and currently at the training and testing stage which is a sub part of the developing system that generates content captioning (subtitle generation) for a video/audio input, by using deep learning techniques.

The key was to give more data and computing power. These types of neural networks are used in services like Siri, Echo and Google Now. The following speech recognition deep neural network is constructed with the Google machine learning framework Tensor flow that learns to recognize spoken numbers. A labeled data set that people saying numbers are used to train the network and currently testing it from validation data in order to improve the accuracy.

A. *Fetch Data*

The model is trained with set of audio files. Each wave file is a recording of different spoken digit. These files contain different people (male/female) speaking numbers from 0 to 9 and they have variable lengths. Each file is labeled with a written digit. A function is written such that it returns list of labeled speech files as a batch. So, this batch is a set of audio files and their corresponding label of written digit. This batch is use for training the neural network from supervised learning technique.

All these utterances are examples of red speech. Audio that are clear with no noise. Choosing a data set is highly dependable with the application that is going to be develop. Reason that red speech is popular is because of its availability. Moreover, even it does not perfectly match to the application, it is free and getting a lot is helping the accuracy. But, it will lose factors such as inflection/conversational tone, Lombard effect and speaker variety. However, it still can be helpful.

B. *Feature Extraction*

Next step is feature extraction process from the audio files. The extracted features and number of features to be extracted depend on application that is going to develop. For subtitle generation, feature extraction, system uses Mel-frequency Cepstrum Coefficients (MFCC)[12], the power spectrum that implements Fourier transformation analysis since input data are red speech data (clear with no noise).

MFCC features are set of specific features defined for set of applications constrains. MFCC values are not very robust in the presence of additive noises it is common to normalize their values in speech recognition system to reduce the influence of noise. Many audio analysis library packages include MFCC[13] feature extraction and the for the system LibROSA[1] python package for audio music analysis have been used.

C. *Neural Network*

Network consist of 3 layers input layer, hidden Long Short Term Memory (LSTM) recurrent neural network layer and output Softmax layer.

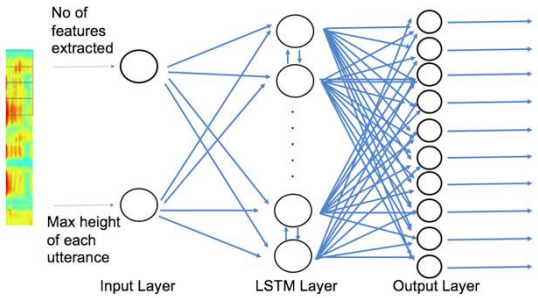


Figure 6: Neural network model diagram

Initial input layer takes output of MFCC features as the input. Here, the input parameters are number of features extracted and the maximum height of each utterance. This input layer initializes by calling tflearn's input data function. This initial input layer will be the gateway to that data is fed into the network and parameter it will help to define shape of input data (input tensor)[14]. Tensor is multi-dimensional array of data.

Next layer is the hidden layer which is Long Short Term Memory Recurrent[15] neural cells layer. A recurrent net was used since its processing sequences. In a recurrent net, the output data's content is influenced not only by the input just put in, but also by the entire history of inputs thorough recurring loop. LSTMs can remember everything it's been fed.[16] Too few neurons will lead to bad predictions and too many will over fit to training data. Currently 128 LSTM neurons have been used for the model.

Final layer is fully connected Softmax layer. There are 10 classes since model is predicting numbers from 0 to 9. Each Softmax cell provide a probability of occurrence of that label, for the given input. Output layer is a regression, which will output single predicted number for the utterance. Adam optimizer have been used to minimize categorical entropy loss function.

D. Training the Model

Model has been trained on Tensor flow platform. Number of epochs and neurons for LSTM layer have been modified randomly at training epochs

```
'1_Tom_60.wav', '1_Daniel_60.wav', '1_Samantha_60.wav',
'1_Tom_40.wav', '5_jackson_32.wav', '1_Daniel_80.wav',
'1_Daniel_40.wav'
```

Figure 7: Sample of testing data

Figure 7 shows sample of set of training data. The label of the file. Label gives the 3 information, number spoken in the file, person who spoked the number and the maximum length of the utterance. At the stage of fetching data, these labels and audio are separated and features are extracted from the audio, and text digit of the spoken number is extracted from the label. The above sample shows the testing data which people are speaking 1,1,1,1,5,1,1 and 1. This batch the audio wav files and the corresponding labels used to train the neural network.

At the output, each Softmax neuron represent a digit from 0 to 9. Finally, Softmax

```
[ 0.09084857  0.11164435  0.1248319   0.10164179  0.0968072   0.10064425
  0.08926999  0.09505827  0.09672815  0.09252549]
[ 0.0905322   0.113589    0.10274187  0.0998391   0.10131323  0.10431337
  0.09550676  0.10178526  0.09272844  0.09765081]
[ 0.08989014  0.12128935  0.10938397  0.10029924  0.10303683  0.09746677
  0.09099476  0.09923676  0.0950058   0.09339646]
[ 0.09087516  0.11178062  0.12466295  0.10169461  0.09656566  0.10082287
  0.08918555  0.09496398  0.09669589  0.09275273]
[ 0.09298929  0.12399727  0.10001957  0.09866378  0.10374859  0.1008506
  0.09278064  0.0972227   0.09398942  0.09573802]
[ 0.09035728  0.11348522  0.10282305  0.09983667  0.10124455  0.10435842
  0.09551056  0.10183558  0.09278652  0.09776221]
[ 0.09054273  0.1134324   0.10293662  0.09982502  0.10126955  0.10428398
  0.09558079  0.10175306  0.09268826  0.09768756]]
```

Figure 8: probabilities produced by the Softmax layer for each wav file in the sample

function of the output neurons (Softmax neurons) will produce a probability for the given input for being that digit.

Figure 8 shows probabilities produced by the neural network, for the sample data in figure 7. One array contains probability of that spoked number being 0 to 9. So together, there are 7 arrays for each 7 audio files. Output of the network is an array of array of probabilities for multiple inputs.

Next probabilities are sorted descending from highest accuracy to lowest accuracy and the corresponding digit for that probability has been returned. Therefor the final output is an array for each audio input such that it contains predicted digit with highest confidentially to lowest confidentially.

```
[2 1 3 5 4 8 7 9 0 6]
[1 5 2 7 4 3 9 6 8 0]
[1 2 4 3 7 5 8 9 6 0]
[2 1 3 5 8 4 7 9 0 6]
[1 4 5 2 3 7 9 8 0 6]
[1 5 2 7 4 3 9 6 8 0]
[1 5 2 7 4 3 9 6 8 0]
```

Figure 9: predicted digits from its highest confidentiality to lowest

In above example, out of 7 inputs, the model has predicted 3 correctly. This result may differ for same inputs in another instance.

E. Testing and Evaluation

The neural network model, currently trained with different training iterations, by changing number of neurons in hidden layer and also changing the drop out value. Model was trained over 2400 audio data with both male and female voices on 2.7 GHz intel processor.

Currently it shows 70.19% accuracy on the Tensor flow platform. This is a simple prototype model that can be configurable according to language and also the type of application. Currently, this neural network model has integrated to offline subtitle generation application as the speech recognition engine.

Application takes video/audio file of mp4 or mp3 format. It recognizes digits from 0 to 9 spoken in the content and transcript it to text. Then application will create a subtitle file (.srt format) for generated text. Subtitle file is timely synced along with the audio so it can play the video file with the subtitle file.

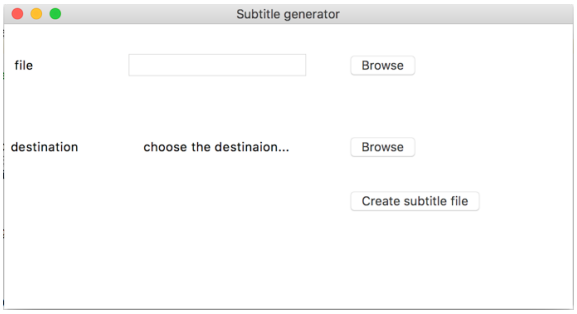


Figure 10: GUI interface of the subtitle generating system

This prototype neural network is reusable for any language with any amount of words with the availability of data and the computing power. Integrating neural network model will decrease the space and runtime complexity of the system than traditional ASR pipeline since there is no any word dictionary or lexicon to map phonemes and hence no mapping. However, since there is a lack of data to train some words like proper nouns, it is good to have some language model process too.

6. Conclusion

With the proliferation of Speech-to-text technology, it is clear that it has become an essential technology in almost every electronic device. Content captioning is an application area which we can use ASR efficiently but it still needs human influence to do a better captioning. Therefore, it is essential to have a good speech recognition system for these type of applications

Traditional Automatic Speech Recognition pipeline comprises of different modules Feature representation, Acoustic model, and Language model. Phonemes are the intermediate in between extracted features and acoustic model because way something is spelled is not always correspond to its sound. Therefor a dictionary or a lexicon is need to map words and phonemes. This phonetics to word mapping make the traditional ASR pipeline more complex.

Deep learning techniques can be used to replace those modules in traditional ASR pipeline. Conventionalist Temporal Classification (CTC) is such an implementation and it provide a solution for the problem of mapping audio signal to a transcription that have a variable length without having any intermediate of phonetics.

Today CTC based implantations are at a level of technology and lot of libraries are available for such implementations. Speech recognition neural networks can be implement to generate content captioning (subtitle generation) and these models can develop by having more data and computing power and can be adopt into many words in many languages.

A simple neural network model has been developed for content captioning application. This neural network model has used to replace the Acoustic model in the traditional ASR pipeline. Currently, this model can recognize English spoken digits from 0 to 9 and transcript it into text. Since there is no any phonetic interaction, this model makes less runtime and space complexity. In addition, the model is reusable for any language with the availability of the data and computing power.

This speech recognition neural network has been integrated to the subtitle generation application. It is an offline application that takes video or audio file as the input and provide .srt formatted subtitle file with timely synced digits. Since the reusability of the neural network model, the application can develop for many language transcriptions or different language transcriptions.

References

[1] B. McFee *et al.*, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, pp. 18–25.

[2] M. A. Anusuya and S. K. Katti, "Speech recognition by machine, a review," *ArXiv Prepr. ArXiv10012267*, 2010.

[3] L. Deng and X. Li, "Machine Learning Paradigms for Speech Recognition: An Overview," *IEEE Trans. Audio Speech Lang. Process.*, vol. 21, no. 5, pp. 1060–1089, May 2013.

[4] A. Mathur, T. Saxena, and R. Krishnamurthi, "Generating Subtitles Automatically Using Audio Extraction and Speech Recognition," 2015, pp. 621–626.

[5] S. K. Gaikwad, B. W. Gawali, and P. Yannawar, "A Review on Speech Recognition Technique," *Int. J. Comput. Appl.*, vol. 10, no. 3, pp. 16–24, Nov. 2010.

[6] D. Gerhard, University of Regina, and Department of Computer Science, *Audio signal classification: history and current techniques*. Regina: Dept. of Computer Science, University of Regina, 2003.

[7] N. Q. Duong and H.-T. Duong, "A Review of Audio Features and Statistical Models Exploited for Voice Pattern Design," *ArXiv Prepr. ArXiv150206811*, 2015.

[8] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[9] K. S and C. E, "A Review on Automatic Speech Recognition Architecture and Approaches," *Int. J. Signal Process. Image Process. Pattern Recognit.*, vol. 9, no. 4, pp. 393–404, Apr. 2016.

[10] K. Duan, S. S. Keerthi, W. Chu, S. K. Shevade, and A. N. Poo, "Multi-category classification by soft-max combination of binary classifiers," *Mult. Classif. Syst.*, vol. 2709, pp. 125–134, 2003.

[11] M. K. Brown, M. A. McGee, L. R. Rabiner, and J. G. Wilpon, "Training set design for connected speech recognition," *IEEE Trans. Signal Process.*, vol. 39, no. 6, pp. 1268–1281, 1991.

[12] K. R. Ghule and R. R. Deshmukh, "Feature Extraction Techniques for Speech Recognition: A Review," *Int. J. Sci. Eng. Res.*, vol. 6, no. 5, pp. 2229–5518, 2015.

[13] J. Martinez, H. Perez, E. Escamilla, and M. M. Suzuki, "Speaker recognition using Mel frequency Cepstral Coefficients (MFCC) and Vector quantization (VQ) techniques," in *Electrical Communications and Computers (CONIELECOMP), 2012 22nd International Conference on*, 2012, pp. 248–251.

[14] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *ArXiv Prepr. ArXiv160304467*, 2016.

[15] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[16] H. Palangi *et al.*, "Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval," *IEEEACM Trans. Audio Speech Lang. Process. TASLP*, vol. 24, no. 4, pp. 694–707, 2016.

Dictionary Based Machine Translation System for Pali to Sinhala

R. M. M. Shalini[1], B. Hettige[2]
[1]Faculty of Humanities and Social Sciences, University of Sri Jayewardenepura,
Nugegoda, Sri Lanka
[2]Department of Computer Science, Faculty of Computing,
General Sir John Kotelawala Defence University, Rathmalana, Sri Lanka
rmmshalini@gmail.com[1], budditha@kdu.ac.lk[2]

Abstract

Machine translation systems are language translation tools that are also capable to be used as language learning tools. This paper presents a machine translation system that has been developed as a language learning tool for the Pali to Sinhala. This Pali to Sinhala translation tool can translate simple Pali sentences into Sinhala through the dictionary-based approach. The translation system comprises of three modules, namely the Pali morphological analyzer, the dictionary-based translator, and the Sinhala morphological generator. The Pali morphological analyzer uses an affix-spiriting approach to identify a relevant Pali root word for the existing Pali word. The Pali to Sinhala translator identifies an available Sinhala-based word for the existing Pali-based word with the support of the Pali-Sinhala dictionary. The Sinhala morphological generator generates appropriate Sinhala words by using the Sinhala root word and the relevant morphological information of the Pali word. This translator uses word-level translation and gives attention only to source and target morphology. The Pali Sinhala translator has been successfully tested for simple Pali sentences as a language learning tool for the grade 6-9 Dhamma schools.

Key Words: *Language Learning Tool, Machine Translation System, Morphological Analyzer, Translator, Morphological Generator.*

1. Introduction

Language Translation is a communication of the meaning of a source-language text by means of an equivalent target-language text through the human concern [1]. This language translation process can be automated through the computers and still is a challenging research task of the Natural Language Processing. Automated Language translation process is normally called as machine translation, and that can be done through the machine translation systems [2]. There are thousands of machine translation systems are available with different translation approaches including google translator [3], Anusaaraka [4], Systran [5] etc. Theses translation systems gives different performance based on the language pair and the approach used to translate. These Machine Translation approaches are normally categorized with considering the level of attention made on morphology, syntax, and semantics of the source and target [6]. According to the translation pyramid, the direct translation is the lowest level machine translation approach. It is not much level of considering syntax and semantics on both source and target languages. Therefore, direct translation approach is more suitable for language pairs with a closed relationship in between syntax semantics and morphology. Figure 1 shows the translation pyramid of the machine translation.

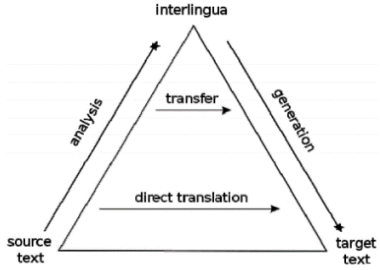


Figure 1: Translation Pyramid

For instance, Sinhala and Pali are the two related languages and has closed relation on syntax and morphology.

Pali is one of the Indo-Aryan languages that has some syntactical, semantical, and morphological relations with Sinhala. Further, Pali is widely studied by the Sri Lankan dhamma school students; thus, it is the language of Buddhism. With this viewpoint, a machine translation system has been developed as a language learning tool for the Pali to Sinhala. This Pali to Sinhala translation tool can translate simple Pali sentences into Sinhala through the dictionary-based approach. This tool would be

beneficial for the Buddhist priests in Pirivenas, Students and who wish to learn Pali language.

The rest of the paper is organized as follows. Section 2 reports brief introduction on Machine Translation systems, morphological analysis and morphological generation with some note on existing systems. Then section 3 reports computational grammar on Pali and Sinhala language, which is required to develop dictionary, based machine translation system. The section 4 presents design and implementation of the Pali-Sinhala machine translation system and section 5 demonstrate how systems work on Simple Pali sentence. Finally, section 6 present conclusions with a note on further works.

2. Related Work

This section briefly describes related works for the machine translation, including Morphological analysis and generation.

A. Machine Translation

The Machine Translation is a sub-area of the Natural language processing which is identified during early days of Artificial Intelligence (AI) [2]. The process of the Machine Translation can be described simply as decoding the meaning of the source text and re-encoding this meaning in the target language. However, due to the complexity of the natural languages, development of machine translation system has become a research challenge.

Further, according to the translation methods, Machine translation systems can be broadly categorized into two groups, namely, the direct translation system and the indirect translation system. The direct translation system translates source language to target language by using word-to-word or phrase-to-phrase mapping. Indirect translation systems use an Interlingua or transfer approach. In addition to the above, in general, machine translation approaches, machine translation systems can be classified into seven categories, namely Human-assisted, Rule-based, Statistical, Example-based, Knowledge-based, Hybrid and Agent-based [6].

Dictionary-based translation is taken as a basic approach to machine translation. This type of machine translation applications is easy to implement and that attempt quite competent tools for retrieving translations for unknown search keys. Commonly, dictionary-based translation is basically a translation with a help of a bilingual dictionary. In addition to the bilingual dictionary most of the systems consists of Morphological analyzer and a generator for the source and target languages. For instance, Sindhu and others have developed a dictionary-based machine translation from Kannada to Telugu [7]. The translation system comprises with five components namely Morph analyzer, dictionary, transliteration, transfer grammar and the morph generator.

Consider the Pali language only very limited research has conducted to translate Pali text into other languages. Phoson and others have developed a prototype Rule-based Machine Translation system for Pali to Thai [8]. This system capable to analysis Pali language structure of the input sentence and re-generate Thai language structure. The next section briefly describes Morphological analysis and generation for Machine translation.

B. Morphological Analysis

Morphological analysis is one of the key components in the rule-based machine translation systems. That helps to identify the based word and the grammar for the existing source language words. In addition to the above morphological analyzer is must essential when language has a strong morphology. In the view of this, the morphological analyzer is one of the essential tools for the Asian language processing systems. Thus, some Asian countries including India, Japan, and Thailand have also developed morphological analyzers for their language processing [9]. For Instance, Anusaaraka has been designed to translate among major Indian languages and its morphological analysis is based on the word paradigms [4]. A Sinhala morphological analyzer has been developed for purpose of machine translation [10]. The system uses affix stripping approach to analyze the Sinhala words [11].

C. Morphological Generation

Morphological generation is the backward process of the Morphological analysis. A morphological generator is also an essential tool used in Natural Language Processing (NLP) applications for the generation of final fully inflected words from the root and a set of morphological properties. Note that, highly inflected languages like the Dravidian languages of Tamil, Malayalam, Telugu, and Sinhala requires a morphological generator for produce grammatically correct results. Further,

morphological generator plays an indispensable role in target language sentence generation in Machine Translation (MT) systems. It is also used in Information Retrieval (IR) systems in the front-end for query expansion [12]. Further, the aim of morphological generation is to produce the inflected form of a word according to the features and values in the Feature Structure. It is also necessary to reuse the linguistic resources created for analysis purpose. From a practical point of view, morphological generation is the inverse process of analysis, namely the process of converting the internal representation of a word to its surface form. The same rule definitions can be used to generate the desired word form as used for analysis. Affix stripping approach is an approach to morphological analysis that checks the root form from the available word by adding or removing letters. This type of approach can be easily applying for grammatically rich languages like Sinhala and Pali. With the above view, a research has been conducted to develop Pali to Sinhala machine translation system through the direct transfer approach.

3. Computational Grammar for Pali and Sinhala

This section briefly introduces Pali and Sinhala Morphology, which are used to analyses and generate the morphology of both languages.

A. The Pali language

Pali is an Indo- European language derived from Sanskrit that for centuries has been the canonical and liturgical language of Theravada Buddhism. Pali continues to be used throughout Southeast Asia as a religious language - in prayer and ritual, as well as in the study scripture and the composition of commentaries and other religious documents. Even into the 20th Century. Some Theravada elders continued to be able to converse in the ancient tongue. The term ‘Pali' originally had the meaning of ‘canonical'; over the years of use preserving the Pali Canon [13].

The Pali alphabet consists of 41 letters, eight vowels and thirty- three consonants [14]. Long and short vowels are only contrastive syllables: in closed syllables, all vowels are always short. Short and long ඉ and ඔ are in complementary distribution: the short variants occur only in closed syllables; the long variants occur only in open syllables. Short and long ඉ and ඔ are therefore not distinct phonemes

Pali is a highly inflected language, in which almost every word contains, besides the root conveying the basic meaning, one or more affixes (usually suffixes) which modify in some way. Nouns are inflected for gender, number, and case; verbal inflections convey information about a person, number, tense, and mood.

Pali nouns inflect for three grammatical genders (masculine, feminine and neuter) and two numbers (Singular and plural). The nouns also, in principle, display nine cases.

However, in many instances, two or more of these cases are identical in form; this is especially true of the genitive and dative cases.

Table 1: inflection for the Pali Nouns

	පුරුෂ ලිංග (බුද්ධ ශබ්දය)	
	ඒක වචන	බහු වචන
ප්‍රථමා	බුදෙධා	බුද්ධා
දුතියා	බුද්ධං	බුදේධ
තතියා	බුද්ධේන	බුද්ධේහි
චතුර්ථී	බුද්ධාය	බුද්ධානං
පඤ්චමී	බුද්ධා	බුද්ධේහි
ෂෂ්ඨී	බුද්ධස්ස	බුද්ධානං
සප්තමී	බුදෙධ	බුදේධසු

	ස්ත්‍රී ලිංග (කඤ්ඤා ශබ්දය)	
	ඒක වචන	බහු වචන
ප්‍රථමා	කඤ්ඤා	කඤ්ඤා
දුතියා	කඤ්ඤං	කඤ්ඤා
තතියා	කඤ්ඤාය	කඤ්ඤාහි
චතුර්ථී	කඤ්ඤාය	කඤ්ඤානං
පඤ්චමී	කඤ්ඤාය	කඤ්ඤාහි
ෂෂ්ඨී	කඤ්ඤාය	කඤ්ඤානං
සප්තමී	කඤ්ඤායං	කඤ්ඤාසු

	නපුංසක ලිංග (ඵලං ශබ්දය)	
	ඒක වචන	බහු වචන
ප්‍රථමා	ඵලං	ඵලානි
දුතියා	ඵලං	ඵලං
තතියා	ඵලේන	ඵලේහි
චතුර්ථී	ඵලාය	ඵලානං
පඤ්චමී	ඵලේන	ඵලේහි

There are Three Tenses, two voices, two numbers, and three Persons in the conjugation of Pali verbs.

The table 2 shows some inflection form for the Pali word.

Table 2: Pali verb inflections

Person	Number	Present tense	Past tense	Future tense
First	Singular	ගච්ඡාමි	ගවජි	ගච්ජිස්සාමි
First	Plural	ගච්ඡාම	ගවජිංසු	ගච්ජිස්සාම
Second	Singular	ගච්ඡසි	ගවෙජා	ගච්ජිස්සසි
Second	Plural	ගච්ඡථ	ගච්ජිත්ථ	ගච්ජිස්සථ
Third	Singular	ගච්ඡති	ගවජිං	ගච්ජිස්සති
Third	Plural	ගච්ඡන්ති	ගවජිංම්හ	ගච්ජිස්සන්ති

Further, the Sinhala language is closely related to the Sinhala language. The next section briefly reports some required computational grammar for Sinhala.

B. The Sinhala language

Sinhala is the main language of Sri Lanka. Approximately 21 million speak Sinhala as their mother tongue and many others in the country such as Tamils and Muslims speak it as a second language. There are also considerable numbers of Sinhala speakers in Singapore, Thailand, Canada and the United Arab Emirates. The Sinhala language belongs to the Indo-Aryan subdivision of the Indo-European language family. It was developed with the help of Pali and Sanskrit that can be mentioned as sacred languages of the Sri Lankan Buddhists. Because of the European colonization, Portuguese, Dutch, and English have also influenced the Sinhala language.

The Sinhala language has its own alphabet and writing system, which was developed for far too long. Sinhala is the only language in the world that uses the Sinhalese alphabet. There are 61 characters in Sinhala alphabet including 18 vowels, 41 consonants, and 2 semi-consonants.

There is a rich morphology system in the Sinhala language. These systems were inherited from Brahmi script to the Sinhala language [16]. Both nouns and verbs are generated into many forms with morphology in Sinhala language usage. Sinhala verb is divided into general classes, namely, a transitive verb (sakarmaka) and intransitive (Akarmaka). Further, these two verb categories are inflected for voice (karaka), mood (vidi), tense (kala), number (wachana) and person (purusha). Voice can be either active or passive. There are four types of moods, namely, indicative, optative, imperative and conditional. The Sinhala language has only three tenses [16]. They are Past tense, Present tense, and future tense. Main verb (Akkyathaya) participate three types of inflections namely person, number, and sex. Table 3 shows inflection forms of a verb in the active voice and the passive voice.

Table 3: inflection forms of a verb in the active voice and the passive voice

Person	**Number**	**Present**	**Past**	**Future**
First	Singular	බලමි	බැලීමි	බලන්නෙමි
First	Plural	බලමු	බැලීමු	බලන්නෙමු
Second	Singular	බලහි	බැලීහි	බලන්නෙහි
Second	Plural	බලහු	බැලීහු	බලන්නෙහු
Third	Singular	බලයි	බැලී	බලන්නේය
Third	Plural	බලති	බැලූ	බලන්නෝය

Inflection form of the Sinhala verbs (active)

Person	Number	Present	Past	Future
First	Singular	බැලෙමි	බැලිණිමි	බැලෙන්නෙමි
First	Plural	බැලෙමු	බැලිණිමු	බැලෙන්නෙමු
Second	Singular	බැලෙහි	බැලිණිහි	බැලෙන්නෙහි
Second	Plural	බැලෙහු	බැලිණිහු	බැලෙන්නෙහු
Third	Singular	බැලෙයි	බැලිණි	බැලෙන්නේය
Third	Plural	බැලෙති	බැලුණු	බැලෙන්නෝය

Inflection form of the Sinhala verbs (passive)

4. Design and Implementation

This section briefly describes design and implementation of the Pali to Sinhala translator. Figure 2 shows the design of the Pali Sinhala translator. The translator consists of three modules namely Pali Morphological Analyzer, Pali to Sinhala translator and Sinhala Morphological generator. The task of each module is reported in the below.

A. Pali Morphological Generator

Pali Morphological generator is the key module in the translator that reads Pali word as an input and provide root word and the grammatical information for each word. This Pali Morphological generator uses affix spiriting approach to handle the Pali morphology. Under this research, we have identified irregular and regular forms of the Pali noun and words. Irregular words are put into Pali dictionary and root word indentation table has been developed for the regular words. In the Pali language normally noun participate gender-based conjugation. In addition to that, conjugation can be divided into the last sound of the noun call "Anthaya" The following Table shows the last sound ‘ඇ' for the Pali word "බුද්ධ"

Table 4: Apexes table for the Singular plural noun with last sound ‘ඳ.

Case	Example	Add	Remove
පඨමා	බුද්ධො	ධො	ධ
දුතයා	බුද්ධං	ං	
තතයා	බුද්ධෙන	ධෙන	ධ
චතුථ	බුද්ධාය	ාය	
කරණ	බුද්ධෙන	ධෙන	ධ
පඤ්චම	බුද්ධා	ා	
ඡටය	බුද්ධස්ස	ස්ස	
සතතම	බුද්ධෙ	ධෙ	ධ
ආලපන	බුද්ධ		

With the same method, we have identified affix spiriting table for the Pali Noun and Verbs. By using these grammatical rules, the root word of the Pali word has been identified. This analysis has been done with the support of the Pali lexical database that consists of regular nouns, regular verbs, irregular nouns, verbs and other words.

B. *Pali to Sinhala Translator*

The Pali to Sinhala Translator uses Pali-Sinhala dictionary and identify suitable based word for the existing Pali word. This translator does not consider word level ambiguity and semantics issues. Word is not available in the dictionary translator provides an error message and recorded it as an out-of-vocabulary word. The bilingual dictionary consists of Pali words and related Sinhala words with their grammatical category.

C. *Sinhala Morphological Generator*

The Sinhala Morphological generator uses to generate appropriate Sinhala word for the given Sinhala based word. This Morphological generator uses existing affix spiriting rules that are developed under the BEES project [5] [13]. In here, we use Sinhala Pali sentence, which is available on the Dhamma schoolbooks, therefore, limited numbers of rules are used to generate Sinhala words.

After morphological generating, each word is shown as a translated output of the system. Note that word order of the Pali and Sinhala are closely related to each other. Thus, syntax level generation has not been applied in the current system.

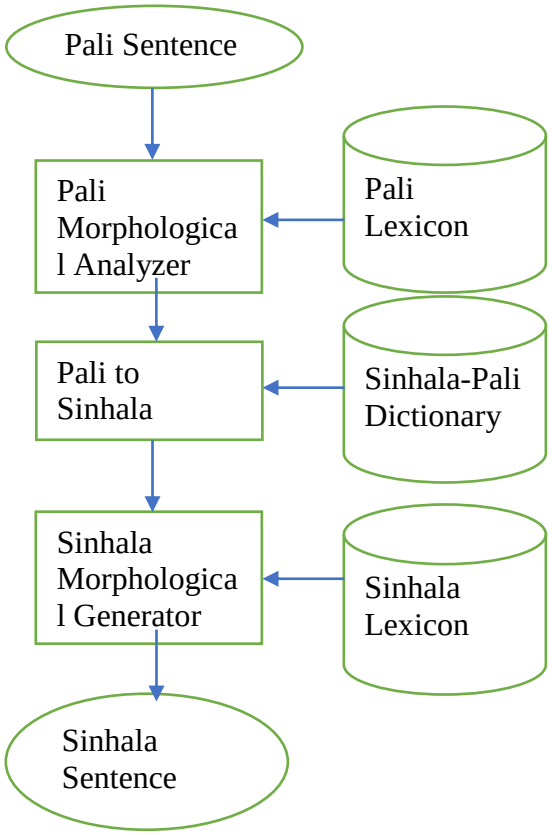


Figure 2: Top-level Design of the Pali to Sinhala Translator

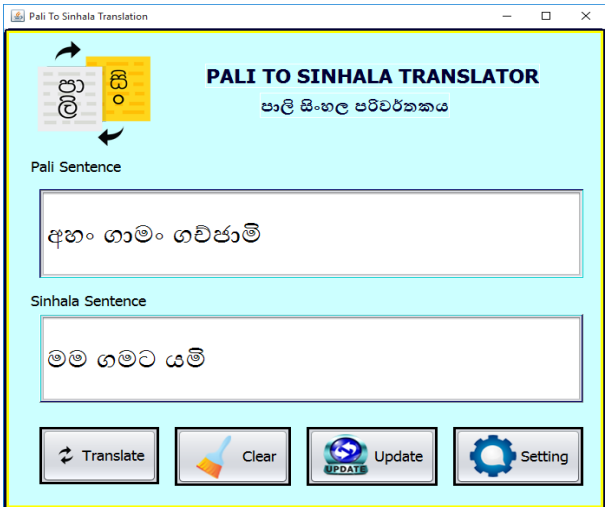


Figure 3: Graphical user interface of the Pali to Sinhala Translator

5. How System works

This section reports how system translates a given Pali sentence into Sinhala. For Instance, System reads "අහං ගාමං ගච්චාමි" (Ahng gamang gachchami) as the input sentence. Then Pali

Morphological analyzer reads words "අහං", "ගාමං" and "ගච්ඡාමි" as the input and identifies based word and the morphology of each word. The table xx shows the result of the Pali morphological analysis. Then Pali-Sinhala bilingual dictionary used to identify each based word for the input base word. Because of the dictionary, based translation system reserved the මම ගම යනවා as the corresponding Sinhala based words. Then Sinhala Morphological generators read these Sinhala words and grammatical information identified by the Pali Morphological generator, generates appropriate Sinhala words for the existing Sinhala based word. After Morphological generation, the translated sentence is shown as "මම ගමට ගමි". Note that, Pali and Sinhala languages are related to each other and syntax of both languages are almost same. Therefore, syntax level attention for translation is not required.

5. Conclusion and Further works

This paper presented design and implementation of the Pali Sinhala translator that was translated Pali sentence into Sinhala with the dictionary-based approach. The system also comprises of Pali Morphological analyzer and Sinhala Morphological generator to analyze and generate Pali and Sinhala words to provide an accurate translation. The Pali Morphological generator has been developed to analyze Pali words. The Pali Morphological analyzer uses affix spiriting rules that are used for word declension. Sinhala Morphological generator also uses Sinhala Morphological rules that are available on the BEES project. This translator uses word-level translation and gives attention only to source and target morphology. The Pali Sinhala translator has been successfully tested for simple Pali sentences as a language learning tool for the grade 6-9 Dhamma schools.

References

[1] "Translation," *Wikipedia*. [Online]. Available: https://en.wikipedia.org/wiki/Translation [Accessed: 18-Oct-2017].

[2] J.Hutchins, "Machine Translation: past, present, future" , New York: Halsted Press, 1986.

[3] "Google Translate." [Online]. Available: https://translate.google.com/. [Accessed: 24-Oct-2017].

[4] A. Bharati, V. Chaitanya, A. P. Kulkarni, and R. Sangal, "Anusaaraka: Machine Translation in Stages," ArXiv Prepr. Cs0306130, 2003

[5] Systran: Past and Present, [Online]. Available: https://lilab.unibas.ch/staff/tenhacken/Applied-CL/3_Systran/3_Systran.html#history, [Accessed: 18-Oct-2017].

[6] B. Hettige and A. S. Karunananda, "Existing Systems and Approaches for Machine Translation: A Review," in *Sri Lanka Association for Artificial Intelligence (SLAAI)*, Moratuwa, 2011.

[7] D. V. Sindhu and B. M. Sagar, "Dictionary Based Machine Translation from Kannada to Telugu," IOP Conf. Ser. Mater. Sci. Eng., vol. 225, p. 012182, Aug. 2017.

[8] N Phonson, a Rule-Based Machine Translation system from Pali to Thai, MSc, Thesis, URL: https://www.ict.mahidol.ac.th /research/thesis/files/2001_25THE.pdf

[9] A. Hatem, N. Omar, and K. Shaker, "Morphological analysis for rule based machine translation," in 2011 International Conference on Semantic Technology and Information Retrieval (STAIR), 2011, pp. 260–263.

[10] B. Hettige and A. S. Karunananda, "A Morphological Analyzer to Enable English to Sinhala Machine Translation," in *International Conference on Information and Automation, 2006. ICIA 2006*, 2006, pp. 21–26.

[11] B. Hettige, "A Computational grammar of Sinhala for English-Sinhala machine translation," M.Phil Thesis, University of Moratuwa,Sri Lanka, Moratuwa, 2011.

[12] A. Pirkola, T. Hedlund, H. Keskustalo, and K. Järvelin, "Dictionary-Based Cross-Language Information Retrieval: Problems, Methods, and Research Findings," Inf. Retr., vol. 4, no. 3–4, pp. 209–230, Sep. 2001.

[13] "The Pali Language and Literature." [Online]. Available: http://www.palitext.com/subpages/lan_lite.htm. [Accessed: 18-Oct-2017].

[14] P. Buddadhatha, *Pali Bshawa Tharanaya*. Rathna Book Prakashakayo, 2015.

[15] J. B. Disanayake, "BasakaMahima 6: Prakurthi", Colombo 10, Sri Lanka: S. Godage and Brothers, 2000.

[16] A. M. Gunasekera"A Comprehensive Grammar of the Sinhalese Language", New Delhi, India : AES Reprint, 1986.

[17] B. Hettige and A. S. Karunananda, "Computational model of grammar for English to Sinhala Machine Translation," in 2011 International Conference on Advances in ICT for Emerging Regions (ICTer), 2011, pp. 26–31.

On Optimizing Deep Convolutional Neural Networks by Evolutionary Computing

M. U. B. Dias[1], D. D. N. De Silva[2], S. Fernando[3]
[1,2,3]Department of Computational Mathematics, University of Moratuwa, Sri Lanka

Abstract

Optimization for deep networks is currently a very active area of research. As neural networks become deeper, the ability in manually optimizing the network becomes harder. Mini-batch normalization, identification of effective respective fields, momentum updates, introduction of residual blocks, learning rate adoption, etc. have been proposed to speed up the rate of convergent in manual training process while keeping the higher accuracy level. However, the problem of finding optimal topological structure for a given problem is becoming a challenging task need to be addressed immediately. Few researchers have attempted to optimize the network structure using evolutionary computing approaches. Among them, few have successfully evolved networks with reinforcement learning and long-short-term memory. A very few has applied evolutionary programming into deep convolution neural networks. These attempts are mainly evolved the network structure and then subsequently optimized the hyper-parameters of the network. However, a mechanism to evolve the deep network structure under the techniques currently being practiced in manual process is still absent. Incorporation of such techniques into chromosomes level of evolutionary computing, certainly can take us to better topological deep structures. The paper concludes by identifying the gap between evolutionary based deep neural networks and deep neural networks. Further, it proposes some insights for optimizing deep neural networks using evolutionary computing techniques.

Keywords: *Deep Networks, Optimization, Evolutionary Computing, Speeding Up Rate of Convergent, Normalization.*

1. Introduction

Training an Artificial Neural Network (ANN) for a given task is a still a demanding research topic in the field of Artificial Intelligence. In order to obtain the highest performance of the network, multi-dimensional optimization is essential which increase the complexity of the problem. The performance of ANN is determined by an aggregation of learning rate, accuracy and generalization. Learning rate is very important as providing an enormous labeled dataset for training. On other hand, the time taken to achieve a given benchmark of accuracy should be reduced as much as possible. Therefore, the learning rate should be maintained properly in order to enhance the performance of ANN. The overall learning rate can be measured by number of epochs and time taken for learning. The accuracy of the ANN is another key component of the performance since the output should be with minimum error after a certain number of epochs. If the final error is beyond the desired margin of error, then the ANN is with poor performance. The error of an ANN is measured by mean squared error, mean absolute error, exponential error method, etc. After all, the designed ANN should be able to generalize well. Generalization means the characteristic of applicability to any problem within a given scenario. In a nutshell, how well it will be able to give the desired output when a new input is given. This generalization is related with the problem of over fitting of ANN. If the ANN is trained more for a particular type of uniform dataset, the accuracy will be substantially increasing for that particular type of input data points. But, when future data point is given, the output will be worst. This incident is called as, over fitting of a network. Therefore, generalization of ANN should be protected throughout the designing and training process. Ultimately, ANN performance can be defined as follows; any well-generalized ANN should be attained for a given subset of problems with high accuracy in less number of epochs.

In order to maximize the performance of ANN, the determination of the optimal network structure and weights for the given task, is a multi-dimensional problem with a vast space of solutions. The process of determining the most appropriate topology of network for the given task is called as structural optimization. The number of hidden neurons and the connectivity between these neurons construct the topology of the network. Some neurons have to be dropped because of their less effectiveness for the network and some neurons have to be added due to their usefulness. Then the feed forward and

recurrent connectivity between these neurons have to be determined in order to construct the complete structure. This connectivity is not about the weight of the connection, but the existence of the particular connection. The structure is usually decided using an ad-hoc choice or some trial and error method. The ad-hoc choice is done by a priori analysis of the task. This is much complicated and non-deterministic due to the lack of a proper model of the task. Furthermore, a conventional ad-hoc structure does not provide the optimum solution for a given task. Apparently, shoehorning the given task to a pre-defined structure is inappropriate; rather the structure should be shoehorned to the given task. The trial and error methods can be categorized as constructive and destructive methods. Constructive methods are initiated with a simple and small structure and later more neurons and connections are added to improve the performance. In contrast, destructive methods initiated with complex and massive network and gradually delete the connections and neurons to make the structure simple and small. These time and error methods are computationally prohibitive and more likely to be trapped in structural local optima because of the stochastic search methods used. Moreover, these search methods are limited for some predefined topologies; hence search through a narrow solution space only. The evolutionary computing comes on stage in this context. Evolutionary computing methods have become successful alternative for topological optimization, due to larger search space, higher speed and more probability to achieve global optima. By using the genetic algorithms and evolutionary programming, some algorithms have outperformed the conventional structural learning methods.

The process of determining the weights of the connections is called as parametric optimization. The weights represent the strength of the connections. These weights are usually found by Stochastic Gradient Descent (SGD) algorithm. The convergent rate and the likelihood to trap in local optima are drawbacks of this method. As per the literature, few researches have been done to apply evolutionary computing methods in parametric optimization.

2. Evolutionary Computing on ANN Training

The research into applying Evolutionary Computing to improve the performance of ANN can be mainly explained under the strategies as representing network in terms of individuals of the population, initialization of population by initial network structure, adoption of proper fitness functions to find optimum network structure, development of parent selection criteria to produce better structure, and reproduction operators. The next couple of sections describe those strategies in detail.

A. Representation of Individuals

In [1], an individual chromosome is interpreted with a bit string which is a combination of several sub strings. The number of bits of a sub string is decided by the granularity. A coarse granularity has a very narrow search space, hence less computational cost. A fine granularity has a large search space, but computationally expensive. Therefore, the number of bits in a sub string is a critical parameter that should be decided corresponding to the given task and network. The first substring of the chromosome indicates the granularity of the string, in binary numbers. Then all other substrings represent the connection links between the neurons. First bit of a substring represents the connectivity. If the connection exists it has value '1' and if the connection does not exist it has value '0'. The rest of the bits of a substring interpret the weight using a binary encoding. Particularly, if the granularity is n, the different number of weight values that can be represented is 2^{n-1}. For example, if the number of bits is 4, it can represent 8 different weight values, such as -2 to 5 with binary encoding 000 ~111 (i.e. -2=000, -1=001, 0=010… 5=111). These weight values are included only when connection are existing. That means, if the connectivity bit is 0 the rest of the sub string will be disappeared. Therefore, the different individuals may have different length of strings. Finally, these substrings are arranged in an order, such a way to keep the substrings, which represent the connections coming to the same neuron, at nearby.

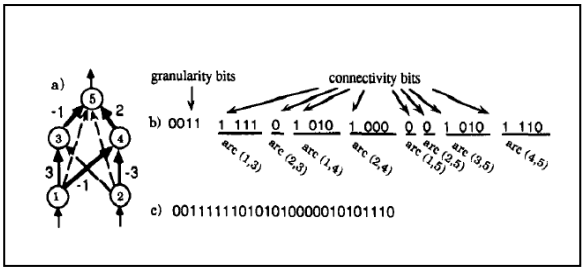


Figure 1: Representation of chromosome in [1]

There are some drawbacks with this representation. As mentioned above the individuals may have different bit lengths and so that difficult to mate two of them straight away. Besides that, this is limited for some range of weight values; consequently, a small search space is utilized. Furthermore, this encoding scheme is not applicable to recurrent networks. Moreover, this representation does not allow add/delete neurons, hence the number of neurons is constant throughout the process. However, the other researches have demonstrated the importance of changing this parameter (i.e. number of hidden neurons). [19] is also use a similar type of genotype, which consist two types of genes. Neuron genes represent the input, output and hidden neurons while connection genes represent the connectivity. Each and every neuron has a particular number that want change throughout the process. The connection genes carry the information of connection input and output neuron number, weight, the availability of connection (connectivity bit), and innovation number (this is explained later).

Since evolutionary programming usually does not use an encoding scheme, [2] and [3] have used real number values to represent the individuals. If the number of input and output neurons are m and n respectively, and maximum number of hidden neurons is N, then size of (m+N+n) x (m+N+n) two matrices are used. One is a binary matrix, which represents the connectivity between the neurons, and other one is weight matrix, which represents the values of particular connections. In [3] these two matrices are upper triangular matrices, because the presented algorithm doesn't applicable for recurrent neural networks. Then, another N dimension vector is used to denote the existence of the hidden neurons. The components of the vector can be either value 1 or 0. If a particular neuron exists it indicates as value '1' and if it doesn't exist it indicates as '0'. Since evolutionary programming uses asexual reproduction, only (no crossover operations) this notation suits well.

B. *Initialization of the Population*

Maniezzo [1] generates an initial population randomly, constrained to the given range of granularity and the given number of neurons. The other researchers [2] and [3] randomly select the number of hidden neurons and the number of connection links, from uniform distributions over user defined ranges. Then, the weight values are also generated from a uniform distribution over a small range. [3] does a further modification after generating the initial population. It trains the population partially using BP and then if the error is reduced in a particular individual then it is marked as "success" or otherwise as "failure". [19] initialize the networks without hidden neurons. The population begins with the simplest network and the hidden neuron are added according to the performance.

C. *Fitness Function*

For supervised learning there are three possible ways to measure the fitness [2]. Summation of square error (1), Summation of absolute error (2), or Summation of exponential error (3) of the n^{th} individual can be used to measure the fitness of the particular individual. t_i is the targeted output of i^{th} labeled data point used for training, and a_i is the actual output for the particular input. The superscript λ denotes an individual of the population.

$$E_{sqr}^{\lambda} = \sum_{i} (t_i^{\lambda} - a_i^{\lambda})^2 \qquad (1)$$

$$E_{abs}^{\lambda} = \sum_{i} |t_i^{\lambda} - a_i^{\lambda}| \qquad (2)$$

$$E_{exp}^{\lambda} = \sum_{i} e^{|t_i^{\lambda} - a_i^{\lambda}|} \qquad (3)$$

Yao and et al [3] uses a more advance method to evaluate the individuals. It calculates the mean square error percentage and normalizes by the factors of number of output neurons and the range of the output values. This method is adopted from [4]. If the number of output neurons is n the number of data points used for validation is T, and the maximum and minimum values of the outputs are $O_{\max}$ and $O_{\min}$, the error for individual λ can be calculated as follows (4);

$$E^{\lambda} = 100.\frac{O^{\lambda}_{max} - O^{\lambda}_{min}}{n}.\frac{E^{\lambda}_{sqr}}{T} \qquad (4)$$

D. Parent Selection

Maniezzo [1] has used a very primary parent selection method; i.e. randomly pair the individuals of the existing population. Then, each and every pair will undergo a crossover operation. The mutation operations are implemented on each individual with a pre-defined probability. However, [3] uses a rank based method to select a parent. Every individual of the population is given a rank from 0 to (M-1) from the descending order of the fitness, i.e. the ascending order of the calculated error. It has reduced the computational load by avoiding taking the reciprocal of the error to calculate the fitness value. The probability of the λ th individual to be selected is (5);

$$P(\lambda) = \frac{M - \lambda}{\sum_{k=1}^{M} k} \qquad (5)$$

The main drawback of this method is; it assumes a uniform distribution of error values, even though it is not. The others [2] use a very simple method to select the parents. The fittest half of the existing population is used to produce the offspring of the next generation.

E. Reproduction Operators

Each and every parent undergoes for 3 type of mutations in [1]. Granularity bit mutations, connectivity bit mutations, and weight bit mutations are executed with three user defined mutation probabilities. The n point crossover operator is used for sexual reproduction. Although individuals may have different lengths of bit strings to represent them, all of them are stored in maximum possible fixed length memories. That is the string length of an individual with maximum possible granularity and maximum connectivity. For example, if the maximum possible connections are 5 and maximum granularity is 3 then, then the maximum possible length will be 22, with two bits to represent the granularity. The individuals with less number of bits just don't use the rest of bits to represent their network. The crossover operator is implemented using these fixed length bit strings, so that no problem of mating different size of individuals. However, one drawback of this operator is the lacking of compositionality property. Compositionality is the meaningfulness of a portion of a string.

Some researchers [2], [3] use asexual reproduction only. Two types of mutations are performed in order to obtain a valued offspring; i.e. parametric mutations and structural mutations. Rather using fixed probabilities like [1], [2] yield the probabilities by uniform distributions. First it calculates a 'temperature value' (T) for particular parent using its fitness value. If the maximum attainable fitness value is f_{max} and fitness of the λ individual is f_{λ} then,

$$T_{\lambda} = 1 - \frac{f_{\lambda}}{f_{max}} \qquad (6)$$

gives the temperature value for that parent. Then it calculates an instantaneous temperature value ($\widehat{T_{\lambda}}$) for every mutation operator implementation.

$$\widehat{T_{\lambda}} = U(0,1)T_{\lambda} \qquad (7)$$

where $U(0,1)$ is a random variable chosen from a uniform distribution over the interval [0,1]. The weights are updated choosing a random variable from a normal distribution;

$$W_{new} = W_{old} + N\left(0, \alpha\widehat{T_{\lambda}}\right) \qquad (8)$$

where, α is a user-defined proportion. There are four kinds of structural mutations used in [2] and [3], those are; adding a hidden neuron, deleting a hidden neuron, adding a connection link and deleting a connection link. [2] apply these mutations on parents with particular number calculated using pre-defined interval of $[\Delta_{min}, \Delta_{max}]$ for each four structural mutation type. This particular number for each individual is given in eq.(9).

$$\Delta_{min} + \left[U(0,1)\widehat{T_{\lambda}}(\Delta_{max} - \Delta_{min})\right] \qquad (9)$$

The researcher [3] uses a hybrid training method of back propagation(BP) and simulated annealing(SA), for parametric mutation. When a parent is selected, if it is marked as "success" then it undergoes for further BP training and no further any mutation is done. Else if it is marked as "failure" then it is trained using SA and update whether success or failure. If it's success then no further mutations are performed, but if it's still "failure" structural mutations are performed followed by partial BP training in the

sequence of hidden neurons deletion, connection links deletion, connection links addition and hidden neuron addition. These structural mutations are executed according to the survivor selection method described in the next subtopic. The number of hidden neurons and connection links to be deleted is random values chosen from small uniform distributions defined by the practitioner. These generated numbers of neurons are deleted uniformly at random. The connection links are deleted according to the importance of a connection. The importance of a connection is evaluated by a variable called 'test' of that particular connection. When T number of validation patterns is used, if the particular connection has a weight of $'w'$ then;

$$test(w) = \frac{\sum_{i=1}^{T} \xi^i}{\sqrt{\sum_{i=1}^{T} \left(\xi^i - \overline{\xi}\right)^2}} \qquad (10)$$

where $\xi^i = w + \Delta w^i$ and $\Delta w^i = -\eta \frac{\partial E_{abs}}{\partial w}$. $\overline{\xi}$ is the average value of ξ^i over T number of validation patterns. According to this test value, the connections are deleted. Same value is used for addition of connection links and these adding connections are selected from the connections, which are currently with zero weights. In neuron addition, a process called "cell division" is used. An existing neuron is spitted into two parts and following weight vector updating is carried out.

$$\begin{aligned} w_{ij}^1 &= w_{ij}^2 = w_{ij} && i \geq j \\ w_{ki}^1 &= (1+\alpha)\, w_{ki} && k > i \\ w_{ki}^2 &= -\alpha w_{ki} && k > i \end{aligned} \qquad (10)$$

Where w^1 and w^2 are weight vectors of new neurons and α is a mutation parameter which may be either fixed value or random value.

One of the main disadvantages of these structural mutations is the generational gap. That means the huge behavioural differences between the parents and the offspring. [2] introduces the new hidden neurons without any connections and connections with zero weights preventing radical jumps in fitness values in these two types of mutations. But, it claims that in addition of connection links/neurons these sudden changes are inevitable. After all four types of structural mutations, [3] use a partial training with BP to avoid these sudden behavioural changes. Additionally, the added connection links are initialized with small weight values in [3], in contrast to zero initial weights in [2], because of the partial training with BP.

[19] uses two structural mutations only; i.e. addition of connection and addition of hidden neuron. Connections are added with random weight values. The new hidden neurons are added by splitting the existing connections. The new neuron will get an input connection with a weight of 1 and output connection with weight of old connection weight. Preserving the old weight value will reduce the generational gap between the parents and the offspring. Whenever one of these mutations are occurred, a new gene is added to the chromosome, which leads to vary the size among the individuals. Every new gene is given a number called "innovation number" which is incremented in every single mutation. This number of a particular gene, want change in the entire process. Hence, the historical data will be preserved and can be utilized whenever needed. This feature is not available in any other method. These data is used to line up the individuals with different sizes in order to implement the crossover operation.

3. Optimization Techniques in Deep Convolution Networks

The techniques that have been used in shallow neural networks have been further advanced and applied in deep neural networks. Especially, normalization techniques which were used to normalize the inputs in conventional neural networks, has been extended to weight normalization as well as intermediate output normalization. Further, different momentum algorithms have been derived to speed up the convergence of deep networks, the next couple of sections sum up the common approaches that have been practiced to smooth and speed up the learning process of deep networks. Here, the paper focuses only on convolutional neural networks (CNN) because it has been recognized as a key approach for object recognition.

A. Normalization Techniques

In deep networks, input to each layer is affected by parameters of previous layers, as network parameters changes (by training), even small changes to the network get amplified down the network. This leads to change in the

statistical distribution of inputs to following layers from previous layers, therefore hidden layers will keep trying to adapt to that new distribution, hence slows down the convergence and make it difficult for training. This may lead to requirement of low learning rates and careful parameter initialization. This is known as internal covariant shift. As a general solution, it introduced to normalize the data (by mean and variance) and several normalization techniques have been tried out.

1) Normalization

Normalize data by both mean and variance is a major technique, which simply is to transform to make data with zero mean and unit variance hence de-correlated, through a series of linear transformations. The process centers, the data around value zero by subtracting the mean and then divide by the standard deviation for scaling. In general, subtracting the mean across every individual feature in the data, and make geometric interpretation of centering the cloud of data, around the origin along every dimension (k). To normalize the data dimensions so that they are of approximately the same scale, divide each dimension (k) by its standard deviation after they have been zero centered. This is also known as simplified whitening process. This simplified whitening only removes mostly the first order covariant shift, but for removing higher order shift requires complex techniques have been introduced.

2) Local contrastive normalization

Local contrastive normalization (LCN) performs a local competition among adjacent features (like pixels) in feature maps and between features at the same spatial location on different feature maps. LCN applies after introducing the non-linearity (ReLu) for whitened data.

For example, let us consider a local field of 3x3-area portion (9 pixels) to clarify the process. First, for each pixel in a feature map, find its adjacent pixels (radius is 1 in this case), so there are 8 pixels around the target pixel in the middle (do the zero padding if the target is at the edge of the feature map). Then, compute the mean of these 9 pixels (8 neighbor pixels and the target pixel itself), subtract the mean for each one of the 9 pixels. Next, compute the standard deviation of these 9 pixels. In addition, judge whether the standard deviation is larger than 1. If larger than 1, divide the target pixel's value by the standard deviation. Otherwise, keep the target value as what they are (after mean subtraction). At last, save the target pixel value to the same spatial position of a blank feature map as the input of the next layer of the deep CNN.

LCN introduces a competition among the output of adjacent convolution kernel. This normalization technique is useful when it deals with ReLU neurons because ReLU neurons have unbounded activations and needs local responsive normalization to normalize that. For detecting high frequency features with a large response, normalizing around the local neighborhood of the excited neuron, it becomes even more sensitive as compared to its neighbors [12].

3) Batch Normalization

Batch normalization (BN) is a learnable whitening process that normalizes the inputs to each following hidden layer so that their distribution kept fairly constant as training proceeds, hence improves training and allows faster convergence. About input distributions, BN algorithm addresses the changing distributions issue which known as internal covariant shift and allows using higher learning rates. These learnable hyper parameters in BN transformation are learned through back propagation during online or mini-batch training. Furthermore, Batch normalization reduces effects of exploding and vanishing gradients while regularize the model. Without BN, low activations of one layer can lead to lower activations in the next layer, and then even lower ones in the next layer and so on [11].

At the beginning batch normalization initialize (with mini batches) by normalizing the data using calculated mini batch mean and variance hence standard deviation Not just normalizing each input of a layer may change what the layer can represent. To address this, it introduced a transformation, for each normalization, a pair of parameters, which scale and shift the normalized value. These parameters are learnable during the training using back propagation and by setting them equal to standard deviation and mean respectively it could even recover original activations. These learned parameters in transformation depend on all the training examples in the mini batch. Also, learning

ensures that as the model trains, layers can continue learning on input distributions that exhibit less internal covariate shift, thus accelerating the training.

B. Momentum Updates

1) Vanilla momentum update

Momentum is one of the key approaches that have been applied to faster the convergence of deep networks getting out of local minima. With momentum m, the weight update at a given time t, m adds a fraction of the previous weight update to the current one as shown in eq.20. When the gradient keeps moving into the same direction, m increases the size of the steps toward the minima. On contrast, when the gradient changes the direction compared to previous few steps, momentum help to smooth out variation.

$$\Delta w(t) = -\alpha.\frac{\partial E}{\partial w} + m.\Delta w(t-1) \quad (20)$$

2) Nesterov momentum update

Nesterov momentum is a derivative of momentum updates, which performs well in convex functions as in eq.21 and eq.22. The basic idea of Nesterov momentum is to compute the gradient of the future approximate position than the current position of the parameter. This accelerated momentum helps to rush the network to its convergence [5, 9].

$$w^* = w(t) + m.\Delta w(t-1) \quad (21)$$

$$\Delta w(t) = -\alpha.\frac{\partial E}{\partial w^*} + m.\Delta w(t-1) \quad (22)$$

C. Effective Receptive Field

Particularly in CNN, behaviour of a unit depends only on a region of the input, which is called a receptive filed of the unit. Theoretically, size of the receptive field of an output unit can be increased by stacking more layers to make much deeper network, or by sub-sampling, which increases the size of receptive field. Deeper analysis into receptive fields has shown that pixels in a receptive field do not contribute equally to response of the output unit. Indeed, the central pixels in the receptive field make much larger impact on the response rather than the boundary pixels. These pixels, which make larger impact on the output, is known as effective receptive field (ERF) [6]. Further, these effective receptive fields are Gaussian distributed and growing bigger in size during the training. Since ERF are smaller at initial stages, and causes problems for the tasks that require a larger receptive field, the weights are suggested to initialize in lower scale at the center and in higher scale at the outside. However, an architectural change to CNN has also been suggested in terms of defining convolution window which emphasis the need of defining window, which connect larger area with some sparse connection, rather than defining small local convolution window [10].

http://www.erogol.com/dilated-convolution/

D. Dropout training

Model combination always improves the performance of neural networks. With large neural networks, averaging the outputs of many separately trained nets is expensive. Training many different architectures is hard, because finding optimal hyper-parameters requires a lot of computation. Large networks normally require large amounts of training data and there may not be enough data available to train different networks on different subsets of the data. Dropout is a technique that prevents over-fitting and provides a way of approximately combining exponentially many different neural network architectures efficiently. Dropout refers to dropping out units (hidden and visible) in a neural network. Training a network with dropout and using the approximate averaging method at test time has led to significantly lower generalization error [15].

E. Annealing Learning Rate

In deep networks, many researchers have tried to anneal the learning rate over time. Knowing when to decay the learning rate is very tricky. Decaying the learning rate slowly can cost high computational time and on the other hand decaying the learning rate aggressively may not help the system to reach the global minimum. Step decay, and exponential decay are common types of implementing the decay. Step decay reduces the learning rate by some factor every few epochs. Exponential decay reduces the learning rate exponentially over the number of iteration. Adaptive learning rates such as Adagrad [13], RMSprop and Adam [14] are some of the significant learning rates that have been proposed to improve the performance of Deep neural networks.

4. Evolutionary Strategies in Deep Convolution Networks

Optimizing topological structure of deep neural networks has become more challenging task as the increase of the depth of the network. No longer it is feasible to find the optimum structure for deep networks for a given problem manually and has opened up new branch in deep networks to look into the possibility of automating the task. As we discuss in early sections, many approaches have been proposed to optimize the deep networks, however Evolutionary Computing (EC) based approaches are being called back to overcome the difficulty. Application of EC into reinforcement learning on Atari games [17], and EC into Long-Short-Term-Memory (LSTM) on image capturing and language modeling [16] are some of the significant recent applications which could confirm the possibility of improving efficiency of deep networks using EC based techniques.

Among them, few researchers have attempted to improve the topological structure of deep CNN. Miikkulainen et al [16] has evolved a CNN network which could reach the convergence after 30 epochs and it is compared to 4 times faster than the manually defined approach for the same problem set (CIFAR-10). In their approach, a convolution layer has been defined as a node in chromosome in evolutionary programming. Problem was solved as multi-population problem that evolves structure and then hyper-parameters for the assembled network. However, in this approach optimizing of hyper-parameters were done after the network structure was assembled by genetic programming. Suganuma et al[18] has applied Cartesian genetic programming to evolve the CNN in image classification task. In addition to convolution blocks, batch normalization and residual blocks have been considered in evolving the topological structure. However, the hyper-parameters of the network have been optimized after assembling the network.

5. Discussion

As the performances of deep neural networks increases, in general the number of layers in deep neural networks is also increases. That is the more the number of layers; the more accuracy has been recorded. Optimizing the deep neural network's topological structure for a given data set has become a challenging task for neural network researchers. Therefore, many researchers are now exploring the possibility of applying genetic programming or evolutionary algorithm to evolve deep neural networks.

In conventional neural networks, this has been mainly achieved by applying evolutionary algorithm. In there, the network hyper-parameters are encoded into genes of chromosomes, and then have tried to optimize the parameters based on a defined fitness-function. Or else, the network topological structure has been tried to encode into chromosomes of genetic algorithm and tried to optimize the network topological structure. However, these approaches are not good enough to optimize the deep neural networks, because compared to conventional neural networks, deep neural networks has variety of architectures combing many different modules or layers. For example, residual layer, normalization layer, inception module, etc. Therefore, new approaches have been investigated to optimize the deep neural networks' topological structure or hyper-parameters. Few researches have encoded the modules of deep neural networks into chromosomes and tried optimize the neural network structure. However, those techniques are required further researches to evaluate their performances thoroughly.

Here the paper proposes to optimize the deep neural network structures as a collective intelligence of genetic programming: representing deep neural networks in terms of acyclic graph in detail (into neuronal level and connection) using genetic programming. As per the concept governed by swarm intelligence, each module or layer in the network is identified as a local genetic program. Hence the entire network can be considered as multiple genetic programs. Each entity is trying to optimize its behaviour locally under the constraint of optimizing overall behaviour of entire network. In this way, we can approach the deep neural network optimization problem, as emergent feature of collective multi-genetic programs.

References

[1] V. Maniezzo, "Genetic evolution of the topology and weight distribution of neural networks," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 39–53, Jan. 1994.

[2] P. J. Angeline, G. M. Saunders, and J. B. Pollack, "An evolutionary algorithm that constructs recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 54–65, Jan. 1994.

[3] X. Yao and Y. Liu, "A new evolutionary system for evolving artificial neural networks," *IEEE Trans.*

Neural Networks., vol. 8, no. 3, pp. 694–713, May 1997.
[4] "1994-21.pdf."
[5] Yu Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence, vol. 269, pp. 543–547, 1983.
[6] W.Luo, Y. Li, R. Urtasun, and R. Zemel, Understanding the Effective Receptive Field in Deep Convolutional Neural Networks, *29th Conference on Neural Information Processing Systems (NIPS 2016)*, Barcelona,Spain.
[7] D.Mishkin, N.Sergievskiy, and J.Matas, Systematic Evaluation of Convolution Neural Network advances on the ImageNet, Computer Vision and Image Understanding, 161 (2017) 11-19.
[8] E.A. Smirnov, D.M. Timoshenko, and S.N. Andrianov, Comparison of Regularization Methods for ImageNet Classification with Deep Convolutional Neural Networks, Conference on Intelligence and Biometrics, 6 (2014) 89-94.
[9] I.Sutskever, Martens, J,G.Dahl and G.Hinton, On the importance of initialization and momentum in Deep Learning, in proceeding of the 30th International Conference on Machine Learning, 2013.
[10] F.Yu and V.Koltun, Multi-Scale context aggregation by dilated convolutions, ICLR, 2016.
[11] S. Joffe and C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, 2015.
[12] A Krizhevsky, I Sutskever, and G. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, 2015.
[13] J. Duchi, E. Hazan, and E.Singer, Adaptive Subgradient Methods for Online Learning and Stochastic Optimization, JMLR, 12(Jul):2121−2159, 2011
[14] D.P. Kingma, and J. Ba, Adam: A Method for Stochastic Optimization, International Conference for Learning Representations, San Diego, 2015.
[15] H.Wu and X.Gu, Towards dropout training for convolutional neural networks, Neural Networks, 71 (2015) 1-10.
[16] R.Miikkulainen, J.Liang, E.Meyerson, A.Rawal, et. at., Evolving Deep Neural Networks, 2017.
[17] T. Salimans, J. Ho, Xi Chen, S. Sido and I. Sutskever, Evolution Strategies as a Scalable Alternative to Reinforcement Learning, 2017.
[18] M.Suganuma, S.Shirakawa, and T.Nagao, A Genetic Programming Approach to Designing Convolutional Neural Network Architectures, Neural and Evolutionary Computing, August 2017.
[19] K. O. Stanley and R. Miikkulainen, "Evolving Neural Networks through Augmenting.

Clustering-based Augmented Tabu Search for Energy Minimizing Vehicle Routing Problem (EMVRP)

G. W. H. H. P. Rathnakumara[1], T. D. Rupasinghe[2]
[1,2]Department of Industrial Management, University of Kelaniya, Sri Lanka
[1]hasinirk@gmail.com, [2]thashika@kln.ac.lk

Abstract

Application of metaheuristics for Energy Minimizing Vehicle Routing Problem (EMVRP) has become extremely important because of practical relevance. The EMVRP aims at serving dynamic demanding customers distributed throughout the world while consuming minimum possible energy, which represent as a product of load and distance. The NP-hard problem nature urges the researchers to utilize metaheuristics to solve them in non-polynomial time. Although there exists several metaheuristics, as of to date the EMVRP is only formulated and solved using the Genetic Algorithm (GA). According to the authors' knowledge there is only one instance reported in literature of metaheuristic based on GA on EMVRP. There are no studies reported on the development of TS based metaheuristic for the EMVRP and use of machine learning to guide the local search in formulating EMVRP. In this study, at first the authors formulate EMVRP using Tabu Search (TS) and evaluate its performance with test cases from the CVRPLib repository for the vehicle routing problems (VRPs). As the second stage, the authors improve the local search in the data set embedding machine learning (ML) techniques in the developed TS algorithm and critically evaluate the applicability using the same test cases. The study introduces a new TS based formulation for EMVRP for the literature as well as derives the fact of augmenting ML brings super performance in metaheuristics in solving EMVRP. Furthermore, the study lays foundation in filling the knowledge gaps in the current literature and proposes future research directions in machine learning augmented metaheuristics to enhance solving the EMVRPs.

Keywords *- Vehicle Routing, Machine Learning, Tabu Search*

1. Introduction

Energy Minimizing Vehicle Routing Problem (EMVRP) comes under combinatorial optimization and importance is gained importance during the past years due to its relevance with the developing industrial world. It is one of the most known np-hard problem which finds optimal route to satisfy dynamic demanding customers distributed throughout the world from a given distribution center while minimizing the total energy [4]. NP-hard problems such as EMVRP is solved using techniques of heuristics because they cannot be solved in a realistic computational time by using exact mathematical models due to its increased variables and complexity [6]. The term-met heuristic was first introduced in [10]. Metaheuristic is a class of heuristic which is more realistic in reaching global optimum due to its nature of accepting temporary deterioration of solutions, which allows them to explore more thoroughly the solution space and thus to get a better solution [5].

EMVRP is mathematically formulated in [1] with the objective of minimizing energy consumption while serving distributed set of customers and the study highlights the computational times of CPLEX for even moderate sized problems are not feasible.

According to [1] the real cost of a vehicle traveling between two nodes depends on many variables, the load of the vehicle, fuel consumption per mile (kilometer), fuel price, time spent or distance traveled up to a given node, depreciation of the tires and the vehicle, maintenance, driver wages, time spent in visiting all customers and total distance traveled etc. As most of the above-mentioned variables are distance or time based and can be approximated by the distance alone. The variables which are not a constant within the route can be vehicle load, fuel consumption per mile (kilometer), fuel price or time spent up to a given node. In a recent study, these variables are presented as a function of the load of vehicles on the corresponding arc being energy of a given route to be equal to the product of distance travelled and the load of the vehicle [1].

The research presented by [7] introduce variety of approximate algorithms, including classical heuristics and metaheuristics to efficiently solve different variants of VRP. The research of [12] is one such special VRP context where it objects to minimize the number of vehicles while minimizing the total route duration. The study of [13] is a heuristic based solution to make constructive routes to pick up and deliver goods based on time windows. Application of metaheuristic for the variant of EMVRP is first ever published by [3] where GA is used as the type of metaheuristic to formulate EMVRP. Literature is novel to EMVRP formulation with other types of metaheuristic. An analysis of metaheuristic application for EMVRP for the research history is done in the study of [9]. It brings out the fact that metaheuristic application on EMVRP other than GA is novel to the literature and no ever publication done in that regard. The authors attempt at bring TS for EMVRP to the literature which will eye open further research domains. Use of information on the past progress of the search in maximizing local search comes under the metaheuristic of Tabu Search (TS), [8]. A distinctive feature of TS compared to memory-less metaheuristics is the use of adaptive memory to explore the search space in a responsive manner. According to [11] findings, TS can use both explicit and attributive memory.

Augmenting metaheuristic using ML for EMVRP is a new research domain. The study of [2] proposes a comparative analysis of different clustering techniques where grouping of similar objectives to clusters. Augmenting metaheuristic with ML is first reported by [3] using k-means clustering for the EMVRP domain. This was carried out tuning the main parameters, which are mutation rate, population size, and number of generations for the better performance of the parameters in concern to improve overall performance of the metaheuristics. Except for [3] literature is silent on the application of ML augmented metaheuristics to solve EMVRP or related variants.

Table 1. Main Stages of Research

Stage	Description
Phase 01	Formulating the EMVRP problem using TS (Generic TS)
Phase 02	Augmenting machine learning concepts to the Generic TS to enhance the performance (ML-TS)
Phase 03	Comparison of energy obtained with Generic TS with ML-TS to identify any performance improvement using test cases extracted from the CVRP Library

2. Methodology

A. *Phases of Research*:

The research flow is designed mainly focusing on 3 stages. First, a new TS metaheuristics is developed and the results for elite energy, which is the lowest achievable energy is obtained. Then, the developed metaheuristic is augmented with ML by guiding local search and elite energies are obtained. Lastly, the comparative study focusing on two methods is done and identify new knowledge gaps to the literature. Main stages of research can be represented as in Table 1.

B. *Data Collection*:

The test data is obtained from a vehicle routing problem library CVRPLib publically available in the internet under the link of http://vrp.atd-lab.inf.puc-rio.br/index.php/en/ and this repository is a prominently used library in many research studies found in the literature. The repository consist data collections of various no of customers class, each class with data sets with specific customer demands, vehicle capacity, and exact geographical location of the customers. The authors has selected an average no of customer class (about 32-80 customers) for testing in result generation.

C. *Development Environment*:

The experiments were carried out on a personal computer with the following configurations of Core i5, with 8GB RAM, and 2.30GHz-2.40GHz CPU speed. Left and right-justify your columns. Use tables and figures to adjust column length. Use automatic hyphenation and check spelling. All figures, tables, and equations must be included *in-line* with the text. Do not use links to external files.

3. Results

A. Phase 01- Formulating EMVRP problem using TS

Generic TS algorithm for the formulation of EMVRP is represent in Figure1.

//Assumptions

Energy is assumed as the product of load and the distance travelled

Weight of the vehicle is 0

There is only a central depot

Every vehicle starts and ends in the depot

Vehicle Capacity is C

//Input

Read list of cities to be served with demands {city1,city2,…,city} from CVRP Lib file Read vehicle capacity from text file

Number of iteration (i)

Number of solutions (n)

Size of tabu list (10)

//Initialization

Take starting point as depot

Create a tour with a random order of cities

Estimated rounds=total energy/capacity

Initial energy = energy consumption of the solution

Loop start {

//Tabu Search

Do this for defined no of times to create a solution pool

Get the best tour from the solution pool

Save as the elite

Energy=energy of the fittest tour in solution pool

//Maintain tabu list

Remember lastly visited cities by the vehicle

Check the next move

If the move is tabu, cut them out in the search

Else if if possible untabu move, go for untabu move

Update the tabu list

Else if no any untabu move

Apply Freeing strategy

//Stopping criteria

Check the number of solutions obtained

While number of solutions is equal to i

Add one more iteration

} End loop start

Figure. 1. Generic TS algorithm for EMVRP

B. Design of Experiment for parameter setting of TS

Design of experiment (DOE) is carried out to select ideal parameter values in obtaining minimized energy with practical relevance.

1) Setting number of iterations

To execute the series of experiments with the CVRP test cases, different number of iteration levels were identified through the Design of Experiments (DOE). The results are presented in Table II.

2) Identifying the energy obtained for each iteration

For each test case, the energy is obtained and identify the best matching iteration level for the study. See Table 2.

Table 2. Energy Obtained For Different Iteration Levels

CVRPLib file	Energy(J) obtained for various iterations considered in the DOE			
	10	100	1000	100000
A-n32-k5.vrp	82467	69588	65275	52065
A-n33-k5.vrp	73065	71548	68312	51201
A-n33-k6.vrp	63340	70347	62139	53890
A-n34-k5.vrp	71789	69281	68391	54139
A-n36-k5.vrp	73629	71341	66288	53189
A-n37-k5.vrp	80186	75904	67190	54399
A-n37-k6.vrp	82950	78491	71291	55193
A-n38-k5.vrp	85006	82132	73190	51298
A-n39-k5.vrp	81298	82467	73575	58104
A-n39-k6.vrp	93020	79978	75198	56104
A-n44-k5.vrp	96424	79296	81211	75120
A-n45-k6.vrp	114476	85195	86241	87368
A-n45-k7.vrp	100666	93184	99761	83190
A-n46-k7.vrp	102963	96929	92078	83094
A-n48-k7.vrp	112557	99159	88241	84201
A-n53-k7.vrp	130355	111328	86131	81445
A-n54-k7.vrp	128316	116288	101901	88412
A-n55-k9.vrp	130226	123899	120458	86945
A-n60-k9.vrp	126910	124902	133073	101405
A-n61-k9.vrp	120946	132199	128394	105399
A-n62-k8.vrp	159454	141271	129385	112901
A-n63-k9.vrp	158459	143199	132787	121678
A-n63-k10.vrp	150705	148134	144283	121328
A-n64-k9.vrp	140798	142908	138779	126803
A-n65-k9.vrp	164953	150344	153373	119045
n65-k9.vrp	168469	148299	142254	119480
A-n69-k9.vrp	177771	151236	142352	128357
A-n80-k10.vrp	197167	148913	143178	126983

C. Phase 02– Combining machine learning to enhance the performance of formulated EMVRP with TS and Acronyms

ML-TS algorithm for the formulation of EMVRP is represent in Figure 2. The special feature in ML-TS algorithm than the Generic algorithm is it make internal search more intelligent by making clusters of customers who are in nearby locations and with under optimized level of constrain of demand which satisfying the vehicle capacity.

//Input
Read list of cities to be served with demands {city1,city2,…,city}
from CVRP Lib file
Vehicle Capacity is C
Number of iteration (i)
Number of solutions (n)
Size of tabu list (10)
Number of clusters is X

//Setting no of clusters
Check the demand of each cities
Total demand=sum of demand of all cities
X =(Total demand / C) + 1

//Create city lists
Set starting city as depot
While all the cities are counted
And if next city=! depot
Add each city to the city list

//Create cluster cities
Select x random cities as centers of clusters
While citylist is empty
Add cities to each cluster IDs of x

//Redefine the centers to obtain lowest possible distance
Get the cluster cities
Calculate the centres of each clusters
Until cluster centers are get fixed
Redefine the new cluster centres by calculating the best middle
Again list the cities to new cluster IDs of X
Obtain the clusters with lowest possible distance to move on

//Get current energy
Current enegy=enrgy to visit the city list while satisfying the demands of cities
While currentEnergy < Elite Energy
Set Elite Energy=Current Energy
Set Elite tour=tour of city list

//Tabu Search
Do this for defined no of times to create a solution pool
Get the best tour from the solution pool
Save as the elite
Energy=energy of the fittest tour in solution pool

//Maintain tabu list
Remember lastly visited cities by the vehicle
Check the next move
If the move is tabu, cut them out in the search
Else if possible untabu move, go for untabu move
Update the tabu list
Else if no any untabu move
Apply Freeing strategy

//Stopping criteria
Check the number of solutions obtained
While number of solutions is equal to i
Add one more iteration
} End loop start

Figure. 2. ML-TS algorithm for EMVRP

D. Phase 03 – Comparative study of Generic TS Vs ML-TS

The Generic TS and ML-TS was compared against each other in order to identify any significant improvement and is presented in Table 3.

Table 3. Energy Obtained With Generic Ts and Ml-Ts Ts

CVRPLib file	Elite energy with Generic TS (Kj)	Time taken to obtained energy(Sec)	Elite energy with ML-TS(Kj)	Time taken to obtained energy (Sec)
A-n32-k5.vrp	52065	3.2	28560	303
A-n33-k5.vrp	51201	3 3	29789	306
A-n33-k6.vrp	53890	3.3	29045	308
A-n34-k5.vrp	54139	3.3	31284	308
A-n36-k5.vrp	53189	3.6	32149	306
A-n37-k5.vrp	54399	3.6	32798	306
A-n37-k6.vrp	55193	3.5	33892	308
A-n38-k5.vrp	51298	3.5	36782	12
A-n39-k5.vrp	58104	3.6	35123	314
A-n39-k6.vrp	56104	3.6	33290	329
A-n44-k5.vrp	75120	3.6	34295	341
A-n45-k6.vrp	87368	3.6	34890	328
A-n45-k7.vrp	83190	3.7	37821	327
A-n46-k7.vrp	83094	3.7	41294	329
A-n48-k7.vrp	84201	3.7	43193	341
A-n53-k7.vrp	81445	3.7	42159	338
A-n54-k7.vrp	88412	3.7	45248	345
A-n55-k9.vrp	86945	3.9	46193	342
A-n60-k9.vrp	101405	3.9	46890	339
A-n61-k9.vrp	105399	4.1	48923	346
A-n62-k8.vrp	112901	4.0	52134	348
A-n63-k9.vrp	121678	4.1	54389	351
A-n63-k10.vrp	121328	4.2	51239	357
A-n64-k9.vrp	126803	4.2	52489	359
A-n65-k9.vrp	119045	4.2	54256	362
A-n65-k9.vrp	119480	4.2	63295	363
A-n69-k9.vrp	128357	4.3	66390	363
A-n80-k10.vrp	126983	4.5 s	69578	368

4. Discussion

A. Comparison of Generic TS Vs ML-TS

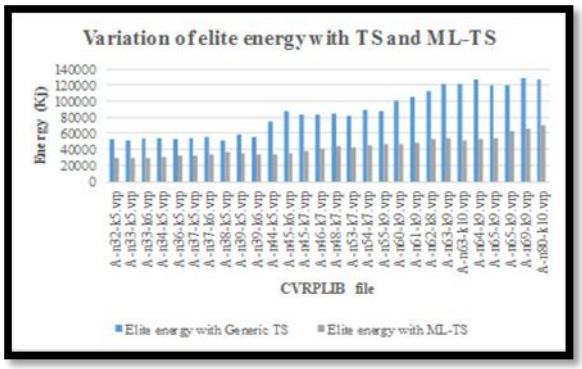


Figure 3. Variation of elite energy with TS and ML-TS

With the results obtained it is identified an

improvement in elite energies obtained with two algorithms and also a increment in computational time as depicted in Figure 3 and Figure 4.

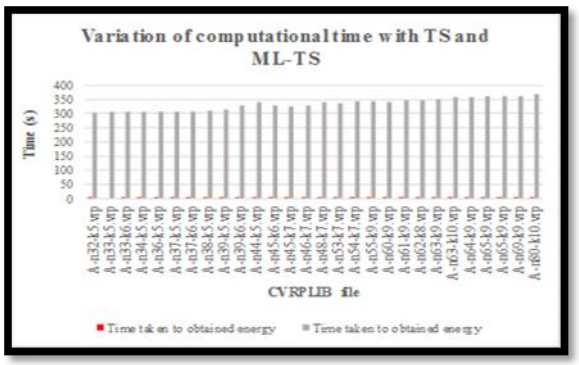


Figure 4. Variation of computational time with TS and ML-TS

Figure 5 represents the tradeoff between the computational time increment and elite energy improvement with TS and ML-TS.

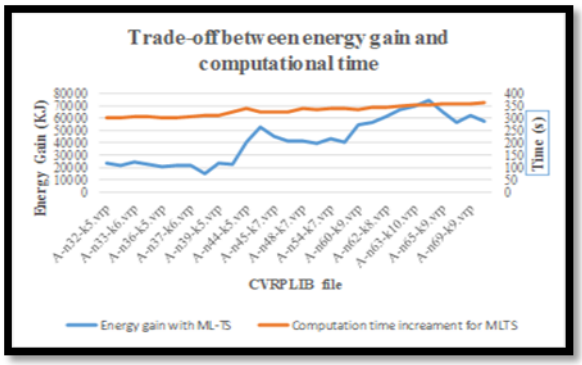


Figure 5. Tradeoff between computational time increment and elite energy improvement with TS and ML-TS

In order to identify that using ML has created a significant impact on the elite energy obtained, data obtained are tested using Freedman's non-parametric test. Here the two hypothesis are built and confidence level is selected as 95% and are summarized in table IV.

Table 4. Factors Considered In Comparison

Null hypothesis H_0	Means of elite energy from Generic TS-EMVRP and ML-TS-EMVRP are the same
Alternative hypothesis H_1	Means of elite energy from Generic TS-EMVRP and ML-TS-EMVRP are not the same.
Significance	0.05
Confidence level	95%
Reject criteria	Reject null hypothesis if p-value <= 0.05

Test statistics has been calculated and the results are depicted in Table V. Through the test statistics obtained p value is 0.000 which is less than the significance level, P value obtained = 0.000 < 0.05. According to the statistic, null hypothesis is rejected, H_0, "Means of elite energy from Generic TS-EMVRP and ML-TS-EMVRP are the same" is rejected accepting the alternative hypothesis. Therefore, it concludes that "Means of elite energy from Generic TS-EMVRP and ML-TS-EMVRP are not the same."

Table 5. Test Statistic Result

	N	Mean	Minimum	Maximum	Percentiles 25th	50th	75th
Generic TS	28	85454.86	28335.234	51201	128357	54597.50	83695.50	117509.00
ML-TS	28	43121.00	11571.090	28560	69578	33440.50	41726.50	51910.25

Post hoc analysis is necessary in order to analyze which algorithm generate means of lowest elite energy. Post hoc analysis is necessary in order to analysis which algorithm generate means of lowest elite energy and the results of box plot is represented in Figure 6.

Post of analysis clearly shows that there is a significant variance in means of elite energies obtained with Generic TS algorithm and ML-TS algorithm. ML-TS shows greater reduction in mean in elite energies and is best suit for EMVRP.

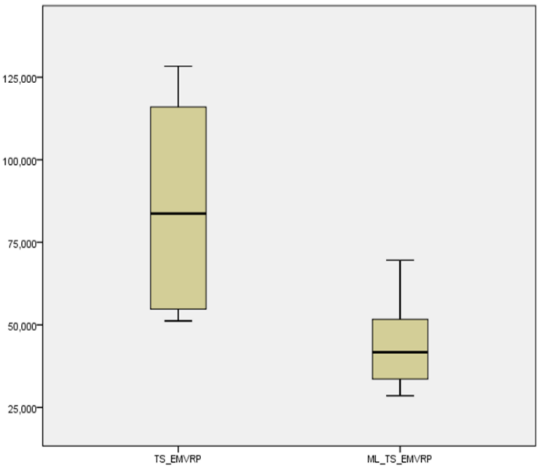


Figure 6. Box plot for mean comparison

5. Conclusion

Application of metaheuristic is a new research domain especially for the problem of EMVRP. According to the authors' knowledge there is only one significant study done on development of metaheuristic based on GA on EMVRP and tuning of parameters of the problem to arrive at practically satisfactory solutions for non-

polynomial problems. There are no studies reported on the development of TS based metaheuristic for the EMVRP and use of machine learning to guide the local search in formulating EMVRP.

This study yields a new TS- based algorithm for EMVRP and the solutions is further augmented with K-means clustering based machine learning algorithm which results in higher solution quality. The results clearly show that augmenting ML has significantly impact on the performance of metaheuristic on formulating EMVRP. The study proves the fact that augmentation with ML results in a significant increment in solution quality while the time taken to arrive at solution also increased. The study will open future research directions on using of metaheuristics in the domain of EMVRP and use of ML to enhance performance of the metaheuristics, which will eye open new research domain.

References

[1] Kara, Imdat, B.Y. Kara, and M.K. Yetis, "Energy minimizing vehicle routing problem," Combinatorial optimization and applications. Springer Berlin Heidelberg, 2007.

[2] K. Shraddha, M. Emmanuel, "Review and Comparative Study of Clustering Techniques," International Journal ofComputer Science and Information Technologies (IJCSIT), Vol. 5 (1) , 2014, 805-812/

[3] P.L.N.U. Cooray and T.D Rupasinghe, "Machine Leaning Augmented Energy Minimizing Vehicle Routing Problem (EMVRP)," International Journal of Industrial Engineering, Volume 2017 (2017), Article ID 3019523, 13 pages https://www.hindawi.com/journals/jie/aip/3019523.

[4] P.L.N.U. Cooray and T.D. Rupasinghe," An Analysis of Green Vehicle Routing Problem (GVRP) Solving Methodologies: A Systematic Review of Literature," Research for Transportation and Logistics Industry (R4TLI), Colombo, Sri Lanka, June 2016, www.slstl.lk/downloads/2016/R4TLI-B42.pdf.

[5] C. Blum, A. Roli, "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison," Journal of ACM Computer Surveys (CSUR) Volume 35 Issue 3, September 2003 pages 268-308.

[6] M. Thalei, "Heuristic for Energy Efficient Vehicle Routing Problem," Department of manufacturing engineering, Faculty of graduate school of Wichita state university.

[7] E. Volna, M. Kotyrba, "Unconventional Heuristics for Vehicle Routing Problems," Journal of Numerical Analysis, Industrial and Applied mathematics (JNAIAM), European Society of Computational Methods, Journal of Numerical Analysis, in Sciences and Engineering (ESCMSE) ,2015.

[8] F. Glover, "A user's guide to tabu search," Graduate School of Business, University of Colorado, Boulder, CO, USA, 1989.

[9] G. W. H. H. P. Rathnakumara, T. D. Rupasinghe, "Applicability of Metaheuristics for Energy Minimizing Vehicle Routing Problem (EMVRP)," Journal of Transport & Logistics, Issue 43/2017, July 2017.

[10] F. Glover, "A user's guide to tabu search," Graduate School of Business, University of Colorado, Boulder, CO, USA, 1989.

[11] A. Georgieva, C. Sin, "Solving Green Vehicle Routing Problem Using Tabu Search," Aarhus University, BSS. Department of Economics and Business Economics, 2015.

[12] R. Aringhieri, M. Bruglieri, F. Malucelli, M. Nonato, "A Special Vehicle Routing Problem Arising in the Optimization of Waste Disposal: A Real Case," Journal of Transportation Science, April 2017, https://doi.org /10.12 87/trsc.2016.0731

[13] S. Ropke, D. Pisinger, "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows." Journal of Transportation Science, Vol 40, Issue 4, August 2005, https://doi.org/10.1287/ trsc.1050. 0135

Predicting Floods in North Central Province of Sri Lanka using Machine Learning and Data Mining Methods

H. Thilakarathne[1], K. Premachandra[2]
[1]Department of Physical Science, Faculty of Applied Sciences, Rajarata University of Sri Lanka, Mihintale, Sri Lanka.
[2]Center for Micro-Photonics, Department of Physics and Astronomy, Swinburne University of Technology, Melbourne, Australia.

Abstract

The frequency of the occurrence of natural disasters at present has increased due to changes in global and regional climate. Hence, being able to forecast natural disasters has shown to be extremely useful in mitigating the loss of damages to the mankind. In this study, a hybrid model is developed for predicting the occurrence of floods in the North Central Province of Sri Lanka, using machine-learning techniques with artificial neural networks. The hybrid model developed combines two sub predictive models. First model predicts the future weather-related measurements using time series modeling. The second model, which is a binary classification machine-learning algorithm, predicts the probability of the occurrence of flood incidences in a future month using the forecasted weather values and historical flood data. The results show that all probabilities predicted by the hybrid model are a 91.7% match to the actual flood occurrences. Hence, this model could be adopted to predict flood occurrences of any region in Sri Lanka using historical weather and flood related data of the region of interest. The predictive model developed has been published as an Application Programmable Interface on Microsoft Azure cloud, illustrating the practical usage and feasibility of machine learning techniques in developing modern intelligent applications.

Keywords: Time Series Modelling, Binary Classification Machine-Learning Algorithm, Predict Flood Occurrences in Sri Lanka.

1. Introduction

Natural disasters such as floods, Tsunamis, drought, epidemic diseases, create major environmental economic and political issues, while causing great danger to human lives in Sri Lanka. Floods in Sri Lanka occur mainly during the monsoon season in which major floods in North Central province usually occur during the Northeast monsoons (December to February). Since preventing a natural disaster is not a practical task, predicting them has become extremely helpful for the policy makers and authorities to alert the public and take necessary precautions to avoid them.

In recent years, nonlinear modelling capability of Artificial Neural networks (ANNs) has become widely used in developing nonlinear predictive models for weather analysis [1]. ANNs, which is a subset of machine learning (past experience is used to optimize a performance criterion) [2], is a method that is inspired by the structure and functional characteristics of biological neural networks. ANNs are used in prediction, clustering, and classification (categorize observations). ANNs contain three main interconnected components: input neurons, output neurons and hidden neurons. Hidden neurons are the elements in between input and output neuron layers. The main feature of neural networks is the iterative learning process in which training data cases are presented to the network one at a time while adjusting the weights associated with the input values. After all cases are presented, the process often starts over again. During the learning stage, the network learns to predict the correct output label of training samples by adjusting the weights. A training dataset in machine learning is a set of data used to identify potential relationships between data. A test set is the set of data used to assess the effectiveness of a predictive relationship [3, 4].

ARIMA model is a statistical model that can be applied for the analysis of time series data. It is a generalisation of an autoregressive moving average model that is used to predict future points in a time series.

In this paper, a computational model using ANNs is developed to predict the probability of the flood incidents occur in the North Central Province of Sri Lanka by investigating the

underlying mechanisms of the flood occurrences. The ANNs in this study have been trained with Levenberg-Marquardt backpropagation algorithm, which is specially designed to solve non-linear least square complications [2]. In addition, ARIMA techniques used in the studies where the predictive models are developed are adopted to perform the time series predictions [5].

The forecasting models are developed in Microsoft Azure Machine Learning Studio, a cloud based machine-learning toolkit, which enabled to analyze and process a large volume of data efficiently [6].

2. Literature Review

Meteorological forecasting by means of numerical models dates back to the early 19th century when a mathematical approach for forecasting was proposed by Abbe [7] in the paper, *"The physical basis on long range weather forecasting"*. However, the numerical forecasting is not so accurate since the scientists lack the knowledge in simplifying complex atmospheric dynamics (occur due to variations of weather) into simple mathematical equations. Since then the nonlinear characteristics of weather data have kindled the interest in scientists to use nonlinear prediction mechanisms for both weather and disaster forecasting.

A. *Machine learning models for weather and disaster prediction*

With the evolution of computers, developing computational models for the purpose of weather forecasting became more accurate. In the paper on the origins of computer weather prediction and climate modeling, Lynch was able to show this by describing the evaluation of computer weather prediction and the methodologies used in the particular domain. He further describes the problems that are prevailing in the numerical weather prediction such as unavailability of linear simplified mathematical equations for weather and climate related parameters [8].

The study on "*Weather forecasting model using Artificial Neural Networks"* is carried out to determine the applicability of ANN approach for developing nonlinear predictive models in weather forecasting [1]. The advantages of having ANNs for weather forecasting over other forecasting methods are emphasized in this study. Since ANNs minimize the errors using various algorithms, the performance is improved in contrast to other models in the weather prediction domain. The tool used to carry out the analysis is Neural Network Fitting Tool, *nntool* available in MATLAB software. Artificial Feed-Forward Neural Network with back-propagation principles is selected as the training element.

B. *Time series predictions for weather forecasting*

Autoregressive Integrated Moving Average (ARIMA) techniques have been used to predict the behavioural pattern of weather attributes such as rainfall in the literature [5, 9, 10]. ARIMA is a statistical technique for modeling time series data [4]. To maximize the prediction accuracy of ARIMA models, model selection is performed over a time series in an automated fashion [5].

The numerous weather and disaster prediction studies provide an extensive base of information and approaches to the problem. These studies cover a wide spectrum of computational efforts from traditional statistical approaches such as the numerical weather prediction modeling, to machine learning and data mining methodologies [7, 8, 11].

In our study on building a model for predicting floods in North Central province of Sri Lanka using machine learning and data mining methods, artificial neural networks are used to adopt the non-linearity in predictive models. ARIMA models are adopted to perform the time series predictions within the model.

3. Methodology

A. *Data Collection*

Weather related historical data (January 1976 to December 2015) of Anuradhapura district in the North Central Province of Sri Lanka including monthly average rainfall data (mm), monthly average minimum and maximum temperature data (°C) have been collected from the Department of Meteorology of Sri Lanka (Table 1) [12]. Flood related historical data of the North Central Province of Sri Lanka is extracted from *DesInventar* – Disaster Information Management System of Sri Lanka [13]. The data includes 51 flood type disaster records from January 1976 to December 2015.

Table1. Statistical summary of the historical weather data obtained from the department of meteorology, Sri Lanka.

	Weather Attributes		
	Rainfall	**Minimum Temperature**	**Maximum Temperature**
Number of records	480	480	480
Mean	111.71mm	23.80°C	32.73°C
Median	73.25mm	24.10°C	33.05°C
Minimum Value	0mm	18.80°C	28.30°C
Maximum Value	683.9mm	26.10°C	37.50°C
Standard Deviation	120.78	1.38	1.80
Number of Unique Values	399	89	100

All data are converted into a monthly attribute and the missing values are substituted by the attribute's mean (average) of the specific field since it is independent of the sample mean [14].

B. Predictive modeling

In order to build the flood type disaster prediction model, two sub predictive models are developed in the study. While the first model involves predicting future values of the weather attributes (average monthly rainfall, average monthly maximum and minimum temperatures), the second model involves forecasting the flood type disaster incidents based on the input weather parameters. The output values of the first predictive component have been used as the input variables of the second predictive component.

The future weather attributes, which are the output of the first component of the flood prediction model, are forecasted using the ARIMA model. By defining historical data as a continuous time series, monthly weather data over a span of 39 years (January 1976 to December 2015) have been used as input variables in this component of the model. The values of weather attributes have been forecasted for the period of one year from January 2015 to December 2015. These predicted values are then compared to the actual weather data from January 2015 to December 2015 to evaluate the accuracy and the reliability of the developed time series model.

The model that predicts the probability of flood incidents is developed as the second component of the hybrid model. It is identified as a binary classification machine-learning problem, where the instances are classified into two classes [2].

The forecasted weather values are utilized as input data of this component of the model. The input data parameters are normalized to values between 0-1, using *minmax normalization* [15]. The output of the model is calculated as a probability, where the values approximate to 1 denoting a positive flood type disaster occurrence and 0, a negative (no) flood type disaster occurrence in the specific month to the future. The data flow diagram of the complete predictive model is shown in figure 1.

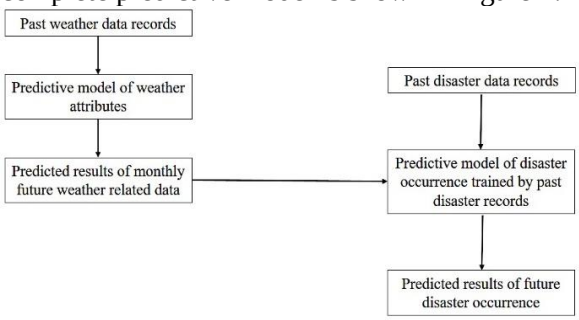


Figure 1: Data flow diagram of the complete predictive model

A binary classifier built with a neural network having a learning rate of 0.70, two hidden neurons and one hidden layer is chosen to build the predictive experiment. The optimal learning rate and the number of iterations are chosen using the '*Tune Model Hyperparameters*' module in Azure Machine Learning Studio [16]. This model uses different combinations of settings of the neural network in order to determine the optimum hyperparameters (parameters whose values are set prior to the commencement of the learning process) for the given prediction task and data [16]. The learning rate is a systematic approach to generalize the model when a large amount of training data is used. The classification model using the training dataset is cross-validated for reliability by dividing the dataset into 10 folds and calculating the classification accuracy for each fold [14].

3. Results

A. Output of the 1st predictive model

The predicted output values of rainfall, minimum temperature and maximum temperature of year 2015, show a slight deviation between the predicted and the actual values (Table 2, Figure 2,3,4). However, it is interesting to notice that both forecasted and actual values of all attributes follow the same distribution pattern (Figure 2,3,4).

The forecasted *accuracy measures* comparing predicted values to the observed values are obtained via calculating the root mean square error (RMSE) of each predicted attribute. The RMSE values of forecasted rainfall, minimum and maximum temperatures (115.58, 0.42 and 0.56) show a relatively high error rate compared to the forecasted minimum and maximum temperature values.

Table2. The actual values of the weather attributes in 2015 vs the predicted output of the first predictive model.

Month	Weather Attribute					
	Rainfall		Minimum Temperature		Maximum Temperature	
	Actual	Predicted	Actual	Predicted	Actual	Predicted
1	15.8	189.79	22.1	21.71	29.7	30.59
2	161.1	154.29	23	22.37	31.3	31.94
3	26	119.20	23.9	23.45	33.6	34.58
4	288.2	206.53	25	24.40	34	34.21
5	264.7	65.46	25.9	25.44	33.4	33.91
6	10.1	26.42	25.5	25.47	33.9	34.00
7	0	32.47	25.3	25.055	34.1	33.99
8	180.2	54.85	25.1	24.98	34.2	34.20
9	147.1	81.70	24.8	24.80	33.5	34.57
10	388.6	257.33	24.6	23.99	32.8	33.11
11	452.4	263.20	23.8	23.26	31.6	31.41
12	286.4	262.17	23	22.82	29.8	29.90

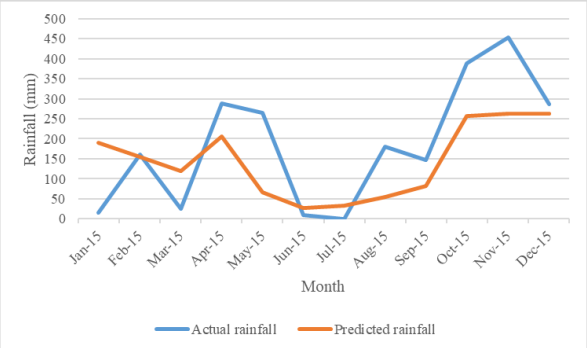


Figure 2: Deviation of actual rainfall data and predicted rainfall values for the year 2015

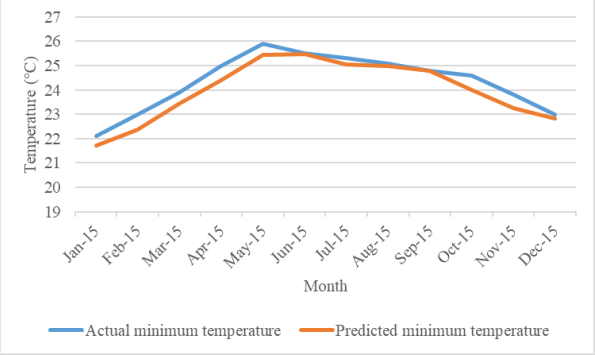


Figure 3: Deviation of actual minimum temperature data and predicted minimum temperature values for the year 2015

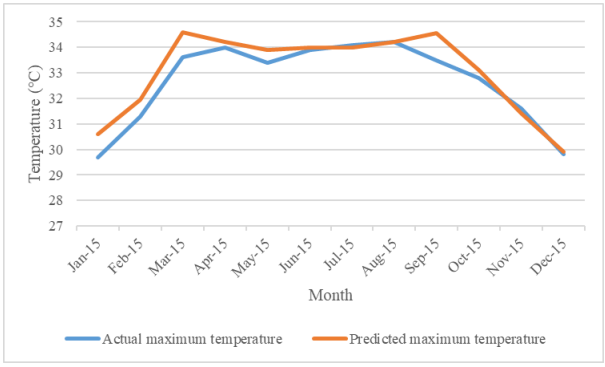


Figure 4: Deviation of actual maximum temperature data and predicted maximum temperature values for the Year 2015

B. Output of the 2nd predictive model

The classification model with 2 hidden neurons and 0.70 learning rate, shows that the model predicts the occurrence of floods with a mean accuracy of 0.917 and a 0.044 standard deviation (Table 3).

Table3. 10-fold cross validation output of the binary classification model.

Fold Number	Accuracy	Precision	Recall	AUC
0	0.854	1.000	0.364	0.958
1	0.917	0.500	0.250	0.747
2	0.875	0.800	0.444	0.923
3	0.958	1.000	0.714	0.951
4	0.958	1.000	0.600	0.944
5	0.979	0.500	1.000	0.979
6	0.917	0.000	0.000	0.815
7	0.917	0.333	0.333	0.911
8	0.938	0.500	0.333	0.889
9	0.854	0.000	0.000	0.948
Mean	0.917	0.563	0.404	0.907
Standard Deviation	0.044	0.384	0.308	0.073

The receiver operating characteristic (ROC) curve plotted between true positive rate and false positive rate (Figure 5) for this two-class classification problem is used to evaluate the accuracy of classification of the flood type disaster prediction model (In a ROC curve the true positive rate is plotted in relation to the false positive rate for different cut-off points of a parameter).

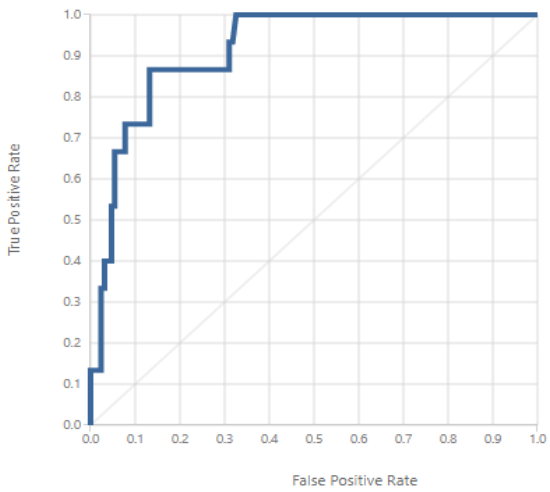


Figure 5: ROC curve of the flood type disaster prediction model

The predictions are yielded to a point in the upper left-hand corner of the ROC space denoting a high probability of a true positive output. The (0,1) point is called a '*perfect classification*', where no false positive or false negative outputs are predicted using the model. The true positive rate defines the number of correct positive results among all positive samples available during the test. False positive rate, on the other hand, defines the number of incorrect positive results among all negative samples available during the experiment [17]. The area under the curve (AUC) is 0.915 square units, showing that the probability of a randomly chosen positive instance is higher than a randomly chosen negative one.

A web application programmable interface (API) is implemented using the flood predictive model in order to obtain the predicted values of flood incidents in the North Central Province of Sri Lanka (https://goo.gl/faICPD).

4. Discussion

Natural disasters cause significant damages to the environment while causing a threat to human lives. The ability to predict disasters prior to occurring helps minimize the damages caused. This study proposes a predictive model with two sub predictive models that forecasts floods using historical weather and flood data. ANNs and statistical concepts of time series forecasting are utilized to develop the model.

RMSE values of time series predictions of weather data (average rainfall, average minimum temperature, and average maximum temperature) obtained from the first predictive model are 115.58, 0.42, and 0.56 respectively. Average minimum and maximum temperature values show very low RMSE values depicting a very small deviation between the forecasted and actual values. However, high RMSE value of average rainfall data shows a large deviation between the forecasted and actual values. High RMSE of predicted rainfall may have occurred due to outliers. These outliers could be due to the rain gauge measurement errors of the ground meteorology stations where the data is collected. In addition, sudden weather changes such as thunderstorms and hurricanes could have caused outliers in monthly average rainfall data. However, removing outliers from the training dataset could reduce RMSE values and hence, could obtain a better-forecasted output and a reliable predictive model.

Limiting the number of hidden neurons to two neurons in the second sub predictive model minimizes the probability of overfitting. Overfitting occurs when the model becomes an exact fit to the training dataset. This makes the model less generalised and leads to poor performance on new data [2].

The results of the predicted model indicate that the forecasted occurrence of flood incidents accurately corresponds to the testing dataset (Table 3). The binary classification that has been adopted for the model uses 0.5 as its threshold giving the optimal AUC.

Increasing the number of input parameters of the neural network increases the accuracy and the reliability of the predictions [1]. Hence, the model could be more reliable if weather data such as relative humidity, atmospheric pressure, wind speed and solar radiation are taken into consideration.

5. Conclusions

Being able to predict natural disasters has become an extremely important task to the policy makers and authorities since they cause significant damage to the environment and human lives. In this paper, a computational model using machine learning techniques with ANNs has been proposed to predict the occurrence of flood incidents in a particular month in the North Central Province of Sri Lanka.

This predictive model is built as a combination of two separate forecasting models. The first model forecasts the future values of weather attributes using time series forecasting methods (ARIMA). The second model predicts the probability of flood occurrences in a

particular month as a two-class classification machine-learning task.

The results of the predictive model show that it performs with a 91.7% accuracy, proving that this model can be used to predict the occurrence of flood incidents in any region in Sri Lanka using the historical data on weather and floods of the region of interest.

This study also proves that machine-learning models interconnected with data mining approaches are capable of performing environmental predictions based on historical data.

References

[1] K. Abhishek, P. M. Singh, S. Ghosh and A. Anand, "Weather forecasting model using Artificial Neural Network," Procedia Technology, vol. 4, pp. 311-318, 2012.

[2] A. Smola and S. Vishwanathan, Introduction to Machine Learning, Cambridge University Press, 2010.

[3] I. Bose and R. K. Mahapatra, "Business Data Mining - a machine learning perspective," Information & Management, pp. 211-225, 2001.

[4] I. Ibragimov, "Encyclopedia of Mathematics," 2 February 2011. [Online]. Available: http://www.encyclopediaofmath.org/index.php?title=Time_series&oldid=16499. [Accessed 14 August 2016].

[5] M. J. Kane, N. Price, M. Scotch and P. Rabinowitz, "Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks," BMC Bioinformatics, vol. 15, no. 276, 2014.

[6] J. Barnes, Azure Machine Learning, Redmond, Washington: Microsoft Press, 2015.

[7] C. Abbe, "The physical basis of long range weather forcasting," Mon. Weather Rev., vol. 29, pp. 551-561, 1901.

[8] P. Lynch, "The origins of computer weather prediction and climate modeling," Journal of Computational Physics , 2008.

[9] S. M. Sundaram and M. Lakshmi, "Rainfall Prediction using Seasonal Auto Regressive Integrated Moving Average model," Indian Journal of Research, vol. III, no. 4, pp. 58-60, 2014.

[10] D. K. Patrick, P. P. Edmond, T. M. Jean-Marie, E. E. Louis and K.-t. N. Ngbolua, "Prediction of rainfall using autoregressive intergrated moving average model: Case of Kinshasa city (Democratic Republic of the Congo), from the period of 1970 to 2009," Journal of Computation in Biosciences and Engineering, vol. 2, no. 1, 2014.

[11] D. P. Solomatine and Y. Xue, "M5 Model Trees and Neural Networks: Application to Flood Forecasting in the Upper Reach of the Huai River in China".

[12] "Department of Meteorology Sri Lanka," 2016. [Online]. Available: http://www.meteo.gov.lk. [Accessed 1 October 2017].

[13] "Disaster Information Management System - Sri Lanka," Disaster Management Centre (DMC),Ministry of Disaster Management, [Online]. Available: http://www.desinventar.lk. [Accessed 6 August 2016].

[14] J. Han, M. Kamber and J. Pei, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, 2012.

[15] G. K. Patro and K. K. sahu, "Normalization: A Preprocessing Stage," in arXiv preprint arXiv:1503.06462 , 2015.

[16] Microsoft, "Tune Model Hyperparameters," 21 July 2017. [Online]. Available: https://msdn.microsoft.com/en-us/library/azure/dn905810.aspx. [Accessed 15 September 2017].

[17] L. G. Michael Goebel, "A Survey of Data Mining and Knowledge Discovery Software Tools".

[18] M. S. G. A. A. Kumar Abhishek, "Weather forecasting model using Artificial Neural Network," Procedia Technology , pp. 311-318, 2012.

[19] P. Baldi and S. Brunak, Bioinformatics: The Machine Learning Approach, The MIT Press, 2001.

A Genetic Algorithmic Approach for Optimizing Supply Chain Network Designs for Retail Distribution Networks

I. U. Munasinghe[1], T. D. Rupasinghe[2]
[1,2]Department of Industrial Management,
Faculty of Science, University of Kelaniya, Sri Lanka
[1]inokaudayangi221@gmail.com, [2]thashika@kln.ac.lk

Abstract

The modern industries are compelled to become more competitive for providing the quality product at a minimum cost. In particular, companies have to analyze the efficiency of their logistic operations and need to adopt a holistic view of the entire supply chain network. The network design optimization attempt to optimize the flows of goods traverse among the main entities of the supply chain, not limited to the suppliers, warehouses, distribution centers, retailers and customers. In this study, Network Optimization (NO) will be carried from the perspective of the Retail Supply Chain (RSC) and study proposes a Genetic Algorithm (GA) for optimizing a novel mathematical model of the retail SCND. Most of the real world scenarios involve multi-source, multi-stage and multi-product with large instances. Therefore, the developed GA is more beneficial for real world scenarios strategic decisions such as location and allocation decisions and also to perform powerful 'what if' analyses. The authors have elaborated the GA with numerical examples and compared the results. The outcome of the study proposes the GA approach to reduce the distribution network design cost and the most appropriate number of facilities to minimize total supply chain costs to match the organization's service goals.

Keywords: *Genetic Algorithm (GA), Mixed Integer Linear Programming (MILP), Network Optimization (NO), Retail Distribution, Supply Chain Network Design (SCND), Supply Chain Performance*

1. Introduction

A world-class transformational supply chain begins with a network that employs an all-encompassing view of the various business areas that manage the delivery of products to customers. The result is significant capital, operational, and tax savings while achieving optimal customer satisfaction. The result is significant capital, operational, and tax savings while achieving optimal customer satisfaction.

As customer service requirements become more complex, supply chain optimization studies have become the foundation for some of the most successful companies' logistics and fulfillment operations. Thus, world-class organizations now realize that non-integrated manufacturing processes, non-integrated distribution processes and poor relationships with suppliers and customers are inadequate for their success [4]. The network design problem is one of the most comprehensive strategic decision problems that need to be addressed for a long-term efficient operation of the entire supply chain. The SCND provides an optimal platform for efficient and effective supply chain management [8] and also determines the number, location, capacity and type of plants, warehouses and DCs to be used [2].

Due to the complexity of Location Routing Problem (LRP), the exact methods have been used for solving small sized instances, and only for two-index formulations [6]. Three-index formulations, a more versatile but more complex, have not been solved optimally. Therefore, multi-stage formulation is realistic as well as a timely research which has a significant practical importance. Many of studies have focused only a two-stage supply chain network design with multi-product and/or multi-source scenarios. Therefore, in this study, authors consider the multiproduct, multi-source, multi-stage nature of the SCND. The multi-stage formulation has more advantages as opposed to the single SCND formulation. Furthermore, the proposed GA considered six stages on SCND and the Network Optimization (NO) will be carried from the perspective of Retail Supply Chains (RSC) and study proposes a Genetic Algorithm (GA) for optimizing the MILP. Most of the real world problems deal with large instances consisting multi-source, multi-stage and multi-products. Therefore, the developed model is more beneficial for real world strategic

decision making such as location and allocation decisions. The authors have developed the new GA to optimize the retail SCND and has validated the solutions with the CPLEX optimization, which was developed by [15].

2. Literature Review on SCND Optimization

The authors report a rigorous literature survey with the approaches gaps and applicability of existing literature related to the network design optimization of supply chain network. The review process is carried out by extending the published review literature from [12] and [10], which consists articles up to 2016.

Network design is the basis for the efficient operation of the supply chain and consequently one of the most important problems a supply chain manager is called to solve. During the past decades, there have been some interesting studies in the literature concerning the network design problem [12]. Review of the SCND problems are depicted in Table 1 with application areas and solving techniques.

As the review, most of the studies are in related to the production-distribution planning. A few of the studies focused with on the distribution network planning. The authors have reviewed the literature across the production-distribution and distribution network planning studies to enhance the understanding of problem formulation and applicability of the proposed model. There are two studies, which have concerned about supply chain network design optimization. One of the studies have solved by using empirical test cases, whereas the other one has solved by using CPLEX. The CPLEX approach cannot be used for large instances and only one study has used GA with a single source approach. Therefore, there is an open research area for the SCND optimization with GA for multi-source and multi products context.

Table 1. Summary Of The Literature Review

Study	Application Area	Solving Technique
[19]	Production- distribution, Facility location-allocation	Heuristic procedure with the aid of Lagrangian relaxation
[23]	SCND optimization, production-distribution planning	Lagrangian heuristic and GA
[1]	Capacitated location-allocation	MILP and Spanning tree based genetic algorithm (GA)
[22]	production–distribution model with consideration bills of materials (BOM)	MILP problem solved using LINDO
[24]	Production-distribution	Heuristics
[25]	Facility location-allocation problem	Non-linear integer program and propose a Lagrangian-relaxation solution algorithm
[2]	Facility location-allocation problem	Non-linear integer program and propose a GA
[3]	Production-distribution planning	MILP and heuristics
[13]	Transportation planning	Non-linear integer program and propose a spanning tree-based GA
[9]	design task involves the choice of facilities (plants and distribution centers (DCs)) to be opened and the distribution network design to satisfy the customer demand	GA and result compared with CPLEX, Lagrangian heuristic, hybrid genetic and simulated annealing algorithms
[16]	Facility location	Non-dominated sorting algorithm (NSGA II)
[18]	Distribution network planning	MIP and heuristics
[20]	Distribution allocation	A mathematical model is formulated as an integer-programming and the model is solved using GA-based heuristic
[17]	Production-distribution	MILP and particle swarm optimization heuristic algorithm
[10]	Production-distribution planning	Heuristics based on Tabu Search (TS)
[14]	Supply chain network design optimization	Empirical test cases
[15]	Supply chain network design optimization	Simulation and optimization based approach. Optimization has performed with CPLEX

3. Formulation of the GA

The Genetic Algorithm (GAs) using the Darwinian concept was first introduced by Holland (1975). GA is useful when a large search space with little knowledge of the solution is presented, and has been successful in providing near optimal solution for many diverse and difficult problems [10]. Overall GA procedure is depicted in Figure 1.

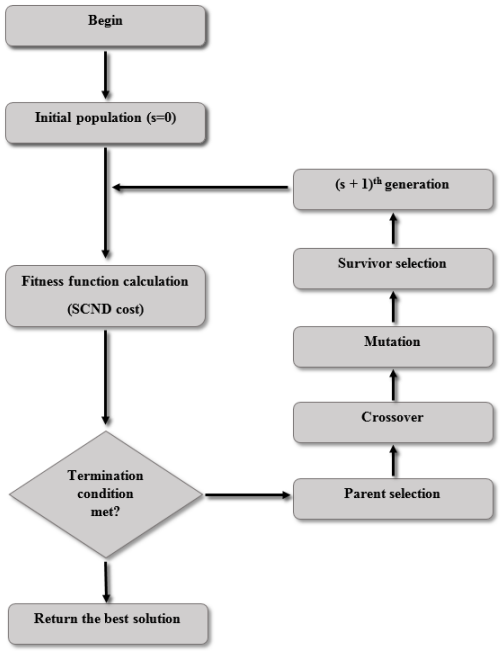


Figure1. Overall GA Procedure

A. *Chromosome Representation*

Literature suggests to use the flexible multi logistics network (fMLN) with vertex based encoding GA solution to find the best route for delivering the product to each customer when the network is single source [11]. In our approach to depict the multi product, multi-source SCND nature. The chromosome is represented as an edge based. The formulation has a SCND with six different delivery routes along with multiple products as supplier to warehouse, warehouse to DC, DC to retailer, supplier to DC, supplier to retailer and warehouse to retailer. Therefore, the length of every chromosome is equal to:

$$X \times (I(J + K + L) + J(K + L) + K(L)) \quad (1)$$

Where;
X: Number of Products
I: Number of Suppliers
J: Number of Warehouses
K: Number of DCs
L: Number of Retailers

There are six different types of decision variables as $Q1_{xij}$, $Q2_{xjk}$, $Q3_{xkl}$, $Q4_{xil}$, $Q5_{xik}$ and $Q6_{xjl}$ and they are depended on the number of suppliers, warehouses, DCs, retailers, and products. Every gene represents the product amount shipped between different facilities. Figure 2 is depicted the proposed chromosome representation to solve the multi-product, multi-source SCND problem.

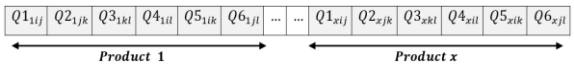


Figure 2. Chromosome Representation

B. *Initialization*

A random chromosome generation has been implemented to generate the initial population and the fitness of each chromosome is equal to the corresponding SCND cost, which is obtained by multiplying shipping quantity, unit shipping cost and distance. The first generated population is improved by applying crossover and mutation genetic operators.

The value for each gene is generated randomly within a defined and specific range. In this study, this specific range is defined by generating random value in between 0 and upper limit value. This upper limit value is defined by considering retailers demand values. Because, each SCND should be satisfied the retailer's demand.

The fitness of each chromosome equal to the corresponding SCND cost. Developed GA fitness function calculation is based on the model developed by [14] where the total supply chain network design cost is minimized. The overview of the proposed GA is depicted in Figure 3.

C. *Genetic Operators*

In order to enhance the ability of producing new solutions, this study is employed the cross over and mutation operators. Due to the different natures of the different products in the chromosome, crossover and mutation operations are performed in separately with each product.

```
Procedure: GA for SCND optimization
Input: Problem data
    initialize P(g) // population of chromosomes
Output: Best Solution
    for g = 0 to G // g generation number
        GA capacitated constrained checking
        fitness evaluate P(g) // evaluation function is objective function
            while (termination criteria is reached) do
                select C1 and C2 by tournament selection from P(g)
                apply cross over to obtain C
                apply mutation to C
                // generate new offsprings
                fitness evaluation of C
                 select P(g+1) from P(g) and C by romoving the worst solution
                g ← g+1
            end while
    output: best solution
    end for
```

Figure3. Proposed GA

1) Crossover

This operator allows genetic material to be exchanged between two chromosomes with the aim of exploring new solution spaces [5]. Cross over operates on two parent chromosomes at a time.

It generates offspring by adding both chromosomes' features. A sample illustration of cross over operator is depicted in Figure 4, Figure 5 and this sample is illustrated for one supplier, one warehouse, one DC and one retailer for multi-product.

Step 1: generate a random split point from each product to cross over. (Figure 4)

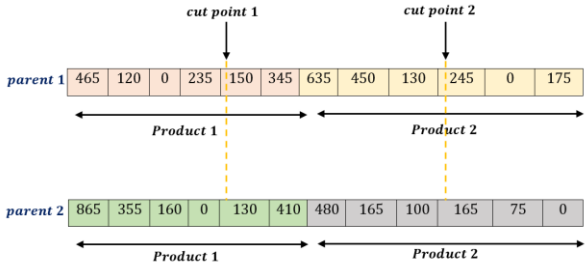


Figure 4. Step 1 Overview

Step 2: Exchange substrings between parents and generate new offspring. (Figure 5)

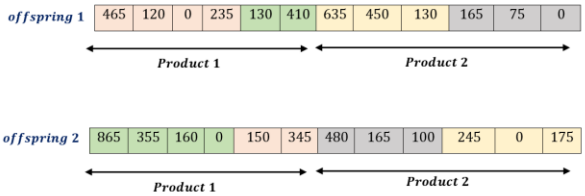


Figure5. Step 2 Overview

1) Mutation

This operator produces spontaneous random changes in one or more chromosomes of a population generation selected on a fitness basis [5]. In the proposed GA, a gene swapping mutation operator is executed. It randomly selects two genes from each product within a chromosome and exchange their positions each other. A sample illustration of mutation operator is depicted in Figure 6.

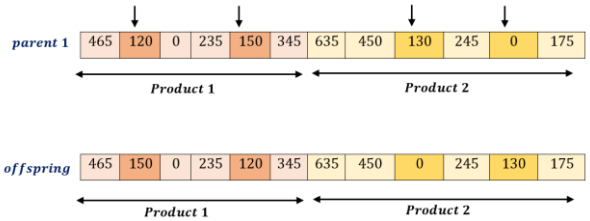


Figure 6. Mutation Overview

In this study, GA was developed to solve the SCND problem, which was formulated by [14], and performance comparison was carried through GA and CPLEX solution computed previously by the authors [15] lot of diversification in the solutions. Consequently, it is better to use an evolutionary algorithm like GA in this phase [7].

4. Discussions and Findings

The proposed GA chromosome representation, initialization, crossover and mutation have been explained in previous section. The algorithms were coded in C#.Net language and the parameters have been tuned for the proposed GA. Results of the proposed GA is depicted in Table 2.

A. Performance Comparison of Developed GA

With an approximation method like GA, it is not ensured the optimum solution is found.

Therefore, the solution obtained with the GA is compared with those obtained with CPLEX [15].

According to the results obtained in Table 2, it can be seen that as the number of generations increase, the solution quality enhances. From the GA, the minimum value is reached when the GA parameter of the generation size is fixed to five. In Figures 7 and 8 show this phenomenon.

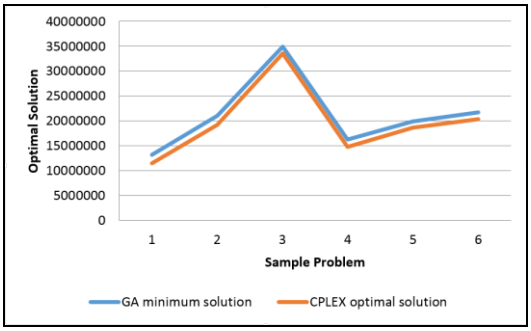


Figure 7. Performance comparison for first six sample problems

Table 1. GA solutions for sample problems

Sample Problem	Parameters tuning with varying size of the Generation size in the GA				
	1	2	3	4	5
01	13, 636, 420	13, 356, 785	13, 347, 360	13, 269, 780	13, 158, 690
02	21, 436, 450	21, 342, 830	21, 198, 760	21, 136, 960	21, 065, 490
03	35, 087, 960	34, 969, 520	34, 723, 695	34, 725, 875	34, 926, 650
04	16, 526, 555	16, 465, 452	16, 136, 450	16, 136, 985	16, 236, 895
05	20, 325, 645	20, 136, 540	20, 136, 525	19, 888, 995	19, 865, 750
06	22, 236, 450	22, 036, 980	22, 036, 475	21, 936, 210	21, 723, 920
07	17, 656, 980	17, 536, 565	17, 523, 325	17, 223, 680	17, 232, 640
08	27, 269, 545	27, 220, 075	27, 034, 895	26, 958, 750	26, 932, 125
09	49, 923, 695	49, 823, 680	49, 767, 855	49, 775, 660	49, 678, 930
10	28, 136, 450	27, 936, 495	27, 648, 750	27, 769, 840	27, 469, 780
11	34, 526, 985	34, 526, 970	34, 264, 450	33, 969, 785	33, 865, 210
12	36, 598, 750	36, 572, 655	36, 498, 760	36, 149, 320	36, 126, 695

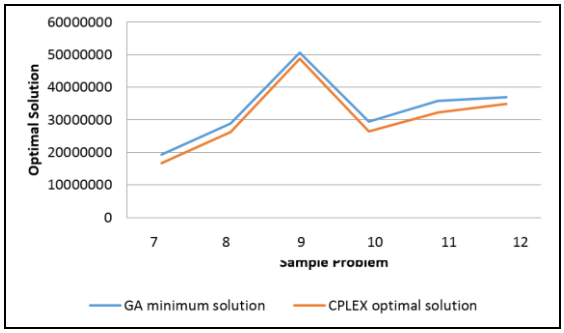


Figure 8. Performance comparison for last six sample problems

Since most of the real world scenarios are large cases, the model should be facilitated to reach optimal solution within a reasonable timeframe. To evaluate the accuracy, the authors have calculated a gap calculation between CPLEX and GA solutions for above 12 sample problems within 10 generations. Gap calculation are depicted in Table 3 and 4. It is facilitated to enhance the accuracy and fine function of the proposed model with GA.

$$\text{Gap} = \frac{\text{GA Minimum Solution} - \text{CPLEX Optimal Solution}}{\text{CPLEX Optimal Solution}} * 100 \quad (2)$$

This study has used the GA and CPLEX together for validation of the proposed approach using different scenarios and compared the outputs to demonstrate the quality of the performance [15]. By using the gap calculation, the authors concluded that the newly proposed GA is more suited for large-scale real world scenarios to optimize the SCND with minimum distribution cost.

6. Conclusion

In this study, initially a thorough systematic review has been carried out on SCND model approaches and problem formulations. Subsequently, the authors have considered the new MILP model for retail SCND by taking into account several factors such as transportation cost, capacity allocation and optimal facility location based on the study of [14]. The novelty of this study is carried out by considering distance from facility to facility with SCND and multi-echelon approach with new proposed GA. Therefore, the proposed GA is a new approach, which covers multi-product, multi-stage and multi-source scenarios together.

The solutions obtained have been compared solution with CPLEX optimal solution, which was obtained by [15]. Finally, the proposed GA is a more suitable approach for solving the SCND real world scenarios.

Furthermore, this study provides solutions for determining; the most profitable combination of supplier-warehouse-DC-retail mapping for product flows, locations of Distribution Centers (DCs) and, warehouses and locations for setting up new warehouse/DC, and capacities of DCs and warehouses.

Table 3. Gap Calculation for Sample Problems 01-06

Sample Problem	01	02	03	04	05	06
GA minimum solution	11, 708, 690	19, 465, 490	33, 626, 650	15, 036, 895	18, 865, 750	20, 543, 920
CPLEX optimal solution	11, 520, 880	19, 248, 380	33, 589, 850	14, 755, 220	18, 655, 860	20, 331, 350
Gap (%)	1.63	1.12	0.10	1.90	1.12	1.04

Table4. Gap Calculation for Sample Problems 07-12

Sample Problem	07	08	09	10	11	12
GA minimum solution	16, 932, 640	26, 332, 125	49, 078, 930	26, 669, 780	32, 565, 210	35, 326, 695
CPLEX optimal solution	16, 713, 430	26, 237, 330	48, 681, 050	26, 362, 970	32, 202, 610	34, 942, 930
Gap (%)	1.31	0.36	0.81	1.16	1.12	1.09

References

[1] Admi Syarif, YoungSu Yun, Mitsuo Gen, Study on multi-stage logistic chain network: a spanning tree-based genetic algorithm approach, *Computers and Industrial Engineering*, vol. 43, pp. 299-314, 2002.

[2] Altiparmak, F., Gen, M., Lin, L. and Paksoy, T., A genetic algorithm approach for multi-objective optimization of supply chain networks, *Computers & Industrial Engineering*, vol. 51, pp. 196-215, 2006

[3] Burcu B. Keskin and Halit Üster, "Meta-heuristic approaches with memory and evolution for a multi-product production/distribution system design problem," *European Journal of Operational Research*, vol. 182, no. 2, pp. 663–682, Oct. 2007

[4] Chandrasekaran Sowmya Danalakshmi and Gabriel Mohan Kumar, Optimization of Supply Chain Network Using Genetic Algorithm, *Manufacturing and Industrial Engineering*, Issue 3,pp. 30-35, 2010

[5] Costa, A., Celano, G., Fichera, S. and Trovato, E., A new efficient encoding/decoding procedure for the design of a supply chain network with genetic algorithms, *Computers and Industrial Engineering*, vol. 59, no. 4, pp. 986-999, 2010

[6] D. Ambrosino and M. Grazia Scutellà, "Distribution network design: New problems and related models," *European Journal of Operational Research*, vol. 165, no. 3, pp. 610–624, Sep. 2005

[7] Esmaeil Mehdizadeh and Fariborz Afrabandpei, Design of a Mathematical Model for Logistic Network in a Multi-Stage Multi-Product Supply Chain Network and Developing a Metaheuristic Algorithm, *Journal of Optimization in Industrial Engineering*, vol. 10, pp. 35-43, 2012

[8] F. Varasteh, "Using Genetic Algorithm for Supply Chain Network Optimization," pp. 01–06, 2007. Available.https://www.researchgate.net/publication/228551885_Using_Genetic_Algorithm_for_Supply_Chain_Network_Optimization

[9] Fulya Altiparmak, Mitsuo Gen, Lin Lin, Ismail Karaoglan, A steady-state genetic algorithm for multi-product supply chain network design, *Computers & Industrial Engineering*, vol. 56, pp. 521-537, 2009

[10] Gabriel Machado Braido, Denis Borenstein, Gilmar D'Agostini Oliveira Casalinho, Supply chain network optimization using a Tabu Search based heuristic, São Carlos, vol. 23, no. 1, pp. 03–17, 2016

[11] Gen, M., Cheng, R. and Lin, L., Network Models and Optimization: Multiobjective Genetic Algorithm Approach, Springer-Verlag London Limited, 2008

[12] Kostis Taxakis and Chrissoleon Papadopoulos, A design model and a production–distribution and inventory planning model in multi-product supply chain networks, *International Journal of Production Research*, http://dx.doi.org/10.1080/00207543.2016.1158882, 2016

[13] Ming-Jong Yao and Hsin-Wei Hsu, A new spanning tree-based genetic algorithm for the design of multi-stage supply chain networks with nonlinear transportation costs, *Optim Eng*, vol. 10, pp. 219-237, 2009

[14] Munasinghe, I. and Rupasinghe, T. , A supply chain network design optimization model from the perspective of a retail distribution supply chain. In: *Manufacturing & Industrial Engineering Symposium* (MIES). Colombo: IEEE, 2016

[15] Munasinghe, I. & Rupasinghe, T., Simulation and Optimization-based Approaches to Assess the Behaviour of the Retail Supply Chain Network Designs. Colombo, *TIIKM*, pp. 17-29. , 2017

[16] Ranjan Bhattacharya and Susmita Bandyopadhyay, Solving conflicting bi-objective facility location problem by NSGA II evolutionary algorithm, *International Journal Advance Manufacturing Technology*, vol. 51, pp. 397-414, 2010

[17] Sasan Khalifehzadeh, Mehdi Seifbarghy, Bahman Naderi, A four-echelon supply chain network design with shortage:Mathematical modeling and solution methods, *Journal of Manufacturing Systems*, vol. 35, pp. 164-175, 2015

[18] V. Golmohammadi, H. Afshari, A. Hasanzadeh, and M. Rahimi, A heuristic approach for designing a distribution network in a supply chain system, *African Journal of Business Management*, vol. 4, no. 3, pp. 308–311, 2010

[19] Vaidyanathan Jayaraman and Hasan Pirkul, Planning and coordination of production and distribution facilities for multiple commodities, European *Journal of Operational Research*, vol. 133, pp. 394-408, 2001

[20] Vinay V. Panicker, R. Sridharan, B. Ebenezer, Three-stage supply chain allocation with fixed cost, *Journal of manufacturing technology management*, vol. 23, pp. 853-868, 2012

[21] Ronald G. Askin, Ilaria Baffo, Mingjun Xia, Multi-commodity warehouse location and distribution planning with inventory consideration, *International Journal of Production Research*, vol. 52, pp. 1897-1910, 2014

[22] Yan, H., Yu, Z., Cheng, E.T.C., A strategic model for supply chain design with logical constraints: Formulation and solution, *Computers & Operations Research*, vol. 30(14), pp. 2135–2155, 2003.

[23] Yang-Ja Jang, Seong-Yong Jang, Byung-Mann Chang, Jinwoo park, A combined model of network design and production/distribution planning for a supply network, *Computers and Industrial Engineering*, vol. 43, pp. 263-281,2002

[24] Y.B. Park, An integrated approach for production and distribution planning in supply chain management, *International Journal of Production Research*, vol. 43, pp. 1205–1224, 2005

[25] Zuo-Jun Max Shen, A multi-commodity supply chain design problem, IIE, vol. 37, pp. 753-762, 2005 http://dx.doi.org/10.1080/07408170590961120

Knowledge Discovery with Data Mining for Predicting Students' Success Factors in Tertiary Education System in Sri Lanka

K. T. S. Kasthuriarachchi[1], S. R. Liyanage[2]
[1]Faculty of Graduate Studies, University of Kelaniya, Dalugama, Sri Lanka
[2] Faculty of Computing and Technology, University of Kelaniya, Dalugama, Sri Lanka.

Abstract

Knowledge discovery in educational data would be so basic to determine better expectations on the undergraduates. Distinguishing proof of the components influence to the execution of undergraduates in light of various attributes will be supportive for instructors, educators and managers viewpoints. This paper endeavors to utilize different data mining ways to deal with find forecast manages in undergraduates' data to distinguish the components influence to the scholarly accomplishment in their tertiary education. The approach of this exploration observed the aftereffects of three mining algorithms with about 3800 undergraduates' records and the calculation which demonstrated the most elevated exactness has chosen as the best model and the connections acquired through that were gotten to foresee various elements against the objective of whether they will get the degree or not following three years of the university life. Naïve Bayes, Decision Tree and Support Vector Machine were used in predicting the most affecting factors to the performance of students. According to the prediction accuracy levels, the results of Decision Tree were selected since it outperforms the rest for the selected data set. Finally, the results were evaluated using a correlation analysis to select the most prominent factor. According to the test, the age, past failure modules, performance of past semesters were selected as the most influencing factors to the success or failure of the students in tertiary education system in Sri Lanka.

Keywords: *Educational Data Mining, Algorithms, Knowledge Discovery, Feature Extraction, Validation*

1. Introduction

Education is a key zone for accomplishing a long haul financial advance. Amid the most recent decades, innovative education in Sri Lankan tertiary education framework has enhanced with the blossom of mechanical establishments, which drives the undergraduates towards the degree level. In Sri Lanka, the tertiary education is proving by both state universities and degree awarding institutes. The University Grant Commission (UGC) was established to allocate funds to the universities and university institutes, serves as the central admission agency for undergraduate studies in universities, planning and monitoring of academic activities of the university system in order to maintain academic standards and implement national policies with respect to university education in Sri Lanka. The UGC selects students for admission to undergraduate courses based on the z-score [1]. The degree awarding institutes are recruiting students based on the performance of the selection test that they conduct and never worried about z-score. However, the majority of the students who enroll to institutes for their degree are having the pass mark to the Advanced Level examination, which is the minimum criteria for the university education. However, the statistics grabbed from institutes have proven that, only few students are completing the degree due to various reasons.

Data Mining (DM) / Business Intelligence (BI) which are very interesting methods used to predict what are the main reasons for student' failures?, what are the factors which affect mostly to the success of academic activities? and many more. Educational Data Mining (EDM) is a creating region of data mining, which the use of data mining strategies to a particular kind of data that chose from educational airs to address vital educational inquiries. There is a boundless uses of EDM to examine the conduct of data gathered from infant education, primary education, secondary education, higher education, auxiliary training, advanced education and option instruction have a place with conventional training conditions. Computer based educational condition incorporates Learning Management Systems (LMS), Intelligent Tutoring Systems (ITS), test and quiz frameworks, recreations and wikis discussions which use for e-learning and e-

mentoring, online guideline frameworks and so forth. [2].

Predicting the factors, which affect to student performance, is very important for educators, educational decision makers and even for students. It would be a great benefit to educators and course designers to identify the characteristics of the students who have to be enrolled to the degree program if, a quality educational outcome is expecting. Students also be able to get away from the negative criteria to reach to the maximum level of their education by these predicted outputs.

This research study is focusing to identify the factors, which affect to student performance in their tertiary education. The paper is organized as follows. First, discusses the background of EDM. Then, the materials and methods used to mine educational data to prediction are described. Then, the results of the analysis is discussed.

2. Background

Knowledge discovery and data mining can be thought of as apparatuses for basic leadership and authoritative adequacy. It has its root in machine learning, computerized reasoning, software engineering, and measurements. There are different data mining strategies and methodologies, for example, classification, cluster and association [3]. Each of these methodologies can be utilized to quantitatively examine huge data sets to discover shrouded significance and examples. While data mining has been connected in an assortment of enterprises, government, military, retail, banking, telecommunication and healthcare services, data mining has not gotten much consideration in educational setting. Educational data mining is a field of concentrate that examinations and applies data mining to take care of educational related issues.

Educational data mining characterized scholastic examination as the utilization of factual methods and data mining in ways that assistance workforce and guides turn out to be more proactive in recognizing at-risk students and comparing in like manner. It would fairly supportive in enhancing understudy maintenance, advancement of recommender framework, overseeing new courses and so forth. Data mining focuses on identifying the useful insights of improving student success and processes directly related to student learning, build the patterns of student behaviour, assist faculty members of institutes to enhance learning and supporting educational processors [4].

In effect, several studies were conducted in similar area of predicting students' performance. Students' grades were predicted by feed forward neural networks and back propagation algorithms [5]. Feature examination from logged data in a web-based system has applied by another study for the prediction, monitoring and evaluation of students' performance [6]. Prediction of university students' satisfaction was another research that has been done using regression and decision tree algorithms [7]. Another set of researches were used Naïve Bayes algorithm to predict the performance of the students [8] and different rule based systems were used to predict the performance in an e-learning environment [9]. Different regression techniques were used to predict students' marks in an open university using locally weighed linear regression, linear regression, model trees, neural networks, and support vector machines and for predicting high school students' likelihoods of success in university was another study conducted in educational data mining domain [10]. There was another study about a student performance recommender system, which was developed to provide recommendations to the students during the enrolment period using Decision tree, Naïve Bayes, Neural Networks and K-nearest neighbor algorithms [11].

In this research, Naïve Bayes (NB), Decision Tree (DT) algorithm and Support Vector Machine (SVM) were used as mining approached to identify the most correlated factors with the final year performance.

3. Materials and Methods

A. Sample

This study will consider the data collected by querying the students' database of an institute, which offers Information Technology (IT) degree in three years. The dataset consists of 3794 instances with 10 attributes as shown in table 1.

Table 1. Description of Data Set

Attribute	Details
Sex	Gender (binary: Male "M", Female "F"
Age	Age (numeric: from 18 – 45)
Failure	Has past failure modules (binary: yes, no)
S1	GPA of semester 1 (nuemeric: from 0-4)
S2	GPA of semester 2 (nuemeric: from 0-4)
S3	GPA of semester 3 (nuemeric: from 0-4)
S4	GPA of semester 4 (nuemeric: from 0-4)
S5	GPA of semester 5 (nuemeric: from 0-4)
S6	GPA of semester 6 (nuemeric: from 0-4)
IsPassDegree	Is the student pass the degree (binary: yes, no)

Gender and age are the demographic facts and all the others are academic facts related to the selected sample. During the pre-processing stage, all discriminative values were identified. All missing values, incomplete values were handled using, median imputation algorithm. Case deletion has been done to simply, removing some tuples from the data set.

Following histogram in Figure 1 describes the behaviour of our target variable.

According to the above figure1, 65% of students have successfully completed their degree while the remaining 35% have failed to complete the degree.

In prediction, the popular methods that are used by researches are classification, regression, and density estimation.

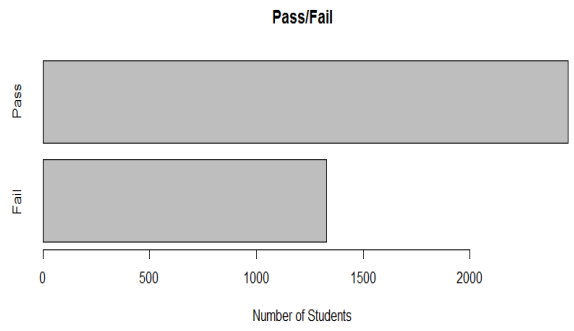


Figure 1. Histogram for the students' Pass/ Fail state

B. *Data Mining Algorithms*

Three classification type data mining methods were applied in this study. They are;

i) Naïve Bayes algorithm
ii) C5.0 Decision Tree algorithm
iii) Support Vector Machine algorithm

Naïve Bayes algorithm is a classification method based on supervised learning approach. This Classification is named after Thomas Bayes based on his Bayes Theorem. Bayesian classification gives practical learning algorithms which prior knowledge and observed data can be consolidated. It gives a valuable point of view to comprehension and assessing many learning algorithms. Naive Bayes Algorithm is a quick, exceptionally versatile algorithm. It can be utilized for Binary and Multiclass classification. It gives distinctive sorts of Naive Bayes Algorithms like Gaussian Naïve Bayes, Multinomial Naïve Bayes and Bernoulli Naïve Bayes. It is a basic algorithm that relies upon doing a group of checks, which can be effectively, prepare on little data set.

The enhanced C5.0 decision tree algorithm was selected to obtain a better analysis result to produce a smaller decision tree with less memory in less time with the support of boosting, weighting and winnowing.

Support vector machine (SVM) is a supervised learning methods used for classification, regression and outlier detection tasks in data mining. The analysis could be able to perform using Linear or Gaussian methods.

C. *Mining Environment*

The results of this experiment were gathered using R studio statistical software package, which runs on windows platform. R is a free and high-level matrix programming language with an intense tools for statistical and data analysis [12]. It consists of several helpful packages/ libraries to perform almost all mining tasks including Naïve Bayes C5.0 Decision Tree analysis and Support Vector Machine.

The sex/ gender of the student has converted to a numerical value at the time of executing the algorithm and '1' implies 'male' and '0' implies 'female'.

The data set has inputted to classification algorithms to categorize them. The input data set is comprising of vector values of attributes with equivalent classes. In the analysis, the data set has segregated into training sets and testing sets. Training dataset is to train the model and model learn from this. Testing data set is to measure how much the model has learnt using the train data.

3. Experiment Results

In the start of this investigation the impact of input attributes have been broken down in order to think of a better prediction result. For the identification of importance of every attribute,

Info Gain and Gain Ratio were tried utilizing R package. According to the results of impact test, the greater part of the attributes in the dataset as in table 1 were included to the further analysis.

The results of the mining algorithms were assessed to think of a better classification model. For this assessment of classification accuracy, 5 fold cross validation technique has been utilized. The cross validation has repeated several times and in every time utilize one sub set as a test set. Under the cross validation; prediction accuracy, Kappa statistics, Precision, Recall and F-measure could be recorded. The performance of the selected algorithms were measured as shown in table 2.

Table 2. Performance of classification methods

Criteria	**Algorithm**		
	Naïve Bayes	Decision Tree (C5.0)	Support Vector Machine
Correctly classified instances	1153	1220	1149
Incorrectly classified instances	111	44	115
Prediction Accuracy	92.17%	97.1%	92.05%
Kappa Statistic	81.91%	93.57%	80.58%
Precision	89.403%	94.85%	89.24%
Recall	99.75%	98.71%	98.86%
F-Measure	87.5%	95.79%	86.95%

As per the above outcomes shown in table 2, it was watched that all the classification algorithms delivered moderately great outcomes, which were more like each other. However, the highest result is obtained by Decision Tree classification.

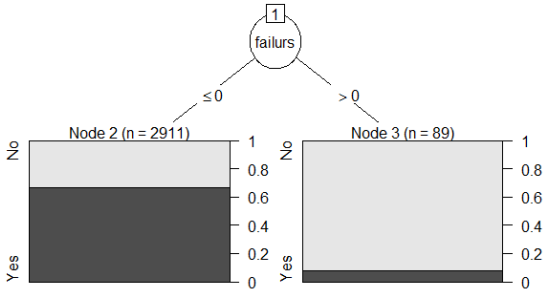


Figure 2. Decision Tree for fail modules and pass/fail status

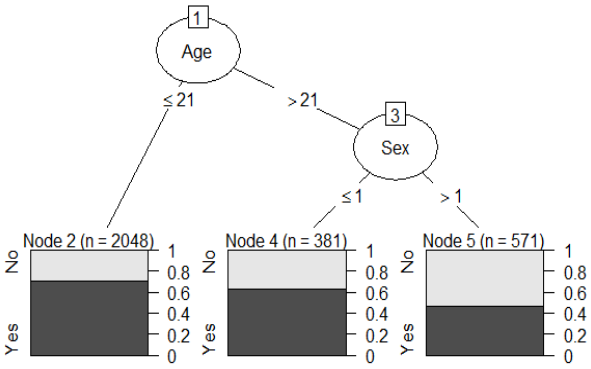


Figure 3. Decision Tree for age, sex and pass/fail status

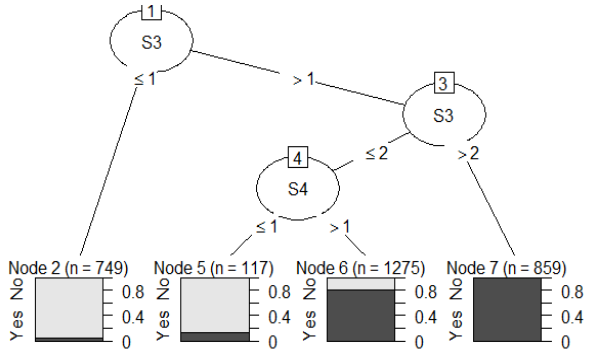


Figure 4. Decision Tree for semester 1, semester 2 performance and pass/fail status

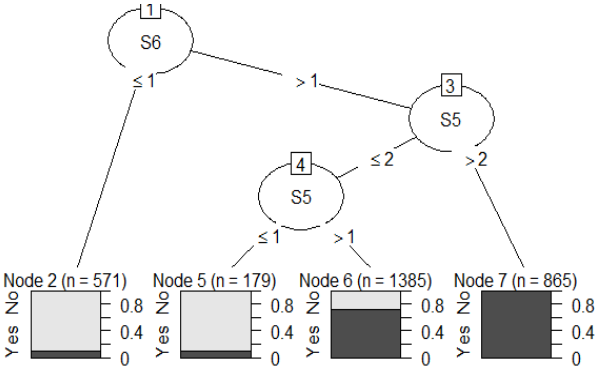


Figure 5. Decision Tree for semester 3, semester 4 performance and pass/fail status

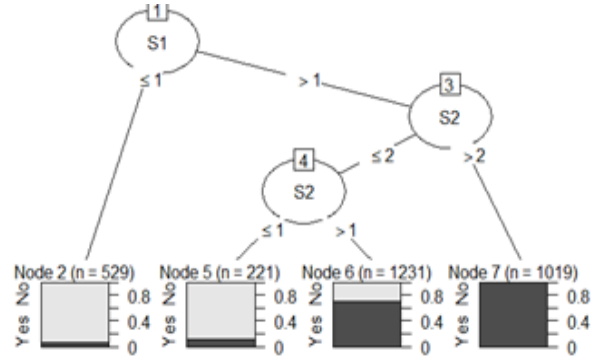


Figure 6. Decision Tree for semester 5, semester 6 performance and pass/fail status

According to the Decision Tree generated by R package, following interesting rules could be extracted.

Rule 1: if ! failure modules then pass

Rule 2: if age <= 21 then pass

Rule 3: if age > 21 ^ sex = "Female" then pass

Rule 4: if semester 1 GPA > 1 ^ semester 2 GPA > 2 then pass

Rule 5: if semester 3 GPA > 1 ^ semester 4 GPA > 2 then pass

Rule 6: if semester 5 GPA > 1 ^ semester 6 GPA > 2 then pass

Figure 7. Decision Tree Rules

The relationships discovered by the decision tree algorithms were evaluated using correlation analysis. The results of the correlation analysis was shown in table 3.

According to the results shown in above table, failure modules and pass/ fail status are significantly correlated with a correlation coefficient of -0.15 and p-value of 2.2e-16 which consists of a negative relationship among the variables.

Table 3. Results of Correlation Analysis

Variable	P-value	Correlation
Failure modules	2.2e-16	-0.1500
Age	1.82e-13	-0.119235
Semester1 GPA	2.2e-16	0.3429429
Semester2 GPA	2.2e-16	0.3705321
Semester3 GPA	2.2e-16	0.3901625
Semester4 GPA	2.2e-16	0.3191271
Semester5 GPA	2.2e-16	0.3393523
Semester6 GPA	2.2e-16	0.3160398

Age of the students and pass/ fail status of them are significantly correlated with a correlation coefficient of -0.12 and p-value of 1.82e-13 16 which consists of a negative relationship among the variables.

GPA of semester 1 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.34 and p-value of 2.2e-16.

GPA of semester 2 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.37 and p-value of 2.2e-16.

GPA of semester 3 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.39 and p-value of 2.2e-16.

GPA of semester 4 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.31 and p-value of 2.2e-16.

GPA of semester 5 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.33 and p-value of 2.2e-16.

GPA of semester 6 and pass/ fail status of them are significantly correlated with a correlation coefficient of 0.31 and p-value of 2.2e-16.

Based on the correlation results it was confirmed that the above decisions derived using Decision Tree algorithm were valid.

Therefore, the results were depicting that on the off chance that the undergraduates are not having past failure modules, clearly they will pass the degree. As the rule 1 and 2 says, dominant part of the undergraduates who completed the degree were enrolled to the degree at the age of 21 and below, which imagines that if the age is less, there exists a high likelihood of passing the degree.

4. Conclusion and Future Work

In this research study, the most prominent factors, which predict the performance of tertiary level students, are discussed. Three different data mining algorithms, i.e. Naïve Bayes, Decision Tree, and Support Vector Machine were used in the prediction stage. The results of mining algorithms were tested and the output of the algorithm, which had the highest prediction accuracy, was selected as the best prediction baseline. Accordingly, the extracted rules depicts that the age, past failure modules, performance of past semesters are having a significant correlation with the success or failure of the students of three-year degree program.

As the subsequent stage of the research study, the looks into hope to dissect the kind of the relationship measurably and will utilize other advanced mining algorithms to gauge the prediction accuracy to affirm whether similar outcomes will create by them.

Acknowledgement

The authors gratefully acknowledge Sri Lanka Institute of Information Technology for supplying students' data for the analysis.

References

[1] University of Colombo, Sri Lanka, "Student Handbook 2012/2013", http://www.bit.lk/downloads/handbook/2012-handbook.pdf. [online] accessed on 08/09/2017.

[2] Romero C, Ventura S, Data Mining in Education, WIREs Data mining knowledge discovery, Vol 3, pp. 12- 27, 2013.

[3] Kasthuriarachchi K.T.S., Chintan M. Bhatt, S.R.Liyanage, "A Review of Data Mining Methods for Educational Decision Support", International Postgraduate Research Conference, University of Kelaniya, Sri Lanka, pp- 30, December 2016.

[4] Richard A. Huebner, "A Survey of educational data-mining research," *Research in higher educational journal.*

[5] T. D. Gedeon and H. S. Turner, "Explaining student grades predicted by a neural network," in Int. Conf. Neural Netw., Nagoya, Japan, pp. 609–612, 1993.

[6] D. Shangping, Z. Ping, "A data mining algorithm in distance learning", Proc. Int. Conf. Comput. Supported Cooperative Work in Design, pp. 1014-1017, 2008.C.

[7] N. Myller, J. Suhonen, E. Sutinen, "Using data mining for improving web-based course design", Proc. Int. Conf. Comput. Educ., pp. 959-964, 2002.

[8] P. Haddawy, N. Thi, T. N. Hien, "A decision support system for evaluating international student applications", Proc. Frontiers Educ. Conf., pp. 1-4, 2007

[9] A. Nebot, F. Castro, A. Vellido, F. Mugica, "Identification of fuzzy models to predict students perfornance in an e-learning environment", Proc. Int. Conf. Web-Based Educ., pp. 74-79, 2006.

[10] S. Kotsiantis, C. Pierrakeas, and P. Pintelas, "Preventing student dropout in distance learning systems using machine learning techniques," in Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst., Oxford, U.K., pp. 3–5, 2003

[11] Jorge Chue, Juan Pablo Peche, Gustavo Alvarado, Bruno Vinatea, Jhonny Estrella, Álvaro Ortigosa, "A data mining approach to guide students through the enrolment process based on academic performance", User Modeling and User-Adapted Interaction, Springer, Volume 21, Issue 1–2, pp 217–248

[12] "What is R?, R Development Core Team 2006", https://www.r-project.org/about.html [Online] accessed on 08/09/2017

A Simulation-based Analytical Approach to Enhance Distribution Networks in Pharmaceutical Supply Chains

J. D. N. N. Rathnasiri[1] , T. D. Rupasinghe[2]
Department of Industrial Management, Faculty of Science, University of Kelaniya, Sri Lanka
[1]neranjanayani.hk@gmail.com, [2]thashika@kln.ac.lk

Abstract

This paper focuses on the most efficient factors to be considered when designing a distribution network for pharmaceutical supply chains (PSC). When considering the impact on the consumers, distributing pharmaceutical products has become a crucial process than distributing other commercial goods. The study has a two-fold approach. Where in the first phase, the paper represents a comprehensive systematic review of literature based on PSC in terms of simulation and modelling techniques and the distribution networks. In the second phase, it designs distribution network models to simulate pre-identified clusters of the pharmaceutical based on the inherent characteristics. Finally, the most efficient distribution network designs are identified for each cluster. Here, four different distribution network designs were modelled using SupplychainGuru® simulation and modelling platform. These models help make strategic decisions such as designing and operating different distribution networks to better suit the product characteristics of a given PSC by contributing to reduce overall supply chain cost, reduce overall supply chain time, increase product availability and to increase product quality in order to deliver better quality products to the patients in need.

Keywords: *Distribution networks, Pharmaceutical supply chain, Simulation and modelling*

1. Introduction

Distribution of medicine has always been a vital part of the history of human civilization. Be it a war or an outbreak of an epidemic, the pinpoint accuracy of the pharmaceutical supply chain will decide the fate of thousands of lives. It is not just delivering medicine, it needs to be efficient, safe, in addition to being time sensitive. It is found out that the pharmaceutical supply chains have a highly dynamic and uncertain behaviour. Because of this reason, it is difficult to maintain an efficient distribution network for pharmaceutical supply chains. Proposed here is a solution to this in the form of four clusters of pharmaceuticals.

2. Literature Review

Pharmaceutical Supply chains preserve higher importance than the other commercial goods and services, since it costs a significant amount of money and time to produce and distribute pharmaceutical products. Not only that, when it comes to pharmaceutical product supply chains, all the contributors have to go through strict rules and regulations laid by the governments [13]. As the distribution networks of PSC are not well-managed, thousands of patients have to suffer and significant amount of investments are wasted [3]. The displacement of pharmaceutical products from the manufacturer to its consumer involves multiple parties such as suppliers, manufacturers, distributors, pharmaceutical benefit managers, health insurance companies, hospitals and pharmacies [1]. Therefore, pharmaceutical distribution process has become highly complex and fragmented. According to the literature, researchers are yet to address the role of complex and dynamic relationships that exist between individual members of PSC [8]. When considered about all the processes in PSC, distribution networks become more important since it bears a higher cost portion among all PSC functions [11] and [7]. Since different pharmaceutical products require different distribution networks, standard logistic strategy models that have been useful to guide managerial policy in other distribution industries cannot be easily applied to PSC [5] and [6]. The importance of having an efficient distribution network design is to determine the least cost system design such that the consumer demand is satisfied with their expected level. As literature emphasis, there are two factors to be considered when designing a distribution network [2].

1.) Will the product be delivered to the consumer location or picked up from a pre-ordained site?
2.) Will the distribution network have any intermediaries?

Based on these two factors, the decision makers can decide on the various types of distribution network designs to deliver products from the manufacturers to the consumers. In this paper, the authors intend to develop the most effective distribution network design for the distribution of pharmaceutical from the manufacturer to the consumer. Here the authors consider about pharmaceutical distribution network with manufacturers, warehouses and consumers. For this study, a two-echelon network is considered. The main perspective of this study is to determine the number and locations of manufacturing plants, warehouses related to the pharmaceutical supply chains.

When considered about the PSC, it shows a very dynamic behaviour than other commercial goods. When we compare different pharmaceutical products, they tend to show a large deviation from one to another and when considered about the prominent characteristics of the pharmaceuticals, it is difficult to have one distribution network design to distribute all the pharmaceuticals. Therefore, the products need to be categorized and distributed according to their special characteristics. The literature clearly emphasizes that the traditional distribution network designs cannot be used to distribute pharmaceuticals and there should be special network design for them [14]. Inherently PSC is dynamic and risky. In order to run multiple interactive scenarios, simulation methods can be used. In the literature review, the importance of the use of simulation and modelling techniques to enhance distribution networks in PSC has been clearly laid out. Out of all approaches for network optimization, simulation and modelling techniques have been identified as one of the best means to analyze and deal with stochastic facets existing in the supply chain [7]. Since the simulation methods have the capability of capturing uncertainty and complex system dynamics make it well suited for supply chain studies, this approach has the ability to help the optimization process by evaluating the impact of alternative policies [4]. Since it is a dynamic and a risky environment, decision makers tend to use simulation and modelling approach when designing models for the PSC.

PSC is composed of multiple firms interacting to produce and distribute drugs in an uncertainty environment. Even after a successful drug development, demand is highly variable and at the end of it, patent life demand all but vanishes due to competition from the generics. To counter this uncertainty, drug-specific contracts exist between suppliers, manufacturers, and distributors those results in complex, competitive, and dynamic relationships in the PSC.

The movement of pharmaceutical products from manufacturer to its consumer involves multiple players such as suppliers, manufacturers, distributers, pharmaceutical benefit managers, health insurance companies, hospitals and pharmacies. Therefore, pharmaceutical distribution process becomes highly complex and fragmented [3]. According to the past literature, around 6% of sales is sold directly by manufacturers. Hence, it shows that the distribution networks of the pharmaceutical industry is quite complex than the other commercial goods.

There are many models developed to determine logistic strategies. These standard logistic strategic models have been useful to guide managerial policies in other distribution industries and are not easily applicable to PSC. Researchers have yet to address the role of complex and dynamic relationships that exist between individual members of PSC [3]. In a PSC, the number of consumption points, the role and number of intermediaries, and the long lead times and highly unpredictable nature of bio-pharma manufacturing have created a web of contingencies, interdependencies, and uncertainties. Because existing research does not address the realities of PSC, there is an acute need for supply chain researchers to begin investigating these phenomena.

3. Methodology

Overview of the methodology is as follows and are presented in four main phrases:

1) Systematic Review

Literature review is the first step of the research that will be conducted to identify the studies that have been already done related to PSC. In this phase, current studies are thoroughly examined to identify the fundamental characteristics in PSC. Then the

case studies are identified in order to gather data for the study.

2) Classification

Pharmaceutical products can be classified into several clusters based on their prominent characteristics. There can be different pharmaceutical such as tablets, hazard drugs, liquids and etc. There are different policies and strategies to distribute pharmaceuticals. Therefore these several types of products can be distributed in various different ways. In this phase it is decided to identify the classifications of the drugs and follow up with relevent clustering for PSC.

3) Simulation and Modelling

Simulation and modelling techniques are used since there is a significant need of using these technologies to enhance distribution networks in PSC. Simulation and modelling techniques are highly applicable for uncertain environments. In this phase, simulation and modelling techniques are used for the following purposes,

1.) Distribution network models are developed to simulate the pre-identified clusters.
2.) Use data from the previous literature to simulate the developed models.
3.) Change policies and strategies for these models and simulate them in different scenarios.
4.) Identify and analyse the behaviours of distribution networks in pre-defined clusters.
5.) Compare the behaviours of the distribution networks.
6.) Develop most capable distribution network which can be apply to enhance the distribution networks in PSC.

4) Validation and verification

- The model will be verified by testing the data using the model and testing the data under normal conditions without the model.
- The model will also be validated in this stage using appropriate validating techniques.

A. Clustering Process

As per the literature emphasis, it is ineffectual to use the same distribution network for all the pharmaceutical products, since one product is highly deviated from the other product [4]. Different types of pharmaceutical product categories require specialized distribution networks and there are different policies and strategic levels to distribute pharmaceuticals. Therefore, it is difficult to use same distribution network design for all the pharmaceutical products. Therefore, at the second stage of the study, pharmaceutical products are clustered based on their prominent characteristics. In this phase of the study, it is decided to identify the classifications of the drugs and do relevant clustering for PSC. Pharmaceutical products are clustered based on their prominent characteristics. According to the literature, four main pharmaceutical clusters were identified [11] and [12]. These identified pharmaceutical clusters are described in the Table 1.

Table 1. Description of Pharmaceutical Product Clusters

Cluster Name	Description
Time Sensitive Pharmaceutical Product Cluster	These products are highly sensitive to the time. They have a minimum lifetime. Products should be distributed within minimum time period
Hazardous Pharmaceuticals Product Cluster	Hazardous pharmaceutical products are, which has the potential to cause harm to the other products. Therefore, these pharmaceuticals should be transported individually, without having any interaction with the others.
Hybrid Pharmaceutical Product Cluster	These products are highly sensitive to the time and they have the potential to cause harm to the other products. Therefore, these products should be transported individually within minimum time period.
General Pharmaceutical Product Cluster	Products can be distributed as the commercial goods.

As described, each and every cluster consists of different characteristics. These clusters have different Key Performance Indicators (KPI). In order to identify, analyze and compare the behaviours of each and every cluster, the relevant KPI's were identified first. These identified KPI's are described in the Table 2. When simulation techniques are used on the distribution networks of each and every pharmaceutical cluster, these KPI's are used to identify the necessary requirements related to each and every pharmaceutical cluster [14].

Table 2. Considered In The Pharmaceutical Clusters

Cluster Name	KPI's
Time Sensitive Pharmaceutical Product Cluster	Distribute products using minimum time Distribute Products before the due date Minimize the product exchanging rate among distribution centers. Actual adjusted on-time delivery compared to the customer service level target established
Hazardous Pharmaceuticals Product Cluster	Maximize the usage of possible warehouses. Monthly tracking of slow moving inventory Actual cycle count of finished goods inventory
Hybrid Pharmaceutical Product Cluster	Distribute products using minimum time Distribute Products before the due date Minimize the product exchanging rate among distribution centers. Maximize the usage of possible warehouses.
General Pharmaceutical Product Cluster	Minimize the total cost (Inventory cost, Distribution cost, Operational cost)

B. Simulation And Modelling Procedure

In order to identify the most efficient distribution network design for each and every pharmaceutical cluster, the simulation techniques are used. Since the authors consider the problem from the process side, discrete event simulation method is used. As described earlier, simulation and modelling techniques are highly applicable in uncertain environments [10]. For the model development process, the authors derived the test cases from the literature and used them as the input data of the model development process [8]. These derived test cases different from one to another based on the prominent characteristics align with each pharmaceutical cluster. This network design is used to distribute all the types of pharmaceutical products in the industry. All the four pharmaceutical product clusters use the same distribution network design, which is shown in Figure 1.

This is a multi-sourcing distribution network and all the plants and distribution centers connect to each other. Then the distributed network design is modelled using modelling and simulation tool. State-of-the-art supply chain modelling and simulation software, Supply Chain Guru ® was used for the simulations. The modelled design of the baseline scenario is shown in Figure 1. Then the interactive scenarios were run related to each and every cluster to identify best distribution network design for them. Here, scenarios were selected based on the KPI's which are mentioned in Table 2 and the constraints related to the clusters. Then the different scenarios were run to identify the most suitable distribution network design for each and every cluster. Then different distribution networks for different clusters were derived. The relevant scenarios for the pharmaceutical clusters are as mentioned in Table 3. As the final stage, the new distribution network design and the previous network design were compared.

C. Results

When distribution network design changes from the baseline design to another, the cost of the distribution network is reduced. The results of the simulation process are as follows.

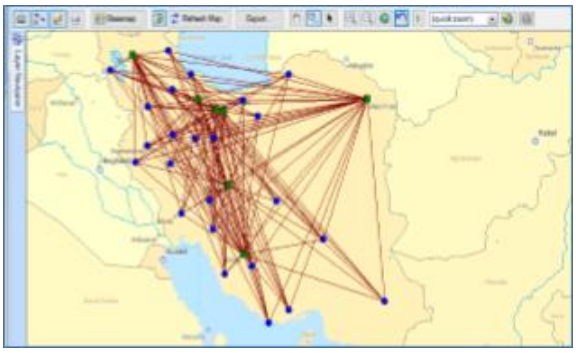


Figure 1. Baseline scenario of PSC

Table 3. Scenarios Considered For the Simulation Process

Cluster Name	Scenarios
Time Sensitive Pharmaceutical Product Cluster	Here time constraints such as expiry dates and minimum travelling time is considered to ensure that the products are distributed within minimum time throughout the distribution network.
Hazardous Pharmaceuticals Product Cluster	Here it is a must to assign one product time to one warehouse. Therefore, all the possible warehouses need to be considered in the distribution network.
Hybrid Pharmaceutical Product Cluster	All the scenarios, which were considered for both Time Sensitive Pharmaceutical Product Cluster and Hazardous Pharmaceuticals Product Cluster, were considered.
General Pharmaceutical Product Cluster	Consider as a general distribution network. Instead of creating transportation policies for every source/destination/product, author use groups to apply the same policy across a number of products, sources, or destinations.

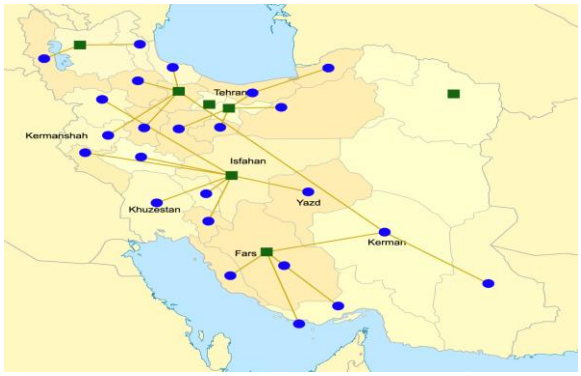


Figure 2. Optimized Distribution Network Design for Hybrid Pharmaceutical Cluster

When time sensitive pharmaceutical products are distributed using the network design mention in Figure 3, rather than using the baseline scenario, the distribution cost can be reduced by 5.88 percent

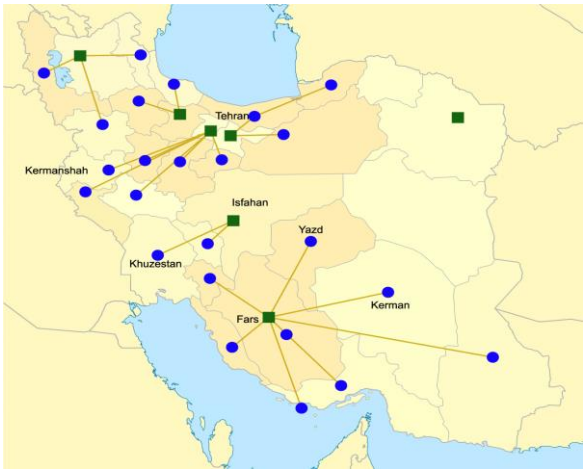


Figure 3. Optimized Distribution Network Design for Time Sensitive Pharmaceutical Cluster

When hybrid pharmaceutical products are distributed using the network design mention in Figure 2, rather than using the baseline scenario, the distribution cost can be reduced by 6.34 percent. When hazardous pharmaceutical products are distributed using the network design mention in Figure 4, rather than using the baseline scenario, the distribution cost can be reduced by 10.82 percent. When general pharmaceutical products are distributed using the network design mention in Figure 5, rather than using the baseline scenario, the distribution cost can be reduced by 72.1 percent.

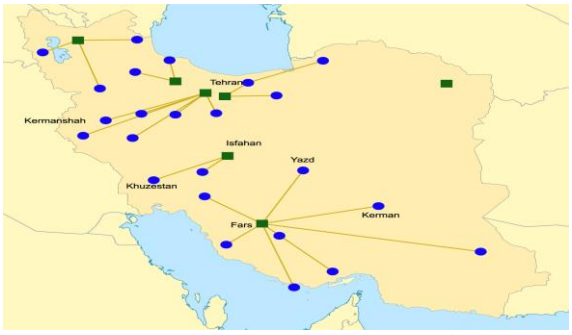


Figure 4. Optimized Distribution Network Design for Hazardous Pharmaceutical Cluster

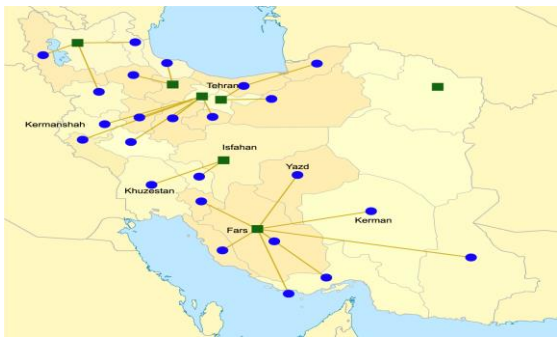


Figure 5. Optimized Distribution Network Design for General Pharmaceutical Cluster

4. Conclusions

The simulation model developed in this paper addresses the PSC distribution network design problems. The aim of this study is to design most effective distribution network design considering different clusters of pharmaceuticals. Through the application of the modelling and simulation techniques, the most efficient distribution network designs were developed for all the four main pharmaceutical clusters. The cost differences of the distribution network designs are summarized in the Table 4 and Figure 6.

Table 4. Percentage of Cost Changes

Cluster Name	Percentage
Time Sensitive Pharmaceutical Product Cluster	5.88%
Hazardous Pharmaceuticals Product Cluster	10.82%
Hybrid Pharmaceutical Product Cluster	6.34%
General Pharmaceutical Product Cluster	72.1%

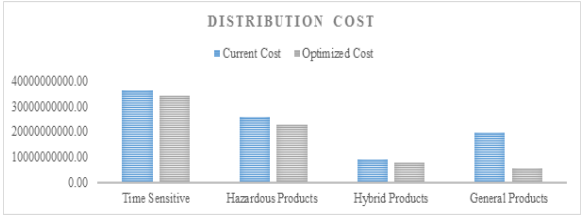


Figure 6. Cost Differences

Since PSC deals with higher degree of uncertainty in input parameters, simulation-based analytical approach is applied to cope with the dynamic relationships. This study emphasis that, as a result of highly dynamic behaviours and the prominent characteristics of the PSC it is difficult to distribute pharmaceuticals by using general distribution network designs. Hence, through this study four main pharmaceutical distribution clusters were identified. Then, simulation-based analytical approach is used to identify the most efficient distribution network design for PSC clusters. Through the results of the study, it is clear that distributing products with in the relevant cluster makes a significant cost difference. The literature emphasized that due to the highly dynamic behaviours of the PSC it is difficult to find solutions to the problems in PSC. This study suggests clustering approach to categorize products with equal characteristics. This study introduces four main pharmaceutical distribution clusters with the relevant KPI's.

The clustering approach can be used to reduce the influence of the dynamic behaviours of PSC. Furthermore, this study suggests the scenarios for all the clusters to simulate and find the most efficient distribution network designs. This study proposes new avenues in designing and operating different distribution networks to better suit the product characteristics of a given PSC in order to deliver better quality products to the patients in need.

References

[1] Berman, O. and Wang, Q., Inbound Logistic Planning: Minimizing Transportation and Inventory Cost, Transportation Science, vol. 40, no. 3, pp. 287-299, 2006.

[2] Chopra, s. and Meindi, p.,Supply Chain Management, 2001

[3] Jetly, G., Rossetti, C. and Handfield, R., A multi-agent simulation of the pharmaceutical supply chain, Journal of Simulation, vol. 6, no. 4, pp. 215-226, 2012.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp. 68-73.

[4] Ding, H., Benyoucef, L., Xie, X., Hans, C. and Schumacher, J., "ONE" A NEW TOOL FOR SUPPLY CHAIN NETWORK OPTIMIZATION AND SIMULATION, Proceedings of the 2004 Winter Simulation Conference, 2004.

[5] Goetschalckx, M., Vidal, C.J. and Dogan, K. (2002), "Modeling and design of global logistics systems: a review of integrated strategic and tactical models and design algorithms", European Journal of Operational Research, Vol. 143 No. 1, pp. 1-18.

[6] Elkins, D., Handfield, R.B., Blackhurst, J. and Craighead, C.W. (2005), "18 ways to guard against disruption", Supply Chain Management Review, Vol. 9 No. 1, pp. 46-53.

[7] Swaminathan, J.M., Smith, S.F. and Sadeh, N.M. (1998), "Modeling supply chain dynamics: a multiagent approach", Decision Sciences,Vol. 29 No. 3, pp.607-32.

[8] Jetly, G., Rossetti, C. and Handfield, R., A multi-agent simulation of the pharmaceutical supply chain, Journal of Simulation, vol. 6, no. 4, pp. 215-226, 2012.

[9] Lendermann, P. and Gan, B., Distributed simulation with incorporated aps procedures for high-fidelity supply chain optimization, Proceedings of the 2001 Winter Simulation Conferenc, 2001.

[10] Selim, H. and Ozkarahan, I., A supply chain distribution network design model: An interactive fuzzy goal programming_based solution approach, 2008.

[11] Solo, K. and Paich, M., A Modern Simulation Approach for Pharmaceutical Portfolio Management, 2004

[12] Yadav, P., Tata, H. and Babaley, M., The World Medicines Situation 2011 Storage And Supply Chain Management, 3rd ed. World Health Organization, 2011.

[13] http://www.who.int/topics/pharmaceutical_products/en FOX, M., BARBUCEANU, M. and TEIGEN, R., Agent-Oriented Supply-Chain Management, 2000.

[14] Rupasinghe, T., and Rathnasiri, J.D.N.N, Applicability of Simulation Techniques to Enhance the Distribution Networks in Pharmaceutical Supply Chains, Wayamba Simulation Conference, 2016

Programme Committee

- Prof. Asoka S. Karunananda (Conference Chair), University of Moratuwa, Sri Lanka.
- Dr. (Mrs.) Menaka Ranasinghe, The Open University of Sri Lanka.
- Dr. (Mrs.) Uditha Ratnayake, The Open University of Sri Lanka.
- Dr. (Mrs.) Subha Fernando, University of Moratuwa, Sri Lanka.
- Dr. (Mrs.) Thushari Silva, University of Moratuwa, Sri Lanka.
- Dr. Sagara Sumathipala, University of Moratuwa, Sri Lanka.
- Mr. Budditha Hettige, General Sir John Kotelawala Defence University, Sri Lanka.
- Mrs. Mihirini Wagarachchi, University of Ruhuna, Sri Lanka.
- Mr. Dilan Wijerathne, Nations Trust Bank, Sri Lanka.
- Mrs. Chinthanie Weerakoon, University of Kelaniya, Sri Lanka.
- Mr. Tharindu Adhikari.

Panel of Reviewers

- Prof. Asoka S. Karunananda, University of Moratuwa, Sri Lanka.
- Prof. Nalin Wickramarachchi, University of Moratuwa, Sri Lanka.
- Dr. Prasad Jayaweera, University of Sri Jayewardenepura, Sri Lanka.
- Dr. Ajith Madurapperuma, The Open University of Sri Lanka.
- Dr. (Mrs.) Uditha Ratnayake, The Open University of Sri Lanka.
- Dr. (Mrs.) Menaka Ranasinghe, The Open University of Sri Lanka.
- Dr. (Mrs.) Subha Fernando, University of Moratuwa, Sri Lanka.
- Dr. (Mrs.) Thushari Silva, University of Moratuwa, Sri Lanka.
- Dr. Sidath Liyanage, University of Kelaniya, Sri Lanka.
- Dr. Sagara Sumathipala, University of Moratuwa, Sri Lanka.
- Dr. A. Pasqual, University of Moratuwa, Sri Lanka.
- Mr. G. Anthonys, University of Waikato, Newzealand.
- Mr. C. J. Basnayakage, The Open University of Sri Lanka.
- Mr. S. Deegalla, University of Peradeniya, Sri Lanka.

Technical Session Chairs

- Dr. Ruwan Weerasinghe, University of Colombo School of Computing, Sri Lanka.
- Prof. Asoka S. Karunananda, University of Moratuwa, Sri Lanka.

Panel of Judges– Best paper Award

- Prof. R. G. N. Meegama, University of Sri Jayewardenepura, Sri Lanka.
- Prof. N. G. J. Dias, University of Kelaniya, Sri Lanka.
- Prof. R. P. C. J. Rajapakse, University of Moratuwa, Sri Lanka.

Gold Sponsored By

ISSN 2602-8549