|  |  |  |
| --- | --- | --- |
| UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING | | |
| Draft Introduction and Literature Review | | |
| 3204 - Individual Project | | |
| **Project Name** | **:** | **Content and Collaborative based Sinhala Book**  **Recommendation System** |
| **Supervisor** | **:** | Dr. M. G. N. A. S. Fernando |
| **Student Name** | **:** | P. N. C. Perera |
| **Index No** | **:** | 19440677 |
| **Registration No** | **:** | 2019/MCS/067 |

**Table of Contents**

[Chapter Overview 5](#_bookmark0)

[Introduction 5](#_bookmark1)

[Problem domain/ Background 6](#_bookmark3)

[English Language 7](#_bookmark6)

[Sinhala Language 7](#_bookmark8)

[Problem Definition 9](#_bookmark9)

[Research Motivation 9](#_bookmark10)

[Research Gap 9](#_bookmark11)

[Research Contribution 10](#_bookmark12)

[Domain Contribution 10](#_bookmark13)

[Computer Science Contribution 10](#_bookmark14)

[Research challenge 10](#_bookmark15)

[Research Questions 10](#_bookmark16)

[Research Aim 10](#_bookmark17)

[Research Objectives 11](#_bookmark18)

[Project Scope 11](#_bookmark19)

[Resource Requirement 12](#_bookmark20)

[Software Requirements 12](#_bookmark21)

[Hardware Requirement 12](#_bookmark23)

[Other skills required 12](#_bookmark24)

[Chapter Summary 12](#_bookmark25)

[Literature review 12](#_bookmark26)

[Tools and Techniques 13](#_bookmark27)

[Natural Language Processing (NLP) 13](#_bookmark28)

[Machine Learning (ML) 14](#_bookmark30)

[Machine Translation (MT) 14](#_bookmark31)

[Existing approaches for machine translation 14](#_bookmark32)

[English- Hindi /one Indian language to another Indian language 14](#_bookmark33)

[English to Japanese language / Japanese to English language 15](#_bookmark35)

[Hindi to English language 15](#_bookmark37)

[Pali to Sinhala Language 16](#_bookmark38)

[English to Arabic language 16](#_bookmark39)

[Hindi / Bengali to English language 16](#_bookmark40)

[Babel Fish 17](#_bookmark41)

[Approach for the project 17](#_bookmark42)

[Translation Dictionary 17](#_bookmark43)

[Occam's Razor principle 17](#_bookmark44)

[Language model 18](#_bookmark45)

[Probabilistic model of language 18](#_bookmark46)

[N-gram language model 19](#_bookmark47)

[Recurrent Neural Network (RNN) 19](#_bookmark48)

[Long Short Term Memory (LSTM) 20](#_bookmark51)

[Bidirectional Encoder Representation from Transformers (BERT) 22](#_bookmark55)

[Related work 23](#_bookmark57)

[Approaches for English to Sinhala translation 23](#_bookmark58)

[Chapter Summary 25](#_bookmark62)

[Problem analysis and Methodology 26](#_bookmark63)

[Representation of the Problem 26](#_bookmark64)

[Proposed system architecture 26](#_bookmark65)

[Translation Dictionary 27](#_bookmark67)

[Language Model 28](#_bookmark70)

[Training Model Creation 28](#_bookmark71)

[Initial Classifier 29](#_bookmark72)

[Initial Training Model 31](#_bookmark79)

[Test Model 36](#_bookmark94)

[Chapter Summary 39](#_bookmark103)

[Project Plan and Timeline 39](#_bookmark104)

[References 40](#_bookmark105)

**List of Figures**

[Figure 1 : Usage statistics of Sinhala, Sinhalese as content language on the web 5](#_bookmark2)

[Figure 2 : Syntactic level ambiguity 13](#_bookmark29)

[Figure 3 : English to Hindi translation of a sentence 15](#_bookmark34)

[Figure 4 : English to Japanese translation 15](#_bookmark36)

[Figure 5 : Recurrent Neural Network 19](#_bookmark49)

[Figure 6 : Unfolded Recurrent Neural Network 20](#_bookmark50)

[Figure 7 : Internal structure of RNN 20](#_bookmark52)

[Figure 8 : Internal structure of LSTM network 21](#_bookmark53)

[Figure 9 : Symbols used for internal networks 21](#_bookmark54)

[Figure 10 : Expected vs. Actual for the "duck" word 23](#_bookmark56)

[Figure 11 : Generalize version for English to Sinhala translation 24](#_bookmark59)

[Figure 12 : EnSiTip result from a web page 24](#_bookmark60)

[Figure 13 : BEES high-level architecture 25](#_bookmark61)

[Figure 14: Proposed system architecture 26](#_bookmark66)

[Figure 15: Collection of Sinhala to English meanings 27](#_bookmark68)

[Figure 16: Collection of English word with Sinhala meanings 28](#_bookmark69)

[Figure 17: Different classes for specific words. 29](#_bookmark73)

[Figure 18: Different meaning for the word "Mine" 29](#_bookmark74)

[Figure 19: Installed and imported Python libraries 31](#_bookmark81)

[Figure 20: Read training model 32](#_bookmark82)

[Figure 21: Content of the training model 32](#_bookmark83)

[Figure 22: Import BERT model 33](#_bookmark84)

[Figure 23: Chunking mechanism in BERT 33](#_bookmark85)

[Figure 24 : Embedding the sentences 33](#_bookmark86)

[Figure 25: Tokens and token embedding for the sentence 33](#_bookmark87)

[Figure 26: Train the model with specific word 34](#_bookmark88)

[Figure 27: Reduce the dimensions using PCA 34](#_bookmark89)

[Figure 28: Maximum variant for the first principle component 35](#_bookmark90)

[Figure 29: Plot the training model 35](#_bookmark91)

[Figure 30: Accuracy of the training model 35](#_bookmark92)

[Figure 31: Scatter plot of the word "Mine" 36](#_bookmark93)

[Figure 32: Tokenized sentences for the test model 37](#_bookmark96)

[Figure 33: Sentences showing the probabilities for each class 37](#_bookmark97)

[Figure 34: Predicted classes for test model 37](#_bookmark98)

[Figure 35: Code to read classifier to show the class names 38](#_bookmark99)

[Figure 36: Output of the test model 38](#_bookmark100)

[Figure 37: Code to display BLEU score for the sentence 38](#_bookmark101)

[Figure 38: BLEU score for the sentence 38](#_bookmark102)

# List of Tables

[Table 1 : Relations between senses 6](#_bookmark4)

[Table 2 : Some of the Homonyms in English language 7](#_bookmark5)

[Table 3 : Differences between British English and American English 7](#_bookmark7)

[Table 4: English to Hindi translation of a sentence 12](#_bookmark22)

[Table 5 : Different definitions in Sinhala to the word "Mine" 30](#_bookmark75)

[Table 6: One definition highlighted for the word "Mine" 30](#_bookmark76)

[Table 7: Different definitions of Word "Run" based on tense 30](#_bookmark77)

[Table 8: Selected definition for word "Run" 30](#_bookmark78)

[Table 9: Initial training model 31](#_bookmark80)

[Table 10: Test Model 36](#_bookmark95)

# Introduction

# Chapter Overview

This chapter provides a foreword for the project in terms of background study, problem domain, the main aim, objectives, scope and activities that will be carried out towards the completion of the research. Finally, the chapter concludes with an overview on how the other chapters of the document fit into the project context.

# Background

From our childhood, everyone has heard that “Reading makes a man perfect”. People acquire the knowledge by reading a variety of materials. These materials could be a book, an internet article, a newspaper, a magazine, or even a piece of paper, and the gain knowledge by reading these materials is intense. People who read a lot tend to know more about life and are smarter when making decisions and handling difficult situations. (Marappan 2022) It may not be possible for the reader to “know it all,” but a lot of reading brings man close to perfection. Most of them like to read books as a hobby because it imagines readers' own movie in their mind rather than watching a movie directed by someone.

Book readers normally select books by reading some random pages or recommended by someone. When reading that book, if he finds that the book is not interesting, he will not read any book after that. therefore, it is better to suggest books that he is interested in. With the increase in library collections, it is diﬃcult for readers to quickly ﬁnd the books they want when choosing books. It is diﬃcult for readers to ﬁnd Sinhala books of interest in a short period of time in the face of various bibliographies. Therefore, the user experience of the traditional library borrowing method is poor (Dhanda and Verma 2016). Due to the Covid-19 pandemic situation and the geographical barriers also it becomes a tremendous challenge for them (Sarma, Mittra, and Shahadat 2021) to find a relevant book as they do not like to go out and spend time searching books of their preference.

# Motivation

When we navigate through social media specially in Facebook there are so many groups available for almost everything. If you are living in an area, there is a group for that area, if you have an aqua car, there is a group created for aqua car owners. The benefits of such group are you can learn many things and if you have any doubt you can ask from the group and get it clarified. For Sinhala book readers also, there are so many groups available in the Facebook. You can share what your thought for a book or you can see what are latest books released through the groups if you follow those groups. One thing I have noticed is many people ask I have read ‘Adaraniya Victoria’ any one can suggest similar type of books. And some of members ask what are the books related to Sri Lankan history. Some other have asked whether the book is good to read by uploading an image of the book. When considering these three scenarios I thought like it is better to have a system that displays what other users’ thoughts about a book and how much of rate could be given to the book and what are other similar books. It gives the motivation to initiate kind of such system for Sinhala book readers.



# Problem Domain

Most organizations have implemented their recommendation system when users buy products online. But almost all the websites are not developed for the buyer's interest; the organizations force add-on sales to buyers by recommending unnecessary and irrelevant products (Sarma, Mittra, and Shahadat 2021). For instance, if a user has read a book named ‘Madol Duwa’, he would like to read similar books and there is no Sinhala book recommendation system to address this problem. Additionally, some members of readers groups on Facebook have problems like, is this book good or I have read this book and are there any similar kind of books like the mentioned book? Many personal book recommendation systems have emerged to conduct eﬀective search based on user rating and interest.

This paper proposed an effective Sinhala book recommendation system for online users that rated a book list using the content and collaboration (hybrid) method. The solution could be used by all Sinhala book readers to find interesting or suitable books without wasting time or money. Authors also could use the system to have an idea of what kind of books readers rate and are interested more and write books accordingly.

# Aim

The main aim of the research is to analyze, design, implement and evaluate an accurate Sinhala book recommendation system using content and collaboration algorithms with attractive user-friendly interface which display the searched book details along with reviews and recommended books. The main aim can be further divided to three sub aims as;

* Input Data will be the dataset collected from readers.
* Preprocess by removing null values and unwanted data then apply the content and collaborative algorithm
* Final output data will be displayed in the system.

# Objective

The final outcome of this project is helping Sinhala book readers to find correct and recommend books based on their preferences using content and collaboration algorithm. There is no exact formula for determining how much data would be enough for a recommendation system. Even though capturing many data sets, ends up with manageable data sets after preprocessing and removing null values from the collection list. The most online dataset contains parsed data.

The main goal of this research will be achieved by targeting the following objectives.

1. To Collect selected Sinhala book details like title, author, publisher, description, image url and keywords. Online book stores will be used for collecting the details of books.
2. To produce a data set containing user details, selected Sinhala books and rates given by users for those books.
3. To implement web application along with login and registration features.
4. To recommend ten books based on specific field of interest using Content and Collaborative (Hybrid) methodology.
5. To determine the categories preferred by readers so that it motivates authors to write books as per the user’s preference.
6. To Increase the number of book readers by recommending books according to their preferences.

# Scope

The scope of the project can be defined as bellow.

1. Sharing a google form containing selected books along with authors and collecting user rates and reviews for those books based on previous readers experience.
2. Applying sentimental analysis for the reviews collected from above and assigning a new rate.
3. Provide facilities for readers to login or register to the system with their email id which is unique.
4. Provide facilities for readers to search a specific book in the repository and it will display the details such as author, publisher and image along with the reviews and rates given by users.
5. Additionally, the system will display a list of recommended Sinhala books based on users’ rates and reviews of specific interested field. Content based and Collaborative based (Hybrid) approach will be taken place in order to recommend books.
6. Only Sinhala books are recommended and it is based on users’ rates and reviews as well as keywords provided for selected books

We can use library to collect book details, but we will not be able to collect user reviews and rates for selected books. I am using online book store to collect book details but I preferred to get user reviews and rates from users themselves so that they are aware that they have provided the information for the application rather than just coping from the online without their awareness. Until now, around 3750 records have been collected and targeting to collect around 7000 records for the research.

# Resource Requirement

In order to implement and execute the application, following hardware and software requirements should be satisfied.

# Hardware requirement

* A Laptop or desktop with core i3 or above processor
* At least 4GB Ram
* At least 30GB

# Software requirement

* Python latest version – 3.11.5
* VS code as IDE for implementation and execute the application
* MS Excel and Notepad ++ for viewing and manipulating data
* Stable internet connection for downloading relevant libraries.
* GitHub for storing images and implemented code.

# Chapter Walkthrough

The outline of the chapters are as follows.

# Chapter 02: Literature Survey

This chapter will discuss about the review, conducted on the proposed project. It will extensively describe on the stakeholders, the problem, existing solutions, methodologies, and approaches along with their benefits and limitations.

# Chapter 03: Methodology

This chapter will discuss about the methodology to be used to implement the solution. The stakeholders, main technology, libraries, prioritized items, how the collected data is analyzed and the how the architecture of the system will be organized will be in detailed discussed. Furthermore, why the selected technology is more suitable than other existing technologies will be clarified.

# Chapter 04: Implementation

This chapter covers the implementation stage of the project. Algorithms used and challenges faced and how they are resolved will be discussed in this phrase. Screen shots and code segments for some selected functionalities are also provided to facilitate easier understanding and manipulating over the project implementation.

# Chapter 05: Evaluation and Results

The evaluation chapter provides how the results are evaluated based on the feedback collected from Domain experts in this projects Authors. The project will be shown to them and get the feedback for the evaluation. Other than that validation methods will be used to further evaluate the accuracy of the system.

# Chapter 06: Conclusion and Future work

The objectives that were able to be successfully achieved will be discussed in conclusion chapter. The challengers and the limitation of implemented system will be highlighted in order for someone to enhance the system.

# Chapter Summary

The chapter began with explanation on background and problem domain of the system. Although many applications have been developed for book recommending systems, most of them are related to English books. Proper applications that satisfy all the requirements with user satisfaction were limited. The main approach is to make an application that help all Sinhala book readers to recommend Sinhala books based on their preference. A goal followed by objectives was defined to make the effort to be success.

# Literature Review

# Chapter Overview

This chapter provides a foreword for the project in terms of background study, problem domain, the main aim, objectives, scope and activities that will be carried out towards the completion of the research. Finally, the chapter concludes with an overview on how the other chapters

# Chapter Summary

This chapter

# Methodology

# Chapter Summary

Earlier Literature review chapter helped to identified what are similar systems available in the world and what are the limitations in those applications. This chapter is mainly focused on how the problem is analyzed and identify the methodologies that can be used to implement the system. Further why hybrid based application is focused rather than one particular model will be discussed in details.

# Architectural Diagram

# sssss

# Technology Selection

# Implementation

# Chapter Summary

After describing the methodology of the system, the next task is to convert the methodology into a functional prototype. The prototype of the proposed system should address the main objectives that were identified in the first chapter. This chapter will discuss the implementation details individually for identified modules. At the end of the chapter the decisions taken on the low-level implementation would be discussed.

# Preprocessing

When considering the dataset, it is noted that the dataset contains reviews that were written in Sinhala language. Since we are planning to apply sentiment analysis for the reviews, it is compulsory to have the review data in English language. There for the very first task would be to convert the reviews written in Sinhala to English language.

# Language Translation

In order to convert the language, Google Translator library could be used as below. Then all the reviews written in Sinhala will be converted to English reviews which ultimately could be applied sentiment analysis on top of the reviews.

from googletrans import Translator

translator = Translator()



# Sentimental analysis

There are several ways to apply sentiment analysis for a text. Following are some of them

# Using libraries

# SentimentIntensityAnalyzer

# TextBlob

# Spacy

# Using own mechanism

# Convert Uppercase to Lowercase

# Remove Links

# Remove Punctuations

# Remove Numbers

# Remove Stop words

# Apply Stemming

# Build Vocabulary

# Vectorization

# Model training and Evaluaion

# Logistic Regression

# Collaboration based Filter

# Content based Filter

# Web Application

# User Interface

# Authenticate

# Database

# Evaluation and Results

# Chapter Summary

# Conclusion and Future work

# Chapter Summary

# 

# Chapter Overview

There are several ways people can acquire knowledge. Reading a web material or reading a physical book, watching videos and learning from a lecturer in a physical or online classroom are some of them. Each and every way specified above has different aspects. For example reading a web material is different than reading a physical book. There are many things that can perform when reading web material such as select a specific word, copy and paste it in an online dictionary available to get the meaning. But when reading a physical book if there is a difficult word, that person should type it in an online dictionary or search it using a physical dictionary. It is clearly understandable fact that, some ways have easy methods while some have time consuming and need a considerable amount of effort to put to accomplish the task. Unlike past, nowadays people tend to use online resources for their day to day lives. According to the statics by January 2022, 63.6 % of the websites uses English language and the second highest is by Russian language which is 6.9%. Compared to the English language, usage of Russian language is very small. It is a clear indication that most of the online resources are available in English language. According to the usage statics of Sinhala, Sinhalese (See Figure 1), it used by 0.1% in web content over the other language available in the world.

When consider about the Sinhala language, in Sri Lanka there are several webs sites use Sinhala with user preference. Hirufm.lk, Hirunews.lk, Gossiplankanews.lk, Adaderana.lk and Lankadeepa.lk are some of them. Most of the above sites have options to view the content in three languages Sinhala, English and Tamil which make the user to view the content which they familiar.

# Problem domain/ Background

There are several different languages around the world. People may fluent in one or many languages. When carefully analyze a language, every language there is different relations in the senses of words. Relations can be mainly categorized into three, namely: meaning, written form and spoken form. For example there may be words with same meaning and written form or same written form and spoken form likewise there are different combinations. Below is an example of some words in English language categorized into above three types (See Figure 2).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Name of Relation* | *Spoken form* | *Written Form* | *Meaning* | *Examples* |
| Synonym | different | different | same | **happy, glad, joyful,**  **blissful** |
| Homophones | same | different | different | **alms** (almsgiving)  **arms** (hand) |
| **whole** (all/ everything) **hole** (hollow / cavity) **hall** (lobby / building  or large room used for meetings) |
| Homographs | different | same | different | **bass** (fish)  **bass** (music) |
| Homonyms | same | same | different | **bat** (a nocturnal flying mammal)  **bat** (use in sports Cricket/ baseball) |
| Orthographic Variants | same | different | same | **Color Colour** |
| Phonetic Variants | different | same | same | **either**  /iy dh er /  /ay dh er/ |

Table 1 : Relations between senses

# Homonyms

According to the above Figure there are several relations but this research mainly focuses about the homonyms. Homonyms are words that are spelled the same and sound the same but have different meanings. The prefix homo- which means “the same” and the suffix –nym stands for “name”. Therefore a homonym is a word that has at least two different meanings, even though all uses look and sound exactly alike.

|  |  |  |
| --- | --- | --- |
| *Homonym* | *Meaning 1* | *Meaning 2* |
| Address | To speak to | Location |
| Bark | A tree's out layer | The sound a dog makes |
| Fair | Equitable | Beautiful |
| Mean | Average | Not nice |
| Rock | A genre of music | Stone |

Table 2 : Some of the Homonyms in English language

If a person is having less vocabulary, may not understand the correct meaning of the content. Therefore they face difficulties specially when referring to the several sources of particular language which is not there mother tongue. For this research, consider about the online web content as a source because a person is mostly refer them in their day to day life.

# English Language

English is a West Germanic Language which originating from England. It is considered as a universal language. People from non-English speaking countries consider knowing English language is an additional advantage for their academic and professional carrier. Therefore they try their best to learn English language as it common way to communication. The Latin script which is modern English alphabet contains 26 letters which also have capitals from. There are 5 vowels and the pronunciation of vowels in English makes varieties of English. The structure of the English sentence is form by Subject, Verb and Object format. For academic standards there are two national standards are present namely British English and American English. They differ by spellings, grammar and vocabulary (See Figure 3).

For example,

|  |  |  |
| --- | --- | --- |
| *Type of the difference* | *American English* | *British English* |
| Spellings | defense, offense, license | defence, offence, licence |
| color, behavior, mold | colour, behaviour, mould |
| Grammar | „gotten‟ as the past participle of „get‟ | „got‟ |
| Vocabulary | mailbox | postbox |
| apartment | flat |

Table 3 : Differences between British English and American English

Translating from English language to German or French is quite easier than translate to Sinhala because French and German languages have close relationship among each other since all of them are based on a similar alphabet.

# Sinhala Language

Sinhala is derived from ancient Brahmic and is an Indo-Aryan language primarily spoken by Sinhalese people of Sri Lanka. It is written using Sinhala Script and written from left to right. To represent different sounds diacritics called “ ” (“ ”, “ ”, “ ”, “ ”

etc…) are used in before, after, above or below the base-consonant. It does not have upper or lower case letters but it has ( , ග, ච, ට, ත, etc…) and (ඛ, ඝ, ඡ, ඨ, ථ, ඵ etc…) letters which have more weight when pronounce. Structure of Sinhala sentence is form by Subject, Object and Verb format.

The most significant difference in Sinhala letters over other languages are Sinhala Letters are curlicues. These Sinhala letters are ordered into mainly two sets namely Pure Sinhala and Mixed Sinhala. Pure Sinhala ( ධ ), which is a subset of Mixed Sinhala ( ) alphabet. The complete Sinhala script consists of about 60 letters, 18 for vowels and 42 for consonants.

However, only 57 (16 vowels and 41 consonants) are required for writing Pure Sinhala. In Sinhala language the subject has to agree with the tense, gender and the singular and plural (form) of the verb.

For example:

1. “He ran for the train” is translated to “ ට ”.
2. “She ran for the train” is translated to “ ට ”.
3. “I ran for the train” is translated to “ ට ”.
4. “They ran for the train” is translated to “ ට ”.

The verb “ran” in four sentences explained above translated to “ ” , “ ” , “ ” and “ ” respectively. First two sentences are translated “ ” and “ ” because of the gender specified in the sentence. The last two sentences are translated to “ ” and

“ ” because of the singular and plural from of the sentence. But in English language, single word ”ran”(Verb) used for above four sentences. Unlike English any written Sinhalese word is pronounced the same way it is written but there are several variations, accents, words of Sinhala use in several parts of Sri Lanka.

According to the Sri Lanka‟s Department of census and statistics in 2011 the population was made up of Sinhala (82%), Tamil (9.4%), Moor (7. %), Burgher (0.2%) and others (0.5%). The literacy rate of Sri Lanka based on national statistics placed at 91.1% and majority of the Sri Lankans speak the national languages of Sinhala and Tamil (81.8% Sinhala and 14.9% Tamil). Most of the Sri Lankans use English as their second language except Burgher and Malay. As most of the Sri Lankans‟ mother tongue is Sinhala they tend to use Sinhala when reading, writing, listening and speaking.

As discussed above there are very few websites in Sri Lanka is capable to produce Sinhala content in their websites. But for educational or any other purpose people need to refer to the online resources which are not viewable using Sinhala language. Each any every online resources have different language literacy level because the author‟s English language literacy is directly reflects to its content. This language barrier become is problematic not only Sri Lankans but also non-native English people over the world.

# Problem Definition

During the Covid19 pandemic situation people tend to refer online materials for communication unlike early days. Most of the web materials are in English language. So as a Sri Lankan there are only few websites available in Sinhala language but most of them are in English. Therefore if that person is not good at English will have a problem when translating a sentence to Sinhala.

There are some plugins are available to translate the whole sentence or a specific word in Sinhala but with the homonyms of English as discussed above, those tools may not suggest the correct word based on the context. Therefore the proposed system is capable of mouse-hover to specific word and the correct Sinhala translation for that word based on that sentence will pop up in the tooltip.

# Research Motivation

As discussed above currently available tools for English to Sinhala translation will translate the whole sentence. Some are capable of giving all the meaning for that particular English word if it is a homonym. Sometimes people need to type or copy and paste them in an online available dictionary and get the meaning. Those are time consuming and need several steps to accomplish a task. Therefore plugin will help user to stay on the current web page and using mouse-hover to specific word they can see the exact meaning of that word based on the context even though that word is a homonym. There is a big advantage when user can get by using this plugin. That is, it will also help user to increase their vocabulary unknowingly. If the tool is translating a whole sentence user will read the whole translated sentence in Sinhala but he may not even read that sentence in English and identify the difficult word that he may don‟t understand. But using this time to time he may mouse-hover to that word and get the correct meaning. This will help user to read the word and think whether he knows the meaning. If he doesn‟t know then he will mouse- hover to that word. This will help user to learn and buildup their vocabulary unintentionally.

# Research Gap

Most of the researches were conducted to translate the English sentences or words to Sinhala using a rule based techniques with Syntax and Semantics structures. And the EnSiTip plugin which also support user to get meaning of a specific word by mouse-hovering but it contains several different meanings for that word and not consider about the context of the sentence. Unlike English language it is difficult to find corpus for Sinhala word with their meaning for several different sentences. Because of that, most of the researches based on rules. Therefore the proposed system will help to fill the gap by identifying the correct meaning of the word using machine learning approach. Creating a training model for specific word needs to have huge collection of words. Implementing a corpus then help other researches to add new sentences to the collections and have a good accuracy levels for words in future researches.

# Research Contribution

There are several contributions can be get by the proposed system.

### Domain Contribution

Few resources are available in the context of English to Sinhala or Sinhala to English translations. Therefore building a simple corpus which containing several sentences to train a specific word will be very useful if any other researcher in future. This corpus can be expanded time to time with words with related sentences. This will contributes to have a public corpus for Sinhala which includes set of sentences to train specific English word with its Sinhala translation.

### Computer Science Contribution

Creating and enhancing the training model which then helps to get high accuracy level with the evolution of training models.

# Research challenge

There are several challenges when doing this research. Building a training model is challenging task. In order to train a specific word for specific meaning, it needs huge collection of sentences. Not only for one word but also it needs to have several words with several sentences.

Implementing final solution as a plugin is a challenging task because it should identifies the mouse-hovered word in web content and provides the solution accurately and quickly.

# Research Questions

1. What are the major challenges when building a language model to get context sensitive Sinhala translation for a selected English word/phrase in the web content?
2. What attributes can build an accurate model?
3. What are the limitations of such a language model and how to improve?
4. How can the vast knowledge encoded in the English web be understood by native Sinhala speakers while not depriving them of the opportunity to learn English?
5. What kind of language model would best facilitate this goal?
6. How can such a model be evaluated?

# Research Aim

It is very helpful for a reader if the difficult words in the content can be easily translate to their native language. Translating the whole content to native language is not a good way if the person is willing to learn the language. Therefore user can look for a meaning based on the content without translating the entire content. The aim of this research is support for Sri Lankans to read the English based web materials by understanding the content without any hesitation. The user can read online resources comfortably with fewer distractions while reading to get the correct meaning of a difficult word in Sinhala in that content. User only needs to mouse-over the word, so the context sensitive Sinhala meaning will be display in few seconds.

# Research Objectives

The final outcome of this project which will help Sri Lankans to get the correct meaning of the word in Sinhala based on the content of the web-material. The aim of this study will be achieved by focusing on the following objectives.

* Analyze repositories available in online to get the English words and the meaning with the usage (several sentences) of the particular word.
* Building the translation dictionary which will be the main resource for this application.
  1. Collect words and their definitions in Sinhala by popular dictionaries such as MalalaSekara dictionary and other glossaries.
  2. Those Sinhala word collected from dictionary entries then stored in standard encoding (Unicode)
  3. Entries will be validate, clean and filter accordingly.
* Building the language model

1. Use appropriate machine learning algorithms which will be used to get the correct definition of the selected word and display in Sinhala.

* Implement the application, which then helps users to get the definition in Sinhala by mouse-hovering to the specific word.
* The number of similar definitions in Sinhala to an English word or phrase may vary. Therefore to improve simplicity and the readability of the proposed system it considers only one definition.
* The main intention of this application is to reduce the time spent on searching the meaning of the word and not to translate the meaning of the whole sentence.
* By giving an accurate meaning the user can then easily understand it. Therefore mostly focuses on nouns and does not consider the tense of the sentence which will complex the implementation.

# Project Scope

This application will be implemented which has following features and functions:

* Focus only for the English based web pages.
* Users can get the correct meaning of the selected English word by mouse hovering to it.
* Selected word will be display the definition in Sinhala according to the content.
* Help user to display the meaning by mouse-hovering, so that it will display the definition within few milliseconds.
* User can learn English language while enjoy the reading, and improve their vocabulary unintentionally.
* Cover considerable amount of words to build the training model.

# Resource Requirement

All the resources needed for the project are as follows.

### Software Requirements

|  |  |
| --- | --- |
| *Software Tool* | *Purpose* |
| Weka | Preprocessing the data set |
| MS Excel | Building the training model and test model in csv formats |
| STS | For application development |
| Git Hub | Repository for source code management |
| Google Drive | Storing documents and csv files |
| MS Word | Document creation |

Table 4: English to Hindi translation of a sentence

### Hardware Requirement

* Processor – intel Core i5
* Ram – 8GB
* Hard disk(storage) – 25GB

### Other skills required

* Creating a training and test model with collection of sentences.
* Preprocessing using NLP.
* Knowledge of evaluating the model with scientific methods.

# Chapter Summary

This chapter contains the detail information about the research project‟s background and the introduction. Problem domain was explained with the current limitations of the tools and techniques. Research motivations and identify the research gap, challengers the project scope and its contribution computer science and the domain was discussed. The research aim was explained with the set of research objectives. It concludes with the resource requirement which needs to develop this project.

# Literature review

There are two main areas to consider when implementing Sinhala translation tool for context sensitive words and phrases in the English language for online resources. First one is the translation dictionary and the other one is building a language model. Natural Language Processing and Machine Learning is two core concepts use for achieving areas mentioned above. Below it discusses the principles use for building the translation dictionary and the language model using several approaches, techniques and tools.

# Tools and Techniques

# Natural Language Processing (NLP)

This is recognized as a field of artificial intelligence. It will provide the accessibility to identify and understand the human language. Computers understand only the machine language. With evolution of technology, computers have the ability to understand text and spoken words in much the same way human beings can by processing large volumes of data as required. This has two directions. First one is how is the language is understand by the computer and the second one is how the machine is generate the natural language. For example, when the speech is input to the machine then it understand the phonetic, lexical, syntactic, semantic and lastly pragmatics then the machine should uses the reverse process when generating speech. NLP combines computational linguistics rule based modeling of human language with statistical, machine learning and deep learning models. Combination of these of these technologies enable computers to process human language in the form of text or voice data to understand its full meaning, complete with the speaker or writer‟s intent and sentiment. Word sense disambiguation is a NLP task where the selections of the meaning of a word with multiple meanings through a process of semantic analysis that determine the word that makes the most sense in the given context. For example, word “mine” in “this pen is mine” refers to “belongs to me” while “gold mine” refers to “place where gold can be found”. There are several ambulation exists in many levels. If there is a disambiguation is exist in syntactic level, there are several parser trees can be generated (See Figure ).

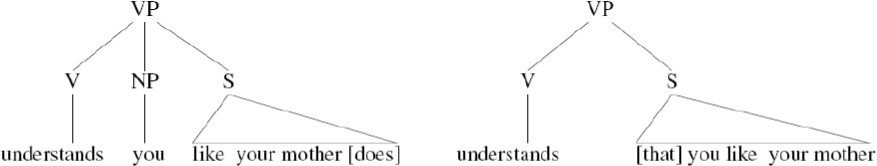


Figure 2 : Syntactic level ambiguity

Therefore because of the ambiguous situations, identifying the correct context of a text is essential. The pragmatic analysis which is the last step of language generation process which is discussed above and it is a crucial part among the NLP components to overcome the drawbacks in the translation processes. There are several applications areas using the NLP approaches.

Machine translation, text mining, Search engines like Google, text summarization, question answering systems for example IBM Watson are some of them. IBM Watson in Natural language understanding process it uses deep learning to extract meaning and Meta data from unstructured text data. Currently the chat bots are also capable of understanding and response to customers‟ issues which are raised through online applications and without human involvement chat bots can address some of the simple questions.

# Machine Learning (ML)

Machine Learning is identified as a field of study which gives the capability for a computer to learn without being programmed explicitly. It is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classification or predictions. Training dataset including several sentences require to identify the specific word meaning. Language like python and their libraries are useful for providing the statistical results regarding the accuracy of them.

# Machine Translation (MT)

This can be simply defined as a process when computer software translates text from one language to target language without human involvement. It can handle large amount of source and target languages that are compared and matched against each other by a machine translation engine. MT has ability to memorize key terms and reuses them wherever they might fit.

# Existing approaches for machine translation

Human-assisted, Rule-based, Statistical, Example-based, Knowledge-based, Hybrid and Agent- based are some of the categories to classify Machine translation systems. Most of the categories use copra for the machine translation. Therefore these approaches are names as corpus based approaches. Statistical approaches are automatic learning methods and they need less time to build MT system but they need large parallel corpus for translation and the result not guaranteed to produce 100% correct translation. Therefore Rule-based approach is better than statistical approach when considering to the output result. When compared to the computer with human being, computers cannot store all the knowledge a person gain through his/her entire life. That is a major bottleneck for this type of methodologies.

# English- Hindi /one Indian language to another Indian language

This approach uses Human-assisted methodology. This project (Anusaaraka) has been developed to translate Punjabi, Bengali, Telugu, Kannada and Marathi languages into Hindi [3]. There is an English-Hindi Anusaaraka where it translates English text into Hindi. The approach and lexicon is general, but the system has mainly been applied for children's stories. This system uses Paninian Grammar model for its language analysis [4]. This system focuses on producing more correct translation rather than giving a meaningful translation and reducing the language barrier by facilitating access from one language to another [3]. Architecture of this system contains mainly Apertium which is for initial analysis the text and Anusaaraka for further processing. The process starts with de-formatting which get only the text by removing Html tags. Analyzer tokenizes the plain text and tagging process uses first-order hidden Markov model to choose corresponding lexical forms. Then lexical transformer gets source language lexical form and sends the matching target language lexical form. These are input to chunking process which then identifies patterns of lexical forms. For each lexical form deliver its target language surface from my morphological generator and then the re-formatter save output [3]. After completing the initial process it sends to the CLIPS expert shell of the Anusaaraka for further processing.

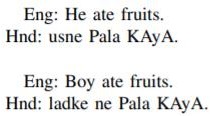


Figure 3 : English to Hindi translation of a sentence

Above is a translation from English to Hindi sentence by Anusaaraka (See Figure 3). The system architecture can be useful for two purposes. If the source language and target language are grammatically distinct, then Anusaaraka (English-Hindi) can be used. If the source language and the target language are grammatically similar then with minimum changes in target language part it can be useful for English-Telugu, English-Tamil pairs [3].

# English to Japanese language / Japanese to English language

This system is translating open-domain written text by using morphological analysis, syntax analysis, translation word selection and structural transformation and morphological generation steps. Rules based approach with common word dictionary, a technical-term dictionary and user- defined dictionary is use by this system. Common word dictionary contains both English- Japanese and Japanese-English translation. Technical term dictionary contains 28 domains including computer, machinery and medicine [5]. Japanese language doesn‟t have spaces between words therefore morphological ambiguity is present when the input text is segmented into words and phrases. Second step is the syntactic analysis where augmented translation grammar which is context-free grammar used for parsing the sentences. Semantic analyzer is use to get the accurate meaning. There are different translation words available in any languages. In Japanese transitive verb “ か け る ”(kake-ru) has different meanings [5]. This type of words should be select carefully based on the context. Lexical transfer in tree conversion is useful for this. In the final step of generation phase, Japanese word order structure is considered to map the English words. This system produces the following output.

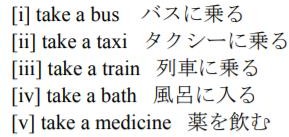


Figure 4 : English to Japanese translation

# Hindi to English language

This system (XFER) is an approach translates from Hindi to English language. System used IIIT Morpher which is the morphology module for inflect words in Hindi. It considers gender, number and tense and uses Roman-WX for Romanized character encoding for Hindi. It contains

70 transfer rules, 58 verb sequence rules, 10 recursive noun phrase rules and 2 prepositional phrase rules [6]. Except of the above manually written rules it consists of 327 rules which are automatically learned transfer rules. However this system is suitable for trained an extremely limited data scenario.

# Pali to Sinhala Language

This system is used to translate Pali words to Sinhala language which was successfully translating simple Pali sentences to Sinhala. This can be used as a learning tool. Dictionary based approached is used in this system. It is mainly combination of three core components namely; Dictionary based translator, Pali morphological analyzer and the Sinhala morphological generator. Pali morphological analyzer is getting an input as Pali word and it shows the grammatical information and the root word. It is the main component among three component listed above. An affix spiriting approach is supports to identify the relevant root word in Pali.

Pali dictionary consists of irregular words and it developed the root word indentation table. With the support of Pali Sinhala dictionary, Pali to Sinhala translator identifies based word for existing Pali word. Semantic issues and word level ambiguity are neglected and it shows an error message if the word is not available in the dictionary. The final phase of the model is the Sinhala morphology generator. Sinhala based word is generate appropriate Sinhala word by Sinhala Morphological generator. Limited number of rules used to generate Sinhala words. This system is not applied syntax level generation because it said that Pali and Sinhala languages are closely related to each other. This system is beneficial for the Buddhist monks in Pirivenas and any student who wish to learn Pali language specially in Dhamma schools.

# English to Arabic language

This system used a mapping system for Arabic to intermediate representation. This mapping system contains three steps namely; selecting lexical items for each Interlingua concepts, mapping the semantic roles and mapping the semantic features for each Interlingua concept to appropriate syntactic feature in the feature structure. It has been test on 29 different structures and has produce good results [7].

# Hindi / Bengali to English language

Dictionary-based machine translation approach is the main methodology to develop this system where it uses cross language retrieval in the system. The queries in Bengali and Hindi translate to the equivalent English query out of Indian language topics. Phonetic translation system is used to overcome the limited coverage dictionary. 26,000 Hindi words and 9,000 Bengali words are used to build the dictionary [8]. According to that it is not considered parts of speech information because of the dictionary limitation, improper stemming and the term is foreign word or a named entity [8]. It stated that the Hindi word “rokanA” (to stop) has 20 translations which makes the average English translation per Hindi word in the lexicon were 1.29. That is because 14.89% of Hindi words have several translations [8].

# Babel Fish

This is a web based application initially called by AltaVista developed to translate single text, phrase or web pages (by proving the url) from one language to another. It can translate 13 languages and English, Chinese, German, Greek, Italian, Japanese, Korean, Portuguese, Russian and Spanish are some of them [9]. This system use statistical machine translation approach. This approach works by analyzing parallel corpora that have already been translated from one language to another. For example, if it search “un perroquet rouge” in French every time “a red parrot” occurs in English. Then it stores these two phrases together in a “phrase table”. It also analyses large amounts of text in individual languages and memorize the frequency that certain words or phrases follow others. Then it build the language model identify different meanings of words when the input word have several meanings.

# Approach for the project

In the modern era machine translation has received much attention because the decrement in time and human effort required for converting and translating words, sentences and paragraphs. Thus as discussed above there are several ways for machine translations. Each and every system described above uses different types of mechanism with different size of word counts to train the model. Most of them have been focused on Indo-European, Indo-Aryan or Sino-Tibetan families [10]. To get accurate result it should contains larger words collections. But that is not practical.

For this project there are two tasks to be accomplished. Creating a translation dictionary and building a language model are two of them.

## Translation Dictionary

This is the building block for the proposed system. This system is going to implement as an extension to the “EnSiTip” [18] and it used Sinhala to English translation dictionary. It is constructed by gathering words and their definitions from the Malalasekara and other popular glossaries represented in standard encoding (Unicode). Those are formatted, validated, cleaned and filtered accordingly by group of experts. The user friendliness is enhanced by reducing the number of Sinhala definitions corresponding to each English headword. In order to achieve that the Occam's Razor principle was used.

### Occam's Razor principle

This principle states that “Entities should not be multiplied unnecessarily”. There may be several theories which are competing for the same prediction, therefore, take the simplest one which makes everything better. William of Occam is the person who developed this concept and razor denotes that cutting off or shaving away the other possibilities.

Example 1, Assume in a rainy day corner of the room spilled out some water. 1st possibility with high probability: Should be leaky roof.

2nd possibility with less probability: Kid who spilled water in the room.

According to the above principle it omits fewer probabilities which help to make it simple rather than complex. That means water spilled because of the rain and the leaky roof.

Example 2, Assume car tire is flat when getting ready to leave.

1st possibility with probability: A nail stuck in the tire wall let the air out 2nd possibility with less probability: Someone slashed the tire.

According to the principle, it is more likely the tire gets slashed by nail.

By following the Occam‟s razor principle omit unnecessary meanings which make this very complex. Therefore according to that EnSiTip‟s translation dictionary is covered nearly 50,000 English words to produce better result for user. This translation dictionary was converted in a way which can be useful to the proposed system. There are many English words has different meanings based on the context of the sentence. Then the below translation dictionary was developed. There are more than 36000 English words are listed there.

# Language model

## Probabilistic model of language

It is attempt to characterizes, capture and exploit regularities in natural language. Determine the probability of a word in a sentence by analyzing the bodies of text data by using statistical and probabilistic approaches. This is more specifically called as statistical language models. Large amount of words are used to automatically determine the model‟s parameter in this statistical language models. Speech recognition, machine translation, context sensitive spelling correctors and next word prediction are the usage of this. It assigns probability to every word sequence which may be grammatical or not.

P [W1 W2 W3 … Wn ].

Related: P (W5 | W1, W, W3) implies that conditional probability of W5 after W1, W, W3.

### Bayes Rule/Chain Rule

P(X1,X2,X3,…,Xn) = P(X1)P(X2|X1)P(X3|X1,X2)…P(XN|X1,…,Xn-1)

For example,

Sentence: I like red apple.

P(I like red apple) = P(I) P(like | I) P(red | I like) P(apple | I like red). To estimate probability for text corpus,

P(I) = count (“I”) / Total # of words

P(like | I ) = count (“I like”) / Count (“I”) and so on.

## N-gram language model

In the basic concept of the Natural language processing N-gram is using for the applications to build language models. For example this N can be 1, 2, 3, etc.. and according to that it defines as Uni-gram, Bi-gram, Tri-gram etc. This language models use some number of preceding words to makes predictions.

Simplest approximation: unigram

P(“I like red apple”) = P(“I”) P(“like”) P(“red”) P(“apple”). Bigram

P(“I like red apple”) = P(“I”) P(“like” | ”I”) P(“red” | ”like”) P(“apple” | ”red”).

The size of the N should be considered based on the situation. Having N > 3 will give more accurate results. But the complexity is exponentially growing with the increase of N value. Data sparsity problem is common issue when handling the local dictionaries / vocabularies because there will be more missing values and not able to provide accurate results. Therefore Recurrent Neural Networks use to address the above issue.

## Recurrent Neural Network (RNN)

RNN can handle inputs having variable lengths. It is suitable for modeling the sequential data such as sentences in natural language. RNN contains loops, therefore same network copies again and again until it reaches to the successor. Figure 5 shows the RNN with its loops. “A” is a chunk of neural network which take the input “Xt” and produce output “ht” and the loop continuously pass the information from one step of the network to the next. With this loops it is same as having several copies of the same network.

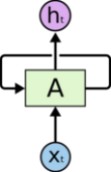


Figure 5 : Recurrent Neural Network

As discussed above RNN network can be unfold. Figure 6 illustrate how the above RNN network is unfold to multiple copies of the same network.

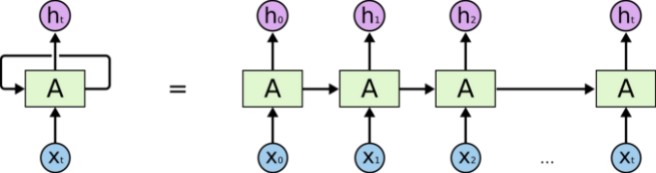


Figure 6 : Unfolded Recurrent Neural Network

Because of these networks have chain like nature they have similarities to sequences and lists which are available in data structures. Currently, there are several researches and systems are built on top of this technique. Speech recognition, language modeling, translation and image capturing are some of the areas that RNN can apply. As above explained the RNN depends on the previous state, but with the long sequences grow over period of time it cannot handle larger among of data because the information which learnt by RNN will decay. This issue is called the “vanishing gradients problem”. Because of that the final result is not very accurate. Therefore several extensions were introduced to address this issue.

## Long Short Term Memory (LSTM)

This technique is type of a RNN which address the issue of “vanishing gradient problem” in the RNN. LSTM can remember information for longer time period by keep track of previous events. It follows the same structure of RNN with introducing four new layers. Figure7 and Figure 8 show the internal structure differences of RNN and LSTM network.

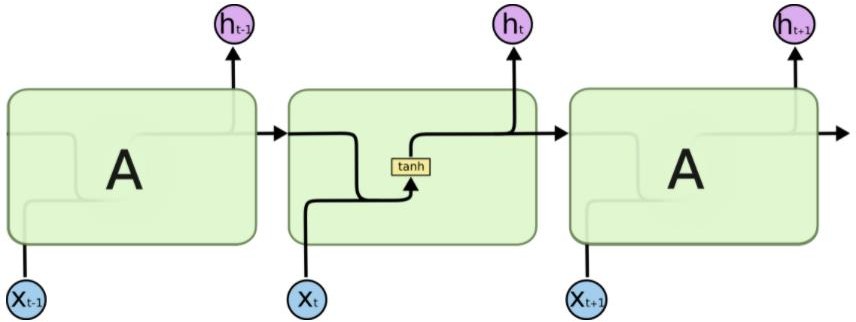


Figure 7 : Internal structure of RNN

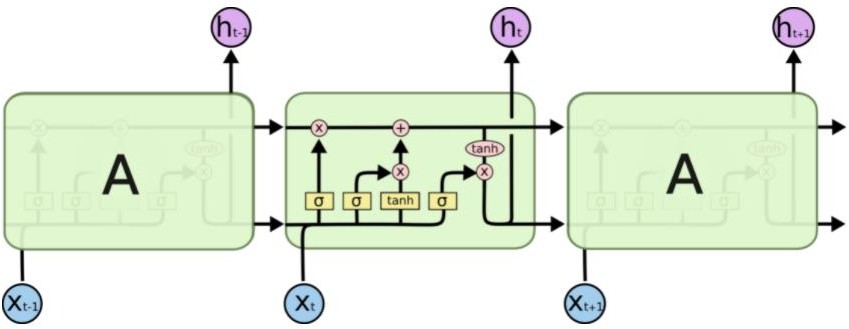


Figure 8 : Internal structure of LSTM network

The symbols of the above two figures are denoted in the Figure 9.

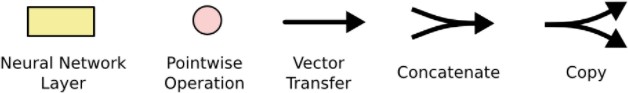


Figure 9 : Symbols used for internal networks

According to the Figure 7 the RNN structure has only one layer in its repeating module. But Figure 8 illustrates that the LSTM contains four layers in its repeating layer.

The internal structure and the behavior of the LSTM network are explained below with an example.

For a language model, if it wants to predict the next word for a sentence;

I grew up in China and I spent most of my life there but currently live in London. Therefore I speak fluent in <….> (Chinese).”

The first step of the LSTM network is to identify which information is going away from the cell state. “forget gate layer” is a sigmoid layer which make this decision. The output will be either 1 or 0 where 0 denotes by “completely get rid of this” and 1 denotes by “completely keep this”.

According to the example the cell state may include the gender of the present subject. Next step decide which new information needs to store in the cell state. This contains two parts; “input gate layer” which decide which part to be modified and second part is creating a vector for the new candidate values. According to the example in this step it adds the gender of the new subject to call state to replace the old one. After it updates the old cell state with the new cell state it actually drops the information about the old subject‟s gender. In the final step, decide what to be taken as output. According to the example it might output if the subject is singular or plural and decide the form of a verb should be produce.

As described above the process is continues and it remember values over arbitrary time intervals. Therefore LSTM is more accurate than RNN results.

## Bidirectional Encoder Representation from Transformers (BERT)

This is used for keyword, semantic search to retrieve information and to produce vector based inputs from the words which will be uses in the Natural Language Processing models. It uses transformer which learn contextual relations between words in a text [22]. Internally this transformer contains two separate mechanisms namely; an encoder to read the input text and a decoder produce a prediction for the task. For the language model only the encoder is necessary. For example, if the BERT is using then for the sentence “The man was accused of robbing a bank. The man went fishing by the bank of the river”, the word embedding for “bank” would be different for each sentence [22].

BERT uses mainly two strategies pre-training and fine tuning. Pre-training strategy contains two tasks. They are Masked Language Model (MLM) and Next Sequence Prediction (NSP). As name implies in the MLM it is to mask some percentage of the input token randomly and predict them. Example, “I am eating an apple”.

* 80% of the time: Replace the word with the [MASK] token e.g., I am eating an [MASK].
* 10% of the time: Replace with random word e.g., I am eating an orange.
* 10% of the time: Keep the word unchanged, e.g., I am eating an apple.

The other task mentioned in the pre-trained strategy is Next Sequence Prediction (NSP). In this method more than one sentence can be combined in a systematic way where it start the sentence with “CLS” and separate the sentence using “SEP”. This approach is used to identify the relationship between the sentences.

For example,

* Sentence 1 -> “Kevin went to the supermarket”.
* Sentence 2 -> “He bought bread and milk”.
* Sentence 3 -> “Environmental pollution is increasing day by day”.

Using this method it will identify the sentence 1 and 2 are closely relate with each other while sentence 1 and 3 or 2 and 3 are not having any relationship. Second strategy in BERT is fine tuning. Fine-tuning is straightforward since the self-attention mechanism in the Transformer allows BERT to model many downstream tasks whether they involve single text or text pairs by swapping out the appropriate inputs and outputs. As discussed above there are many techniques and tools available and they are used in different approaches to build several systems. In early days B-gram language models used with immerge of the technologies latest trends are use. For this project BERT can be applied by training words using collection of sentences [23].

There is an application built to identify the right meaning of the word “duck” where it contains four separate meanings based on the context and for the simplicity it categorized the verb “duck” and corresponding nous as the same meaning.

1. Bird / flesh of the bird „duck‟.
2. Lower head or body suddenly.
3. Durable closely woven usually cotton fabric.

The dataset contains 77 sentences contains the word “duck” and there are 50 sentences identified as animal (type 0), 17 identified as verb (type 1) and 10 identified as fabric (type 2). With the manual identification of the word “duck” and the actual was compared. The following figure shows the result gained from using BERT and the translation from the English-Hungarian translator.

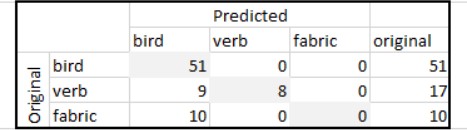


Figure 10 : Expected vs. Actual for the "duck" word

According to that the bird type is correctly identified and the 8 sentences are correctly identified as verbs. All the fabric type is incorrectly identified as bird. It produces 75.641% accuracy [24].

# Related work

# Approaches for English to Sinhala translation

In Sri Lanka there are several universities contributed to develop machine translation system for Sinhala and Tamil languages. Sinhala Corpus [11], Parts of Speech Tagger [13], Optical Character Recognition system for Sinhala language [14] and Sinhala text to speech system [12] are some of them. The corpus based approach for Sinhala to Tamil machine translation system provided reasonable results for the evaluation by BLUE score [15]. Moreover, many prototypes are developed for English to Sinhala machine translation. English to Sinhala translation system for weather forecasting have developed. It can translate simple sentences and works on the limited set of words with limited sentence patterns. This is a rule based and it has used paragraphs and sentence tokenization, simple parses, translators and Sinhala sentence generators for English to Sinhala translation [16]. Bilingual Expert for English to Sinhala is another system which used rule based approach for implementing the system. It handles the language primitives such as person, gender, tense, number, preposition and subjectivity or objectivity. Moreover, it allows deriving all associated words from a given base word and thus it reduces the size of the Sinhala dictionary. Apart from that four lexical dictionaries are used. They are Sinhala dictionary, English dictionary, English-Sinhala Bilingual dictionary and concept dictionary [17]. It contains seven modules which illustrates in Figure 11.

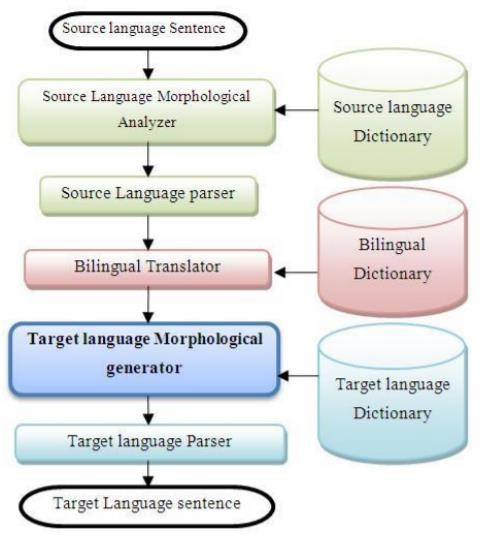


Figure 11 : Generalize version for English to Sinhala translation

EnSiTip, is a word based English-Sinhala translation tool to support users who have less literacy level of English language to understand the web content [18]. It is a Firefox add-on and a user- friendly tool. Users can mouse-hover to a word which then appears a popup displaying the possible suggestion in Sinhala language for that particular word. Building the English-Sinhala dictionary was the core component for this system. It automatically de-inflects verbs and adjectives and from the tool user also capable to listen to the pronunciation of the English word. This system covers nearly 50,000 English words. According to the statics provided, there were 411 active daily users [18]. Figure 12 shows how the EnSiTip will display the Sinhala meaning when mouse-hover to the word.



Figure 12 : EnSiTip result from a web page

There is another system is developed to get the Sinhala translation for selected text. This system is built as an updated version of the English to Sinhala machines translation system; BEES [17].

This new system expanded the designed into three modules as BEES client, BEES server and BEES translator [19]. When the user selects the word BEES client read the highlighted text and send it to the BEES server. This BEES server contains BEES translator which provides the meaning for the selected word. Three lexical dictionaries used for this system. They are English, Sinhala, and English-Sinhala bilingual dictionary. Experimental result shows that this system works with more than 80% accuracy and the human support is needed to test the system. The BEES client is capable to read selected text from document in any format such as word, pdf and html. Incomplete or incorrect text selection is an issue reported in the system because the rule- based top-down parser is less efficient to handle incomplete sentences than the grammatically correct complete sentences. Therefore system needed to improve to handle those issues.

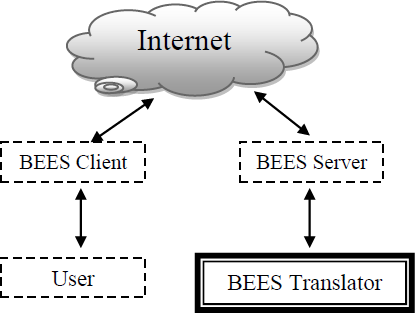


Figure 13 : BEES high-level architecture

Some of the works related to this project are discussed in the above. Currently, there is no exact system developed to get context sensitive Sinhala translation for selected word in English language. The proposed system is going to implement as an extension to the EnSiTip [18]. Using the proposed system users can simply mouse-hover to a specific word and view the context sensitive meaning easily.

# Chapter Summary

In this chapter contains existing work, systems, tools and techniques which will discuss in order to justify the concepts which will be use in the proposed system. Detail of each system highlighted the main concepts behind each system. Most of the existing works are based on dictionary approaches, rule based or hybrid implementation with their limitations. These limitations will be helpful to improve the proposed system which will be implement using machine learning approaches with suitable training model and Natural language processing.

# Problem analysis and Methodology

This chapter is explained how the research problem is analyze and handle with the knowledge gain by the literature review. In the literature review mentioned there are several implementations for translating the whole English sentences to Sinhala language and EnSiTip is showing how the selected English word is translated to Sinhala word. As mentioned earlier the selected English word may contain different definitions based on the context of the sentence. So according to the EnSiTip implementation it showed the several Sinhala words (definitions) for selected English word. Because of this problem this research is based on newer approach to achieve the goal which means for selected English word it will show a relevant Sinhala word based on the content of the sentence. This chapter provides a comprehensive overview of implementation steps which have been carried out during the project to make it successful.

## Representation of the Problem

The aim of the study is to find a solution to display correct Sinhala word (definition) of a selected English word in web content. Collections of sentences are trained for specific English word which may have several classed according to the number of meaning of that specific word. This will be using machine learning algorithms to classify the sentences for correct classes. To classify the sentences according to the specific English word it uses BERT embedding. As for the initial step it needs to build a translation dictionary. The solution mainly contains two phases. First phase is to train a classifier to classify Sentences to the specific English word. And the second phase is to display the correct Sinhala word (definition) of the selected English word in the sentence.

# Proposed system architecture

Figure 14: Proposed system architecture

The application has three separate sections. The initial step of the system is the user mouse-hover to a word in web content. Then the mouse-hovered word and the respective sentence feed to the classification model. Inside the classification model it contains language model. Language model is attempt to characterizes, capture and exploit regularities in natural language and determine the probability of a word in a sentence by analyzing the bodies of text data by using statistical and probabilistic approaches. Internally the model is trained with the training dataset and the classifier file which used to identify the class of the word. The trained model is trained with the help of BERT. In the preprocess step of BERT it uses that every embedding contains three types of embedding namely positional, segment and token embedding. BERT learns and uses positional embedding to express the position of words in a sentence. Token embedding is the embedding learned for the specific token from the WordPiece token vocabulary. For a given token, its input representation is constructed by summing the corresponding token, segment, and position embedding. With the highest probability the relevant class type is identified and with the help of translation dictionary the respective Sinhala definition mapped and it will send back to the browser plugin of which then display the correct definition with the use of a popup box.

## Translation Dictionary

According to “EnSiTip” it created “Sinhala/English translation lexicon” which is having list of Sinhala word with their English meaning (See Figure 14).

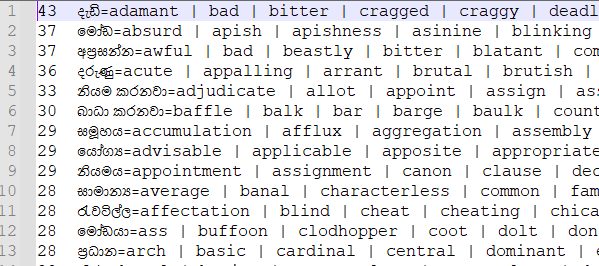


Figure 15: Collection of Sinhala to English meanings

In the above Figure the first index indicate the number of definitions for the Sinhala word “ද ”. Each different English word which has the same definition “ද ” are separated with “|” symbol. This translation dictionary was converted in a way which can be useful to the proposed system. Then created a translation dictionary which having index as number of English words for that English word, English word and the collection of Sinhala definitions relevant to each English word. Then the below translation dictionary was developed. There are more than 36000 English words are listed there (See Figure 15). Entries will be validated and cleaned manually. Finally filter and sort according to the alphabet order by making the English word as an index.

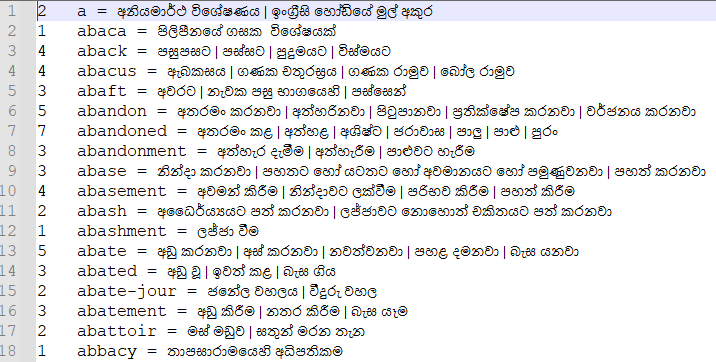


Figure 16: Collection of English word with Sinhala meanings

According to the above Figure the English word “abashment” one definition in Sinhala while “abandon” contains five Sinhala definitions. According to the content of the sentence the relevant Sinhala word may use. These indexes help to identify the number of classes which need to specify when building the training model.

## Language Model

Language model is attempt to characterizes, capture and exploit regularities in natural language. Determine the probability of a word in a sentence by analyzing the bodies of text data by using statistical and probabilistic approaches. This is more specifically called as statistical language models. Large amount of words are used to automatically determine the model‟s parameter in this statistical language models.

## Training Model Creation

Most of the implementations described in the literature review are purely based on the NLP. Therefore no training dataset is available for this scope to train the model. According to the approach that use for this research, it is mandatory to create a proper datasets for specific English words which have several Sinhala meanings. Training dataset needs to be carefully categorized into relevant classes which make the subclass of specific word. In English Language some words have several different meaning in the sentences which described in the introduction chapter and those words are called homonyms. Different homonyms contain different number of classes.

Therefore apart from the training model it needs another model to have all the different classes names with respective the specific word we train. In the test model or in the final output user can select different word of the sentence therefore the specific word also needs to be implemented as dynamic value.

# Initial Classifier

For a sentence it contains several numbers of words. Each word of the sentence are belongs to the homonyms. But when considering the web page it may contain more homonyms. Therefore in this study, first take one homonym as a user selected word and identify the number of classes or the number of meanings that word contains which will be used to train the classifier. The created dataset consist with five main classes (see Figure 16). This collection of dataset will be refereed as initial dataset in upcoming sections.

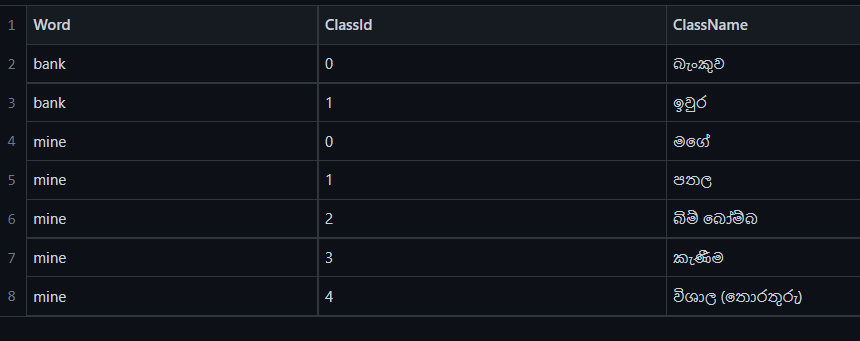


Figure 17: Different classes for specific words.

According to the above dataset it mainly contains two specific words they are “Bank” and “Mine”. With relevant to “classIds” it has relevant “className” based on the selected “Word”. These “classNames” are used to display and evaluate the system. The “ClassId” staring with the “0” it contains number of “ClassId” based on the number of definitions for specific word.

Consider the word “Mine” for building the initial dataset. These meanings are classified according to the Oxford dictionary. The author believes it is best and correct approach of classification needed as the output of the selected word. Since human involved in classification of the dataset that use for train and test the model, it can evaluate the approach in a way that output results against the human classified accurate results.

When selecting a homonym as described in the introduction it may contains many different definitions. As mentioned in the above, after building the translation dictionary can identify the number of classes which can be map. But if we carefully look at the created translation dictionary it contains several meanings which complex the implementation (See Figure 17).

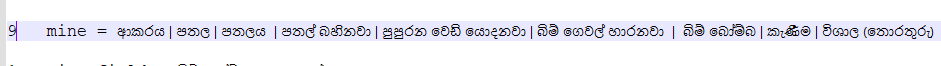


Figure 18: Different meaning for the word "Mine"

Therefore for the simplicity and readability few factors have been taken into consideration. These factors will be described in below.

Factor 1: Remove some of the unnecessary meanings.

According to the above Figure 18, Word “Mine” considering the meaning (an excavation in the earth for extracting coal or other minerals) has several similar meanings in Sinhala Language.

|  |  |
| --- | --- |
| Word | Definition in Sinhala Language |
| Mine | ආ , ත , ත , ත න etc… |

Table 5 : Different definitions in Sinhala to the word "Mine"

Therefore remove the unnecessary similar meanings from the translation dictionary. Factor 2: Consider only one selected Sinhala definition

For example, the word “Mine” considering the meaning (an excavation in the earth for extracting coal or other minerals) it will only shows the Sinhala word “ ත ”.

|  |  |
| --- | --- |
| Word | Definition in Sinhala Language |
| Mine | ත |

Table 6: One definition highlighted for the word "Mine"

This will improve the readability of the user rather than having several similar definitions like showed in table 5.

Factor 3: Focus areas of the words

This model mostly focuses on the nouns, pronouns and verbs. Therefore it is not consider about the tense of the sentences.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sentence** | **Tense** | **Word** | **Definition in Sinhala** |
| I usually run every day. | Present Simple | Run | න |
| I have never run professionally. | Present Perfect Simple | Run |  |
| I think I'll have to run for the bus. | Future Simple | Run | න න |

Table 7: Different definitions of Word "Run" based on tense

Because of the complexity, it is not consider about the tense of the sentence, therefore when user select the word “Run” it will show the Sinhala definition න (See Table 8).

|  |  |
| --- | --- |
| **Word** | **Definition in Sinhala** |
| Run | න . |

Table 8: Selected definition for word "Run"

# Initial Training Model

For Sinhala Language it is difficult to find training model for this purpose. Therefore with the help of online dictionaries and resources build the initial training model. As an initial training model, created collection of 120 sentences for using the word “mine” in the sentence which include the different definitions as listed in the Figure 16. yourDictionary is one of the greatest resource to get collection of sentences when building the training model for this kind of scenario. After collecting these sentences, carefully identifies the meaning of the word “mine” in each sentence. These sentences then map with the “classId” according to the “class name” in Figure

16. It contains sentences with the following distribution: 37 are referring to the pronoun (Type 0), 59 are a form of the “ ත ” noun (Type 1) and 10 are referring to the “ ” noun (Type 2), 10 are referring to “mine” (verb) (Type 3) and lastly 4 are referring to idiom (Type 4).

In this training model it needs another additional field. That is the “Word” which needs to specify from which word in this sentence it needs to train the sentence. The initial training model will illustrate below (see Table 9).

|  |  |  |
| --- | --- | --- |
| **Sentence** | **Type** | **Word** |
| "It was safer to leave him in the mine." | 1 | "mine" |
| "It was a gold mine, wasn't it?", | 1 | "mine" |
| "So I say the horses and chickens are mine and Alex says the other animals are  his." | 0 | "mine" |
| "You may bring mine with you.", | 0 | "mine" |
| "You see, when you die, you have your heaven and I have mine.", | 0 | "mine" |

Table 9: Initial training model

These files saved as comma separated value (CSV) files with the UTF-8 encoding which then easily readable from the Python language. Most of the machine-learning applications are use Python language to train the model because of the powerfulness and the efficiency. Therefore this system is implemented using the Python language. In order to train the model first need to import the relevant Python libraries (See Figure 19).



Figure 19: Installed and imported Python libraries

“pandas” and “numpy” are some of the basic Python libraries which useful when developing the Python applications. The libraries are capable of manipulating high dimensional data and

analyses while offering data structures and operations in order to manipulate numerical values. As discussed in above chapters the place of a word in the sentence need to be figure out in order to get the correct definition of that word. To train the sentences in the language model developed earlier for specific word, it uses BERT which considers the relationships by statistical approach to predict the correct definition. Therefore this application uses BERT then it will consider both left and right context of the word before predicting the definition. Scikit-learn (Sklearn) libraries are used for this system because it contains lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. For this system it requires to classify the sentence to correct classes and mataplotlib library uses for visualize the training model.

Once the required libraries installed then needs to input the training model to train. Created CSV files are easily read from the application when they available in public repository like GIT. These files then read the following Python codes (See Figure 20) and store it in a variable which later part of the application then needs to access.

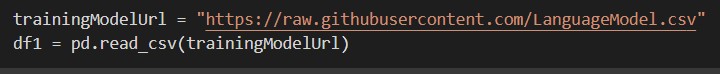


Figure 20: Read training model

Using the “print” command it will display the content of the variable “df1” which it read from the CSV file. The file contains mainly three columns and 120 sentences with the class and the training word respectively (See Figure 21).

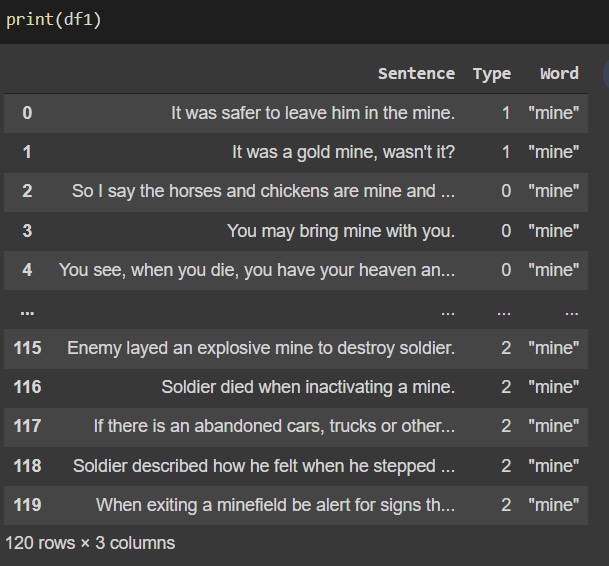


Figure 21: Content of the training model

Then import the “BertEmbedding”. In the constructor of the BertEmbedding can specify the max sequence length which denotes the target length of our encodings (See Figure 22).

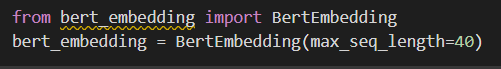


Figure 22: Import BERT model

For large piece of text max\_seq\_length property is need to break the text into small chunks (See Figure 23).

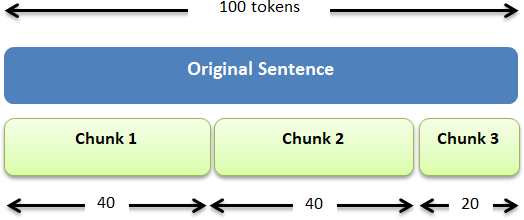


Figure 23: Chunking mechanism in BERT

According to the Figure 21 it illustrates that the model contains 3 columns. Those are “Sentence”, “Type” and “Word”. Using the column name “Sentence”, get all the values of “Sentences” and feed them to the BERT model to tokenize the sentences.

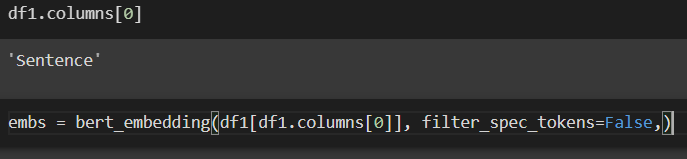


Figure 24 : Embedding the sentences

The output of the above coding (See Figure 25) will display the list of tokens, and tokens embedding.

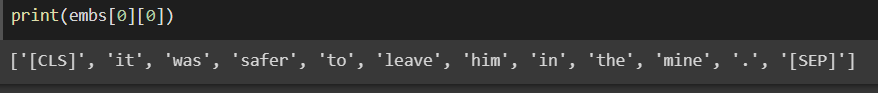


Figure 25: Tokens and token embedding for the sentence

When training large sentences the model itself needs to figure out the sentence starting point and the ending point. Each of the sentences in the CSV file was tokenized by appending the [CLS] and [SEP] tokens. [CLS] indicates the starting point of the sentence and which will appears at the start of every sentence and [SEP] indicates the separate which appears at the ending of each sentences. These tokenized sentences then need to train using the specific word by referring the “Word” column of training CSV file. Using the variable we used to store the file content can retrieve the “Word” column value in each sentence (See Figure 26).

Using that word then create the array of tokenized by identifying the index of the specific word in the sentence. Then it the model only the embedding for the „mine word‟s token.

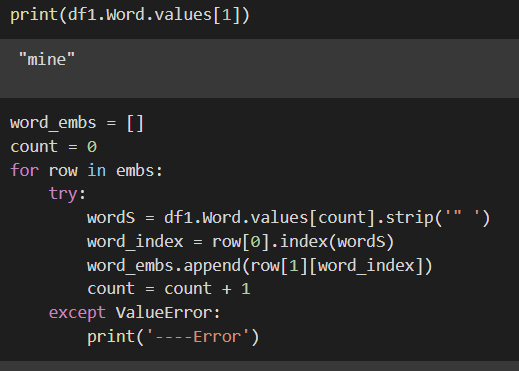


Figure 26: Train the model with specific word

Convert the generated embedding to an array and reshape it using the PCA (See Figure 27).

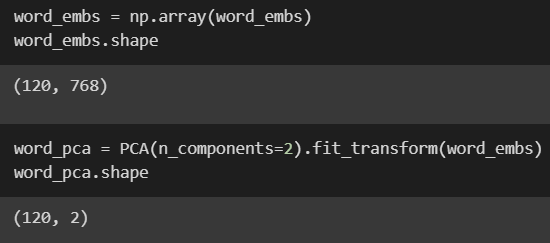


Figure 27: Reduce the dimensions using PCA

Principal Component Analysis (PCA) is an orthogonal transformation that use in the model to reduce the dimension of the vectors. Using the BERT base uncased model use the last hidden layer which generates 768 size vectors for every word. That is huge number and with the support of PCA it can project 768 dimension vectors to a 2 dimension form where the human can easily visualize the data. PCA keeps the maximum possible variance which means the projection loses

information when reducing 738 dimension to 2 dimension, therefore it might keep enough variance that helps to identify the classes on the plot.

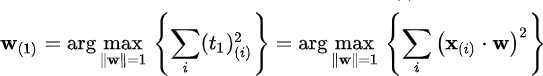


Figure 28: Maximum variant for the first principle component

Once it reduce to 2 dimension the distribution can be plot easily using the below Python code (See Figure 29).

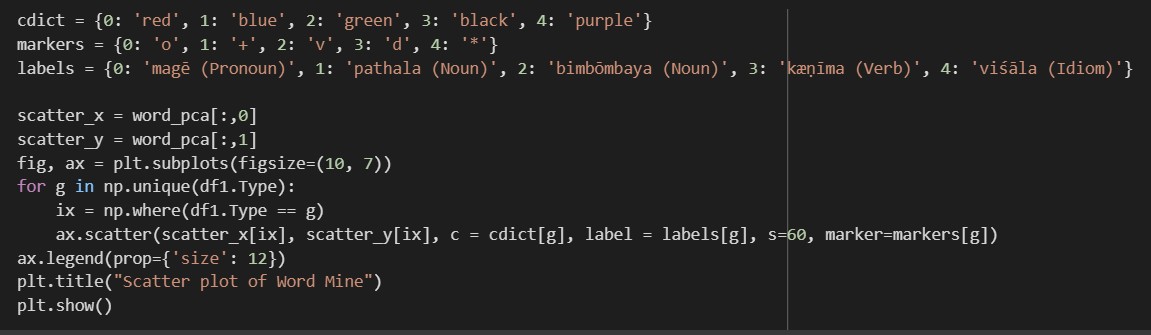


Figure 29: Plot the training model

According to the initial classifier build for the word “mine” contains five different classes. Therefore the number of makers and class labels with colors can initialize. The Figure 31 shows the result of the projection using the first two principal components. The classes are manually annotated types according to the classifier. It clearly illustrates the type pronoun and type idiom easily separated from the others.

Since the dataset is small, k-NN classifier can be used because it uses the k closest samples to predict the class of a new sample. K-NN selects the most represented class from the neighborhood. Leave One Out Cross-Validation (LOOCV) is use for validating the model. As name implies it take 1 sample out of 120 samples and the remaining 119 which will validate using the single sample. Accuracy can be generated using the results obtained (See Figure 30).

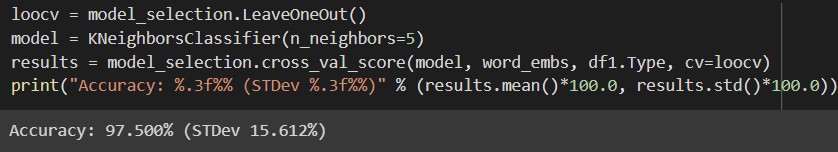


Figure 30: Accuracy of the training model

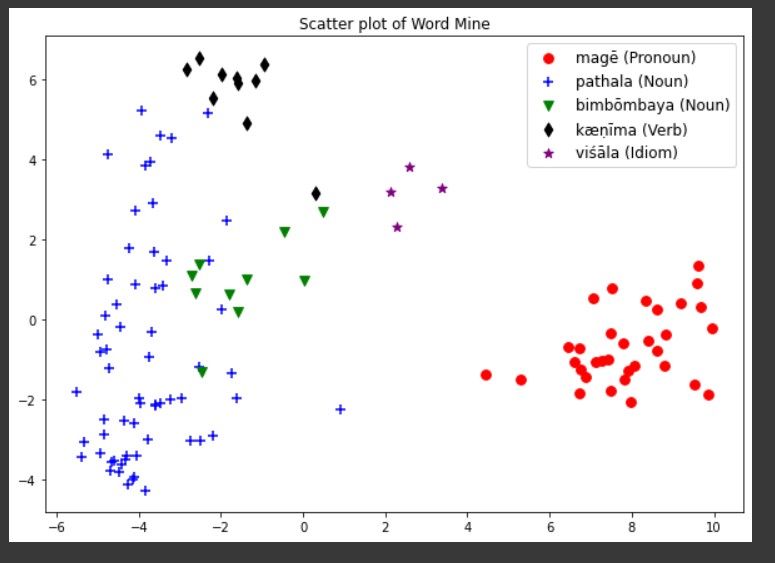


Figure 31: Scatter plot of the word "Mine"

# Test Model

Test model can create as same for the training model by having new sentences which are not included in the training model and most importantly without specifying the class name for each sentence. Indicating the “Sentence” and “Word” as column names, create a simple test model with six sentences (See Table 10).

|  |  |
| --- | --- |
| Sentence | Word |
| "Maybe you should borrow mine.", | "mine" |
| "In Queensland there is one mine 3156 ft" | "mine" |
| "The research of a mine in no way impairs the rights of ownership of the land in  which the mine is located." | "mine" |
| "These papers by leading experts in the respective fields provide a mine of  information that will be referred to for some time to come" | "mine" |
| "They mine a lot of copper around these parts." | "mine" |
| "It was your decision to go, not mine." | "mine" |

Table 10: Test Model

The same process of the training model can use to predict the class name of each sentence. Each tokenize sentence can display as in the Figure 32.

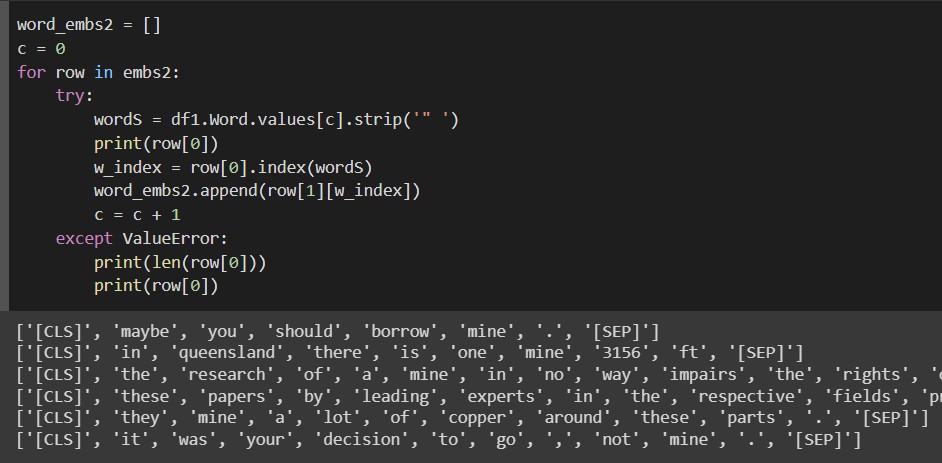


Figure 32: Tokenized sentences for the test model

The variable “model2” indicate the test model which needs to display the predicted class of each Sentence. The score of the prediction can be view as below (See Figure 33).

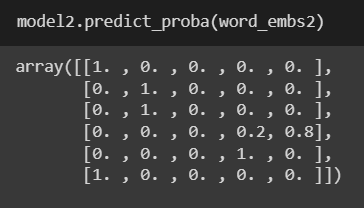


Figure 33: Sentences showing the probabilities for each class

According to the results generate, in the fourth row; the model is 20% predicting that sentence is belongs to “Type 3” and 80% predicting to “Type 4”. Final value is based on the highest probability which is “Type 4”. The predicted class for the sentences in testing model is shown below (See Figure 34).

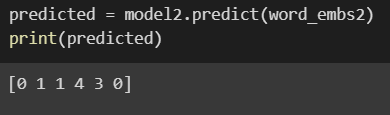


Figure 34: Predicted classes for test model

Using the classifier file developed above can display the class name which is more readable.

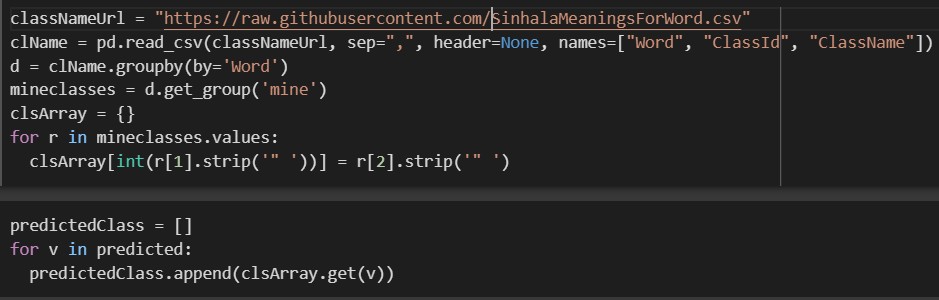


Figure 35: Code to read classifier to show the class names

The results of the test model display as below (See Figure 36). According to the sentences in the test model all the six sentences are correctly classify to its specific class.

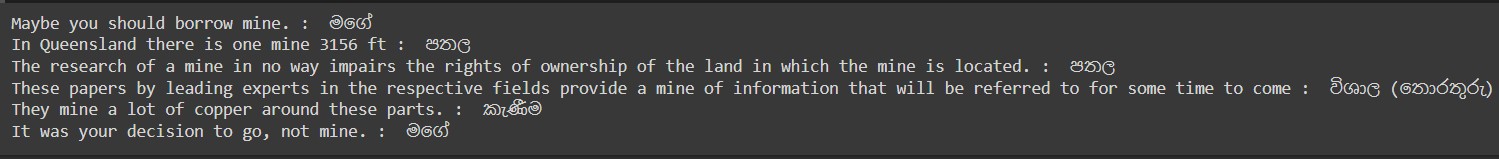


Figure 36: Output of the test model

BLEU score can be measured for test model based while having the training model as reference.

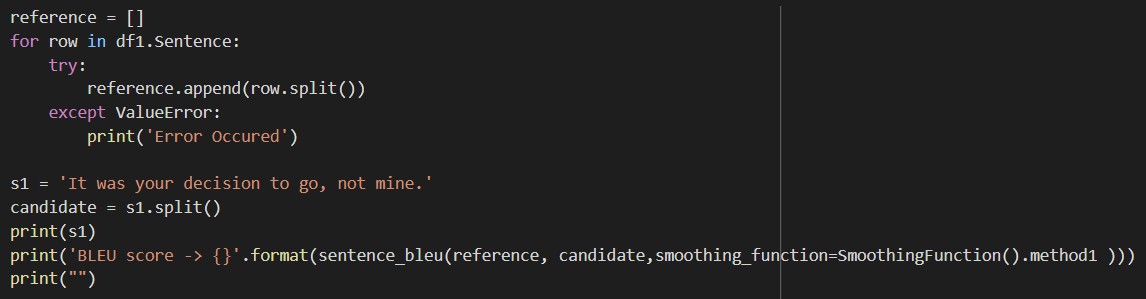


Figure 37: Code to display BLEU score for the sentence

The BLEU score indicating is 0.09193 on Figure 38 which illustrates that the exact sentence is not included in the training model and the test model correctly identifies the class accordingly.

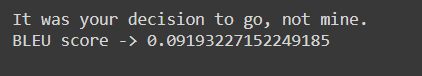


Figure 38: BLEU score for the sentence

As discussed above the initial training model is created with the word “mine” and the training model can add sentences which train for another set of words. Same as for the initial classifier can add another set of words by identifying different classes of that word. Application can train the model and implement it as a browser plugin. Then the plugin will display the correct Sinhala definition when user mouse-hover to the English word in web content.

# Chapter Summary

In this chapter contains problem analyses and the methodology which will discuss in order to provide how these techniques are used the proposed system. Detail of the initial model of the system highlighted the concepts behind each model. The proposed system which is implemented using machine learning approaches with suitable training dataset to train the model and the how the BERT which specifically developed for this purpose is explained.

# Project Plan and Timeline

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Research activity** | **June** | **July** | **Aug** | **Sep** | **Oct** | **Nov** | **Dec** | **Jan** | **Feb** | **Mar** | **Apr** | **May** | **June** | **July** | **Aug** |
| Selection of Supervisor & Tentative Project Title |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Formulated of the research problem, preparing draft research proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Final Updated project Proposal  #PR1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Draft Introduction, Literature Review Chapter #PR2 |  |  |  |  |  | Exam break |  |  |  |  |  |  |  |  |  |
| Interim Report Submission #PR3 |  |  |  |  |  |  |  |  | Exam break |  |  |  |  |  |  |
| Demonstration 1  #PR4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Evaluation Plan #PR5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Demonstration 2  #PR6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Draft Thesis  (Supervisor Version) #PR7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Thesis #PR8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Final Defense |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# References

1. "Usage statistics of content languages for websites", “w3techs.com”, [https://w3techs.com/technologies/overview/content\_language.](https://w3techs.com/technologies/overview/content_language)
2. A.B.I.K.Premasiri, “A Comprehensive Study on Sinhala and English Verbs”, International Journal on Studies in English Language and Literature 2018, 6(9), pp. 43-57
3. S. Chaudhury, A. Rao and D.M. Sharma, “Anusaaraka: An expert system based machine translation system “, Proceedings of the NLP-KE International Conference 2010, Aug 21-23, pp.1-6
4. Bharati and et al, “Natural Language Processing: A Paninian Perspective”, 2002.
5. I. Tatsuya, K. Akira, K. Yuka, “Toshiba Rule-Based Machine Translation System”, NTCIR-7 PAT MT, Proceesings of NTCIR-7 Workshop Meeting, Japan, 2008, pp.430-434.
6. A. Lavie and et al, “Experiments with a Hindi-to-English transfer-based MT system under a miserly data scenario”, ACM Transactions on Computational Logic, Vol. V, 2004.
7. S. Abdelhadi, C. Violetta, J. Abderrahim, "A Prototype English-to-Arabic Interlingua-based MT system", Third International Conference on Language Resources and Evaluation Workshop Arabic Language Resources and evaluation: status and prospects, 2002
8. Mandal and et al, “Bengali and Hindi to English CLIR evaluation”, Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum 2007, pp.95-102, doi:10.1007/978-3-540-85760-0\_12.
9. Yahoo Babel Fish 2008, [Online]. Available: <https://www.babelfish.com/>.
10. A. Sjöberg, “The Use of the Copula in Non-Copula Constructions in the Languages of South Asia”, [Online]. Available: https://www.diva- portal.org/smash/get/diva2:1248101/FULLTEXT01.pdf
11. R. Weerasinghe, D. Herath, V. Welgama, Viraj, “Corpus-based Sinhala lexicon”, 2009, pp.17-23, doi:10.3115/1690299.1690302. 12 A. R. Weerasinghe, A. Wasala, K. Gamage, "A Rule Based Syllabification Algorithm for Sinhala", (2005), pp. 438-449.
12. A. R. Weerasinghe, A. Wasala, K. Gamage, "A Rule Based Syllabification Algorithm for Sinhala", 2005, pp. 438-449.
13. M.Jayasuriya, R. Weerasinghe, "Learning a stochastic part of speech tagger for Sinhala", International Conference on Advances in ICT for Emerging Regions, 2013, pp.137-143.
14. A. R. Weerasinghe, D. Herath, N. P. K. Medagoda, "A KNN based Algorithm for Printed Sinhala Character Recognition", Proceeding of 8th International Information Technology Conference -Colombo, 2006.
15. S. Sripirakas, R. Weerasinghe, D. Herath, "Statistical machine translation of systems for Sinhala - Tamil", 2010.
16. N. V. C. Vithanage, "English to Sinhala Intelligent Translator for weather forecasting domain", Colombo: Thesis, submitted BIT degree, university of Colombo, Sri Lanka, 2003.
17. B. Hettige , A. S. Karunananda, “Varanageema: A Theoretical basics for English to Sinhala Machine Translation”, Proceesing of seventh Annual Session of Sri Lanka Association for Artificial Intelligence (SLAAI), Kelaniya, 2010.
18. A. Wasala, R. Weerasinghe “EnSiTip: A Tool to Unlock the English Web”, 2008.
19. B. Hettige , A. S. Karunananda, G. Rzevski, “Selected Text Machine Translator for English to Sinhala”, 2013.
20. C. blog, “Understanding LSTM networks”, [Online], August 27, 2015, Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> .
21. R. Horev, “BERT Explained: State of the art language model for NLP”, [Online], November 10, 2018, Available: [https://towardsdatascience.com/bert-explained-state-of-the-art-language- model-for-nlp-f8b21a9b6270](https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270) .
22. C. McCormick, “BERT Word Embeddings Tutorial”, [Online], May 14, 2019, Available: <https://mccormickml.com/2019/05/14/BERT-word-embeddings-tutorial/>
23. J. Devlin, Ming-Wei, C. Kenton, L. K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, 2019.
24. G. D. Nemeth, “Identifying the right meaning of the words using BERT”, [Online], October 21, 2019, Available: [https://towardsdatascience.com/identifying-the-right-meaning-of-the-words- using-bert-817eef2ac1f0.](https://towardsdatascience.com/identifying-the-right-meaning-of-the-words-using-bert-817eef2ac1f0)