EXPLORING NEURAL MACHINE TRANSLATION FOR SINHALA-TAMIL LANGUAGE PAIR

By

L.N.A.S.H.NISSANKA

Registration No. : 2015/CS/091

This dissertation is submitted to the University of Colombo School of Computing
In partial fulfillment of the requirements for the Degree of Bachelor of Science
Honours in Computer Science

University of Colombo School of Computing
35, Reid Avenue, Colombo 07,
Sri Lanka
July, 2020

Declaration

I, L.N.A.S.H.Nissanka [2015/CS/091] hereby certify that this dissertation entitled Exploring Neural Machine Translation For Sinhala-Tamil Language Pair is entirely my own work and it has never been submitted nor is currently been submitted for any other degree.

....

Date Signature of Student

I, DR. B. H. R. Pushpananda, certify that I supervised this dissertation entitled Exploring Neural Machine Translation For Sinhala-Tamil Language Pair conducted by L.N.A.S.H.Nissanka in partial fulfillment of the requirements for the degree of Bachelor of Science Honours in Computer Science.

....

Date Signature of Supervisor

I, DR. A. R. Weerasinghe, certify that I supervised this dissertation entitled Exploring Neural Machine Translation For Sinhala-Tamil Language Pair conducted by L.N.A.S.H.Nissanka in partial fulfillment of the requirements for the degree of Bachelor of Science Honours in Computer Science.

....

Date Signature of Supervisor

Abstract

In the face of rapid globalization, the concept of translation performs the most important role in continuing the existence of native languages. Most of the research on Natural Language Processing in Neural Machine Translation has achieved an impressive result through parallel corpus dataset. Low resourced languages confront with low performance due to the lack amount of parallel corpus data. Creating parallel corpus for language pair is more expensive and needs the persons who are expert knowledge for both languages. Sinhala and Tamil consider as low resourced languages due to the lack of linguistics references. Conversely, monolingual corpora data is much easier to find than parallel corpus data and many languages with a limited amount of parallel corpus may perform prominent result with monolingual corpora.

The main aim of this research is to develop a translator for Sinhala and Tamil languages pair using monolingual corpora data. To best of our knowledge, there are no researches which used only monolingual corpora data for developing the translation between Sinhala and Tamil language pair.

In the first part of the research, we have examined the performance in Sinhala-Tamil translation using both parallel and monolingual corpora. We employed the dataset for having the sub-word unit. Sinhala and Tamil consider as morphologically rich languages. Then, we required to apply proper treatment to these languages to reduce their morphological rich nature. We conducted to address the scarcity of data we manipulated back-translation techniques and analyzed its applicability on the translation of Sinhala and Tamil languages.Through our experiment, we received 30 BLEU score for the Tamil to Sinhala translation.

As a second part of the paper, we only examined the unsupervised approach using monolingual corpora for Sinhala-Tamil translation.For this procedure, we used the word embedding technique with segmentation technique for data preparation. we conducted our experimented with a shared-encoder with two decoders architecture. Through this approach, we were able to show that there are availabilities for creating translation only using monolingual corpora data.

Preface

A novel approach for exploring neural machine translation for developing a translator in between the Sinhala and Tamil languages pair using only monolingual corpora data is introducing through this research. By using 25000 sentences of parallel corpus and 1000000 sentences of Sinhala and 400000 sentences of Tamil languages monolingual corpora collected by previous research works, we used for analysed our experiments. As the best of our knowledge, there is no research for Sinhala-Tamil NMT using only monolingual data.Through our research, first, we identify the where is the challenging matters in the Sinhala-Tamil language pair. Mainly we are addressing their nature of the having low resourced. For treating this, we selected to use the monolingual corpora data for the research.

Initially, we did a sub-word segmentation using segmentation techniques for treating the morphological richness nature in both languages. After that our research drove in two main parts. One is addressing the translation using existence parallel corpus data with monolingual corpora data. Transformer architecture was used as a DNN model and used back translation techniques for increasing the data corpus.

Addressing the second main part of the research, we are only used monolingual corpora for training the results to inhere we used normal RNN model for DNN model. For pre-processing part, we used word embedding techniques.
With constant guidance and supervision of my supervisor and co-supervisor, more conclusions were expressed about the translation of Sinhala and Tamil, which we believe, are new contributions to the body of knowledge.

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Dr. B. H. Randil Pushpananda, Senior Lecturer of the University of Colombo School of Computing for their expertise and my co-supervisor, Dr. A. Ruvan Weerasinghe, Senior Lecturer of the University of Colombo School of Computing for their expertise, enthusiasm, patience, motivation and encouragement. Their guidance helped me in every aspect of my undergraduate research study. Without their guidance and support, I would not have been able to complete my work and dissertation.

I would like to extend my sincere gratitude to Dr. D. A. S. Atukorale, Head of the Department - Department of Computation and Intelligent Systems in University of Colombo School of Computing and Mrs. M.W.A.C.R Wijesinghe, Senior Lecturer of the University of Colombo School of Computing for providing feedback on my research proposal and interim evaluation to improve my study.I would also like to thank all the staff and colleagues at the UCSC and UCSC's Language Technology Research Laboratory, for their support and input to make this research a success.

This thesis is also dedicated to my loving family who has been an immense support to me throughout this journey of life. Without them, my effort would have been worth nothing. Their love, support, patience inspires me to overcome all the obstacle in life and achieve goals with success.

Contents

ListofFigures

ListofTables

Acronyms

BLEU	Bilingual Evaluation Understudy
BPE	Byte Pair Encoding
CNN	Convolution Neural Network
DNN	Deep Neural Network
GAN	Generative Adversarial Model
HRL	High Resource Language
LRL	Low Resource Language
LSTM	Long Short Term Memory
MT	Machine Translation
MASS	Masked Sequence to Sequence Pre-training for Language Generation
NMT	Neural Machine Translation
NN	Neural Network
RNN	Recurrent Neural Network
SMT	Statistical Machine Translation
USCS	University of Colombo School of Computing

Chapter 1

Introduction

1.1 Background

Generally, human beings get used to use their native language more than a foreign language. In the face of rapid globalization, the concept of translation performs the most important role in continuing the existence of native languages. The evolution of the Machine Translation was starting from in 1930 and it was the most significant achievement of the computational linguistics. After passing several remarkable approaches to machine translation, Neural Machine Translation was able to represent the major turning point to the machine translation approach.It was almost like the paradigm shift of the machine translation. In the world, it has already shown significant results for European languages with the large content of the data. It has not been observed much on Sinhala-Tamil languages yet. Therefore, through this undergraduate research, we attempt to increase the quality of Sinhala-Tamil translation using Neural Machine Translation approach for increasing the effectiveness of communication in Sri Lanka.

Sri Lanka accords official status to Sinhala and Tamil. The languages are spoken on the island are mainly Sinhala and Tamil. Having a good translation between both languages is good for having better communication with each other in Sri Lanka. Both Sinhala and Tamil languages are morphologically rich and low resourced. The main objective of this research is to implement the translation system for morphologically rich language and a low resourced language pair, Sinhala and Tamil.

When considering the related work in Statistical Machine Translation (SMT) for

Sinhala and Tamil translation,(Weerasinghe, 2004) researched to find out the linguistic distance between Sinhala and Tamil. Researcher was able to show that linguistic distance between Sinhala and Tamil is less than between Sinhala and English. Therefore implementing a Sinhala-Tamil machine translation system is feasible than implementing a machine translation system for Sinhala-English or English-Tamil language pairs. Neural machine translation(NMT) is an approach to machine translation which showed a promising result for many European, Arabic and Chinese languages which have a large number of parallel sentences. Translations produced by NMT is considerably different from the phrase-based system. In the recent research regarding the morphological rich language with NMT (Passban, 2017), they have explained what morphology is and how it uses to construct the new words to the vocabulary. In that paper, they have addressed all the approaches that can be taken to the morphologically rich languages. Such as SMT, NN, NMT and Double-Channel NMT models. They clearly explained the model features and their effect with the morphological languages. Their dynamic programming -based segmentation model was able to perform well for NMT engines and obtained the best result. In the world, Sinhala and Tamil use a low number of population. Then Sinhala and Tamil considered as the low resourced languages. Due to the lack number of linguistic resourced, those languages have limited parallel corpora. Most of the early approach has taken parallel corpora for the research. Those languages monolingual corpora are available in publically more than parallel corpora. Through this research main purpose is to implement the Sinhala-Tamil translator using Neural Machine Translation with mostly available corpora.

1.2 Statement of the problem

Most of the research on Natural Language Processing in Neural Machine Translation has achieved an impressive result through parallel corpus dataset. Low resourced languages confront with low performance due to the lack amount of parallel corpus data. Creating parallel corpus for language pair is more expensive and needs the persons who are expert knowledge for both languages. Sinhala and Tamil consider as low resourced languages due to the lack of linguistics references. Conversely, monolingual corpora data is much easier to find than parallel corpus data and many languages with a limited amount of parallel corpus may perform prominent result with monolingual corpora. The most recent research on translating between low resourced and morphologically rich (R.Pushpananda et al., 2014) has provided a prominent foundation for Sinhala and Tamil machine translation. Most of the low resourced language researches have been taken using monolingual corpora. Few researches have taken approach using monolingual corpora and parallel corpora with the semi-supervised approach. They have used monolingual corpora to generate parallel corpus. Our attempt is to overcome the problem of limited parallel data corpus with a suitable solution and find out the efficient method for Sinhala and Tamil translation based on only monolingual corpora.

1.3 Motivation

Sinhala and Tamil are the national languages in Sri Lanka. Most of the people use these two languages for communicating in Sri Lanka. However most of them are fluent in one language. Because of that communication gap may increase in between two nationalities. One of the main motivation factors is to implement an effective machine translation system to fulfil the communication gap. In addition, Sinhala –Tamil translation is yet to be explored in Neural Machine Translation [NMT].For this study, it was of interest to investigate a solution for this community barrier by creating the translator in between two languages.

1.4 Research Gap

Sinhala and Tamil languages are considered as low resourced and morphologically rich languages. Previous attempts of creating Sinhala-Tamil translation has taken using parallel corpora. However, due to the limited number of parallel corpora and creating parallel corpora is an expensive task, we try to use monolingual corpora instead of parallel corpora for creating a Sinhala Tamil Translation. In addition to that Creating Sinhala-Tamil translation is in early-stage with monolingual corpora. In this research, it is interest to know how monolingual corpora can support to the NMT.As far as we know, no previous research has investigated to create a translator using only monolingual corpora for the Sinhala-Tamil languages pair. Therefor trying out the ability to implement translator to Sinhala Tamil languages pair using open domain monolingual corpora is the main research gap we are going to address through this research.

1.5 Significance of the project.

Sinhala and Tamil are considerably low resourced and morphologically rich languages. Using Statistical Machine Translation approach, it was able to get the best result for Sinhala-Tamil translation. But in the current society, Neural Machine Translation has surpassed accuracy with the best result in a lot of areas that were achieved using Statistical Machine Translation. But translation between Sinhala and Tamil is yet to be developed. A limited number of researches for morphological rich and low resource language have been done in Neural Machine Translation by researchers. Among those research, most of are for one morphological rich language. In Sinhala and Tamil, we need to design a proper technique to translate between two morphological languages. Most of the researchers fill the gap in-between SMT to NMT. Currently, all researchers suffer from the scarce of parallel corpus data. Through this research, we are going to find the solution from the most available monolingual corpora data instead of parallel corpus data. Therefore we can fill this research gap while doing the research. In Sri Lanka, most of people use Sinhala or Tamil as their native language. However, there is no effective translator between these two languages. When the communication is happening between these two languages, there may have misunderstood due to the lack of knowledge about languages. Having this translation will lead to reduce the misunderstanding among languages and grow better communication among others.

1.6 Research Questions

1.6.1 Question 01: What is the effect of the Monolingual Corpora on the translation accuracy?

As mentioned in the above background and statement of problem state, Sinhala and Tamil languages suffer from the issues with performance due to the low amount of parallel corpus data. Hence, Sinhala-Tamil translation fulfils the prominent result with parallel corpora with semi-supervised approach (Arukgoda and Weerasinghe, 2018). The Parallel corpus data represents the dataset with including two languages original text with its translated one. Training our model using parallel corpus data is almost like training model with a labelled dataset. It will be an aid to set-up accurate value to the hyperparameters. By addressing the research question, we are going to analyze the behavior of the accuracy when we increase the monolingual corpora size without changing the size of the original parallel corpora.

1.6.2 Question 02: What is the best approach for achieving high accuracy in Sinhala-Tamil translation using only monolingual corpora?

Earlier mentioned that collecting and creating parallel corpus is a time-consuming and very expensive task. Hence, a lot of monolingual corpora data is available publicly. Then, Collecting the monolingual corpora data is easily available. Using this concession tries to find the best approach for Sinhala-Tamil translation without using parallel corpus data is one of the answers for the research question. In addition to that, by addressing this research question, we will be able to find the quality of the translation when we are using only monolingual corpora for translation. This can apply whenever we received a new language pair for which we have no annotation.

1.7 Project Goal and Objectives

The main aim is to develop more sophisticated methods for implementing Sinhala-Tamil translator. For our first goal, we focus on one main problem Sinhala and Tamil are considered as low resourced languages and they have limited parallel corpus data. Our research aims at finding a solution to this challenging problem. We examine some previous work and propose a new method for using monolingual corpora data. Because both of these languages can collect monolingual corpora data rather than parallel corpus data. Addressing the first research question we are going to find the method for implementing a translator using both existing parallel corpus data and monolingual corpora data.

Addressing our second research question, our objective is to demonstrate the feasibility of having only monolingual corpora data for implementing the translator in between Sinhala-Tamil languages pair. Moreover, few studies have focussed on monolingual corpora data for their research (Lample, Conneau, Ranzato, Denoyer and Jégou, 2018, Lample, Denoyer and Ranzato, 2018). Researchers (Ian J. Goodfellow et al., 2014, Zhang et al., 2018) have suggested some interesting methods for using monolingual corpora. Through the second research question, we can make convince the society to this approach can be applied for any new low resourced and morphologically rich language pair which we have any annotation regarding languages. Our ultimate goal is to produce a translator using monolingual corpora for low resourced, morphologically rich languages.

1.8 Methodology

1.8.1 Details of Corpora

Due to the limited parallel corpus, Sinhala and Tamil consider as low resourced languages. For our research, we use 30000 number of parallel sentences, 10,000,000 Sinhala monolingual corpora and 4,000,000 Tamil monolingual corpora.

1.8.2 Addressing First Research Question

Several studies suggest that there are several deep learning architectures for NMT. Among them, we selected few model architectures for our experimental designs. The first question, we have to increase the size of the monolingual corpus that we have. At the same time, we are collecting more Sinhala and Tamil monolingual datasets from Sinhala and Tamil newspapers. Most of the research corporate with monolingual corpora.

Then we have to exploit this with Sinhala-Tamil parallel corpora as well as monolingual corpora. This translation most suitable for open-domain because we collect data from the newspaper, technical writing and creative writing. We apply the back translation that is the main technique for using the increase the data corpus (A. Imankulova and Komachi, 2017, Burlot and Yvon, 2019, Currey et al., 2017)

1.8.3 Addressing Second Research Question

In the second question, mainly need to identify the suitable model in NMT for Sinhala and Tamil translation (Ian J. Goodfellow et al., 2014, Passban, 2017, Zhang et al., 2018). In here we consider the size of the monolingual corpora. We need to focus on the morphological richness and rare words, grammatical structure etc. Among those things we first take morphological richness because it causes the main effect to the translation. In secondly consider the rare words and grammatical structure. Rare words can overcome with the transfer data augmentation (Fadaee et al., 2017).

1.8.4 Scope including delimitation

In scope

- To evaluate the effort of monolingual corpora in NMT using two morphologically rich and low resourced language.
- Design an Optimum technique for Sinhala -Tamil translation.

Out scope

- In this research continuation of collecting parallel corpus data will not be done. Our main purpose is to create the MT for Sinhala and Tamil using monolingual corpora.

1.9 Thesis outlines

The Thesis is organized as follows structures. Chapter 1 provides an overview of the research gap that where is the research area we need to look closely (Section 1.2) and research background in general (Section 1.3). This thesis declares a problem statement that specifies the goals associated with developing a translator(Section 1.4) and the significance of the research in section 1.5. Afterwards, the main research questions that we are going to address through the research are described under Section 1.6. Project goals and aim describe under section 1.7. The chapter closes with describing methodologies that going to follow (Section 1.8), as well as the scope of the research.

There have been numerous studies to investigate in NMT. Previous studies have shown in Chapter 2. First, this thesis declares what NMT is, which kind of paradigm shift has occurred from SMT to NMT and How NMT can explore to the Sinhala-Tamil languages pair (Section 2.1).
Chapter 03 is divided into three main sections, the first section is for describing the data pre-processing techniques and the second section for the design approach to the first research question. The third part will be described as the second research question. Chapter 4 provides the implementation details for design which describe in Chapter 3. Chapter 5 represents the analysis of result that occur from the designs.Conclusion of the overall research will describe in Chapter 6

Chapter 2

Literature Review

2.1 Literature Review

2.1.1 Neural Machine translation

Neural Machine Translation is the newly popular approach to machine translation. Due to the high performance and high quality of the translation output major people embrace the NMT for their research approach.Neural machine translation based on the encoder-decoder architecture has recently become a new paradigm of MT. A neural network based on Recurrent Neural Network performing as an encoder read and encode the input source sentence into fixed size vector called context vector. The decoder, which another RNN decodes the translation from the encoded vector.

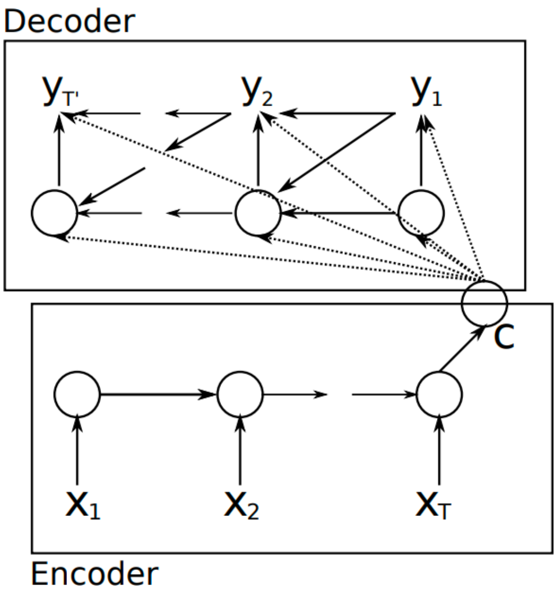


Figure 2.1: Encoder Decoder Architecture

Figure 2.1 shows the architecture of the encoder decoder. A sequence to sequence model aims to map a fixed-length input with fixed-length output where the length of the input and output may differ. There is a more challenging type of sequence prediction problem that takes a sequence as input and requires a sequence prediction as output. Models concern that makes these problem challenging is that the length of the input and output sequences may vary. There fore this architecture more suitable for short sentences. To address these issues, (D. Bahdanau and Bengio, 2014) have introduced novel approach for machine translation while introducing attention model.Figure 2.2 shows the attention model. This method caused to fix the problem with a fixed vector size during the translation. In this research, they have encoded the sentences from left to right as well as from right to left. Then the each time a word is translated. Then the model predicts the target word, based on the context vectors associated with these source positions and all the previous generated target words.

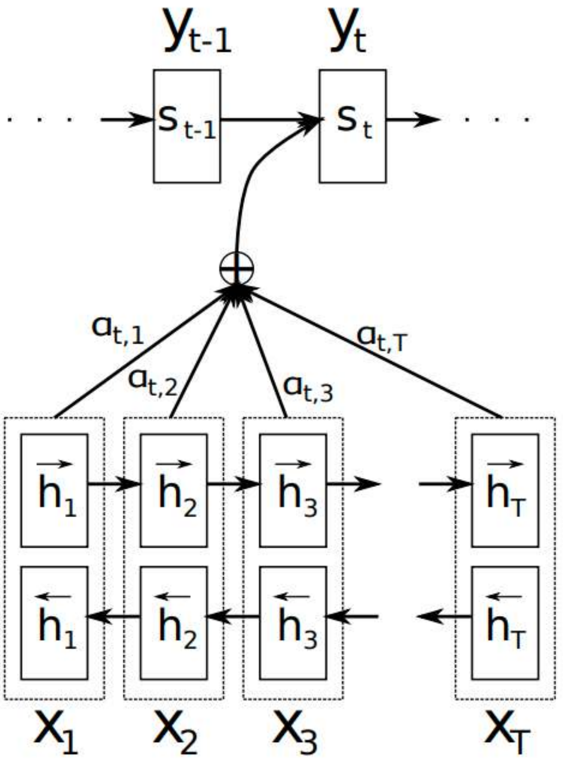


Figure 2.2: Attention Architecture

With the help of the previous research, (Vaswani et al., 2017) has introduced the high performance transform attention model to overcome the problem is time-consuming in the based line attention model. In addition to that,(Ian J. Goodfellow et al., 2014, Zhang et al., 2018) researchers have introduced Generative Adversarial Network model for showing the promising result for NMT.

2.1.2 Low resourced Language

In fact, Neural Machine Translation needs more data for getting a notable result. This was the main drawback for the low resourced languages for their research in NMT. Hence (Currey et al., 2017) researchers keep their direction to the increasing the data set applying some methods. In here, they have used a method to convert monolingual corpora in the target language into a parallel corpus by copying it, so that each source sentences is identical to the target language. NMT system is trained with the parallel data which is mixed with copied data. However, they have used both parallel and monolingual corpora for improving the NMT for low resourced language. (A. Karakanta and Genabith, 2017) research has taken a step for overcoming the challenge of scary of parallel data. They have used closely-related (enrich with morphologically rich nature of language) High Resourced Language and Low Resourced Language pair for transform HRL data into LRL using transliteration model. After that back-translated monolingual LRL data with the models trained on transliterated HRL data. Finally, they have used the resulting parallel corpus to train their final model.

Another notable method for low resourced language is transfer learning which is introduced by (Nguyen and Chiang, 2017, Zoph et al., 2016). Transfer learning is the method which used the first result of one task repurposed on the second task to generate the result. Data augmentation for low resource language in neural machine translation (Fadaee et al., 2017) described the data augmentation technique for gaining the effective result from low resource pairs. Using the data augmentation leads to generate more rare words during translation and consequently to higher translation quality. While using transfer learning with combining byte-pair encoding (Nguyen and Chiang, 2017) cause to increase the performance of translation on a low resource language pair by exploiting its lexical similarity with another related, low resource language.

2.1.3 Monolingual corpora and parallel corpora for Low resourced

Neural Machine translation is a novel approach to machine translation showing the promising result for translation. Arguably, it performs excellent performance in machine translation due to the availability of high-quality parallel corpora. Unfortunately, collecting parallel corpora is an expensive task as they require specialized expertise. In addition to that there is a limited number of parallel corpora for low resourced language and with heavy domain restriction for being limited parallel corpora. Considering all the fact researchers advert their direction to the monolingual corpora. In contrast, monolingual corpora are publically available for all the languages.

This presents of "unlabeled" monolingual corpora give prominent opportunity to leverage result in NMT. (Gulcehre et al., 2019) is one of the research which used monolingual corpora. Through this research paper, they did several experiments using monolingual corpora.

In this work, they present a way to effectively integrate a language model (LM) trained only on monolingual data in the target side into an NMT system. They used encoder as bidirectional RNN – Integrating Language Model into decoder as training architecture. This introduced Integrating Language Model decoder shows the promising result for low resourced language.

Due to the lack amount of parallel corpora, the researcher introduced the back translation technique (Sennrich et al., n.d.) to increase monolingual corpora in a productive way by leveraging the target language data.

In here, researchers propose a method for generating synthetic sentences using back-translating sentences in the monolingual corpora. Because of that, they were able to increase the size of existing parallel corpora. Unfortunately, this proposed method introduces noise and seems really effective only when the synthetic parallel sentences are only a fraction of the true ones. Hence, this approach does not allow to leverage huge quantities of monolingual data.

Moreover, in (Gangi and Federico, 2019) introduced the method for increasing the dataset using the word embedding approach. In this paper, they directly feed the NMT system with external word embedding trained in monolingual source data. Due to the lack of parallel corpus more research focusing on the use of monolingual corpora for NMT. (A. Imankulova and Komachi, 2017, Sennrich et al., n.d.) showed that target side-monolingual corpora data can upgrade the quality of the decoder model without any changes to the network architecture. Here used monolingual data with automatic back-translations for treating as the additional training data. Providing the opposite approach to the above method, (Zhang and Zong, 2016) has introduced a method to exploit source-side monolingual data by feeding to the NMT while generating the synthetic large scale of parallel corpus data and multi-task learning to predict the translation and the reordered source-side monolingual sentences simultaneously.

2.1.4 Monolingual corpora

(Lample, Denoyer and Ranzato, 2018) is the paper which was used only monolingual corpora for both source and target language to neural machine translation. Implementing the model design for two monolingual corpora languages will lead to overcoming the problems when creating the translation when encountering a new languages pair for which have no annotation. Through this research, researchers have introduced a model to improve NMT in using monolingual corpora. They have taken auto-encoder for training the reconstruct a sentence from a noisy version of it and used a translation model as encoder-decoder for training to translate a sentence in the other target language. They have taken a simple unsupervised word-by-word translation model and iteratively improved based on the reconstruction loss, and use discriminator to align latent distributions of both the source and the target language.

(Zhang and Zong, 2016) have used align word embedding space without any cross-lingual supervision on unaligned datasets of two different languages for making the linear mapping in between a source and target language. This research shows that it works for low resource language pairs, and can be used as a first prominent step for unsupervised machine translation.

2.1.5 Segmentation methods for the Morphological Richness

Morphological richness is the most attractive and unique feature of the language and it is very difficult to manage in the translation. Through (Passban, 2017), they have introduced morpheme segmentation models to segment word into the machine-understandable unit.For example, (Renduchintala et al., 2019) research has provided evidence for a character aware decoder to capture the pattern in the target language.

(Hans and Milton, 2016) research, they addressed the morphological rich language like Tamil and English. They have introduced the morphological segmentation to the morphologically rich language before translation process.

2.1.6 Techniques to increase the corpora size for the dataset.

Transliteration, Transfer learning, Data-augmentation, Monolingual word embedding and Back translation have used for low resourced languages.

- **Transliteration:** Transliteration model can apply to the languages which have approximately similar pronunciation with words in both languages, however Sinhala and Tamil languages have no approximate pronunciation. Then the transliteration model can not apply for Sinhala and Tamil.

- **Transfer learning:** Encoder-Decoder based architecture shows a prominent result in large datasets scenarios, but it is less effective for low resource languages. Therefore transfer learning has been introduced by researchers (B. Zoph and Knight, n.d.) for enhancing the performance of machine translation with low resourced language.This has been widely adopted in the field of natural language procesing such as speech recognition, document classification and sentiment analysis. In transfer learning, they have used two models called the parent model and chid model. They have used high resourced language with the common target language and used low resourced language for the child model. If we mapped it to Sinhala-Tamil language pair, then we have to take English as an intermediate target common language. Our main purpose is going to translate Sinhala -Tamil indirectly without using intermediate language. Therefore we didn't use transfer learning to our research.

- **Data augmentation:** Data Augmentation has mainly used in the image processing domain. It is very new to natural langue processing. Their approach to alters existing parallel sentences targeting low-frequency words and augments the data by generating new diverse context for low-frequency words and the corresponding translations. It is almost like paraphrasing which is meaning- preserving. Through this technique, they have increased the size of the training data by diversifying the context of rare words yields better translations.

- **Monolingual word embedding**: Word embedding was introduced as a result of limiting to identify the rare words in NMT. A word embedding is a learned representation for text where words that have the same meaning have a similar representation.

- **Back translation**: We chose the **back translation** approach for addressing the first research question. It has shown the promising result for a low resourced language than other methods.In the back translation techniques, we used source side back translation for our research. To translate from Tamil to Sinhala, we created source side synthetic parallel corpus using target-side monolingual sentences.

  Figure 2.3 shows the process of the back translation. In here there are 4 main steps in the process.

  1. In the first step, we trained the model using our authentic source language as the target language and authentic target language as the source language.
  2. After that, we translate some authentic monolingual target language sentences (it is our source language in back-translation process) into synthetic parallel corpus in the authentic source language.
  3. In the third step, we merge the all authentic and synthetic parallel corpora for creating the new dataset.
  4. In the final stage, we train our model using newly created dataset.

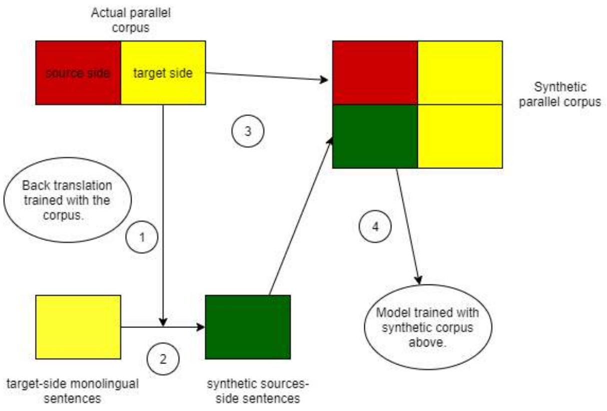

Actual parallel corpus
source side
target side
Synthetic parallel corpus
3
Back translation trained with the corpus.
1
4
Model trained with synthetic corpus above.
2
target-side monolingual sentences
synthetic sources-side sentences

Figure 2.3: Back Translation Technique

2.1.7 Deep Neural Network model.

When going to apply the deep neural model that we identified five neural models. Among them chose Transformer architecture and Generative adversarial network model considering their pros and cons.

Recurrent neural network model

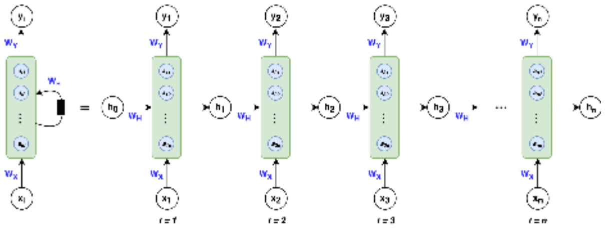


Figure 2.4: Recurrent neural network model

Figure 2.4 shows the architecture of the RNN.

- Clearly, this mechanism is very useful for NLP tasks.
- Sequence input and sequence output (e.g. Machine Translation: an RNN reads a sentence in English and then outputs a sentence in French).
- LHS shows you the RNN and the right-hand side shows you the unfolding view of RNN.

Problem: Simple RNNs are not powerful enough to summarize complex structures and capture their properties. They also have problems in remembering long-distance dependencies.

Long Short Term Memory model

Remembering information for long periods practically their default behavior, not something they struggle to learn. This model introduced for NLP as a solution to the RNN long-distance problem. But when the long sequence sentence come to the translation it gets more time. Because it translates the sentence word by word in a sequence way. It is getting more time when the sentence has a long length. In addition to that, the Decoder needs to wait until all the encoders finish their task. After that decoder can decode the encoded sentence to the target language. It also causes to increase the computation time. It is the major problem encounter in RNN model. To overcome the problem Attention mechanism introduced by the (M. Luong and Manning, 2015)researchers. But the problem still cannot resolve using default attention model. Due to the high computation time,transformer architecture (Vaswani et al., 2017) introduced by researchers. It can reduce the computation time while parallelization the sequence to sequence approach. Considering these factors we chose transformation architecture for DNN model in the first research question design.

Transformer Architecture

In the Section 2.1.1, we looked at the attention method in modern deep learning models. Attention is a concept that supports to improve the performance of neural machine translation applications. As a enhance version of attention model, the transformer was proposed in the paper 'Attention is All your Need' by (Vaswani et al., 2017). The biggest benefit of the transformer model is, it lends itself **parallelization.**

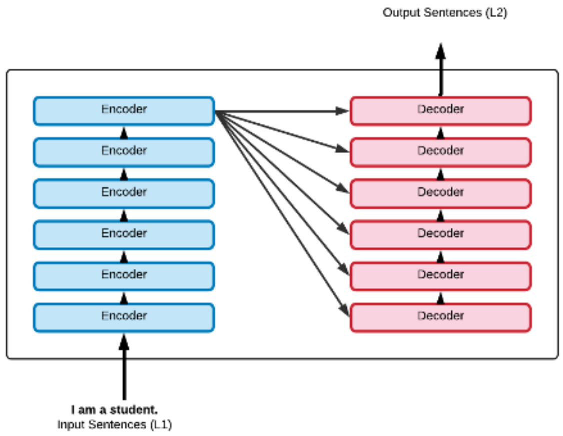


Figure 2.5: High level structure of transformer Architecture

The transformer in NLP is novel architecture that aims to solve sequence to sequence task while handling long-range dependencies with ease.
"The Transformer is the first transduction model relying entirely on self -attention to computer representations of its input and output without using sequence-aligned RNNs or convoluation."

Figure 2.5 displays the high level structure of the transformer architecture. The encoding component is a stack of encoders and decoding component is a stack of decoders of the same number of encoders. The encoder's inputs first flow through a self-attention layer. It helps the encoder look at other words in the input sentence as it encodes a specific word. The decoder has both those layers, but between them is an attention layer that helps the decoder focus on relevant parts of the input sentence. As the model processes each word (each position in the input sequence), self attention allows it to look at other positions in the input sequence for clues that can help lead to a better encoding for this word. We described the attention model in section 2.1.1. We selected the transformer architecture for addressing our first research question.Appendix A refer for transformer layer architecture.

Generative Adversarial Model

GAN the stands represent the Generative Adversarial Networks. Someone can be questioning why we need to study the GAN framework. Generative models can be trained with lost data and can provide prediction inputs that are missing data. The interesting approach is semi-supervised learning. In this approach, GAN used both labelled and unlabeled data. Then it will easy to identify some rare words in the translation.

GAN (Zhang et al., 2018), framework is with two different models called generator and discriminator. The generator creates samples that are intended to come from the same distribution as the training data. Then discriminator analyses the samples to determine whether they are real or fake.supervised learning techniques are used by discriminator to divide the input into two classes real or fake. This model also uses both parallel and monolingual corpora data. Appendix A.3 for the architecture of the GAN.

We used this adversarial network concept to address our research question 02 by encouraging from GAN framework.

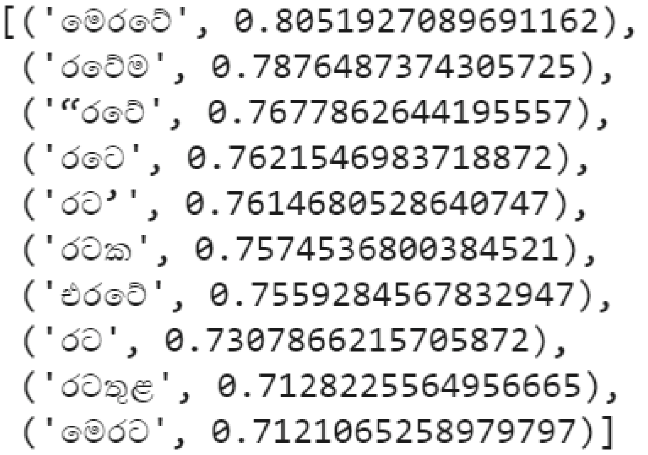
2.1.8 Data pre-processing techniquesues

In research question one we investigated the one pre-processing technique BPE. In addition to that word embedding data representation we used in here.

Word Embedding

Word embedding is a set of language modelling and feature learning techniques in NLP. The surface of word embedding is mapping similar meaning of words to have similar representations. In our research, the main aim is to build a translator model for two languages corpora which are not labelled. Using word embedding, we can identify approximately similar words in corpora. We require to label the word with numerical representation. In word embedding process, each word represents with the n-dimensional vector with 0 and 1. Here n is the number of words in vocabulary. This representation described as one-hot vector encoding. One-hot vector is the traditional approach for word embedding.
One-hot-vector is simple and easy to implement. However, there are some drawbacks of one-hot-encoder. One is when representing the word in word vector matrix it takes alphabetically order. For instances, the word 'endure' and 'tolerate', although they have a similar meaning , their target '1's are far from each other due to the order of the matrix. Therefore, it may reason to not gather these two words together. Sparsity which means wasting a lot of space is another issue that occurs in one-hot-encoding.

There are two state-of-the-art methods for word embedding, Word2Vec and FastText. Both Word2Vec and FastText can address the above-mentiond drawback in one-hot-encoding. We employ word embedding with FasteText embedding method which introduced by Facebook researchers in 2016. Through, Word2Vec has the biggest challenge to identify the rare words, that not in the training dataset.FastText breaks words into several n-grams (sub-words) instead of feeding individual words into Neural Network. It might be a useful approach for segmenting the morphologically rich language Sinhala and Tamil.



```
[('මෙරටේ', 0.8051927089691162),
 ('රටේම', 0.7876487374305725),
 ('“රටේ', 0.7677862644195557),
 ('රටෙ', 0.7621546983718872),
 ('රට’', 0.7614680528640747),
 ('රටක', 0.7574536800384521),
 ('එරටේ', 0.7559284567832947),
 ('රට', 0.7307866215705872),
 ('රටතුළ', 0.7128225564956665),
 ('මෙරට', 0.7121065258979797)]
```

Figure 2.6: Similar words represents for word රටේ Word embedding for the data corpora.

Word embedding enables to identify similar words and group it. FastText helps to build scalable solutions for text representation and classification. As a example, Figure 2.6 presents how to gather the approximately similar word for රටේ in monolingual corpora using word embedding.

Word Vectors Mapping

The main purpose in word vector mapping to built up the relationship with each word in two languages. We use Vecmap tool (M. Artetxe and Agirre., 2018) for building a mapper between Sinhala and Tamil languages pair.Word embedded data use for these mapping.

2.1.9 Attempts to improve translation employing machine translation techniques.

Using shared encoder and decoders

For the experiment so far, we adopted shared encoder which implemented with BiRNN. In the research work presented by (Artetxe and Labaka, 2018). We take publicly obtainable implemented framework UndreaMT (Artetxe and Labaka, 2018) for approaching the second research questions.

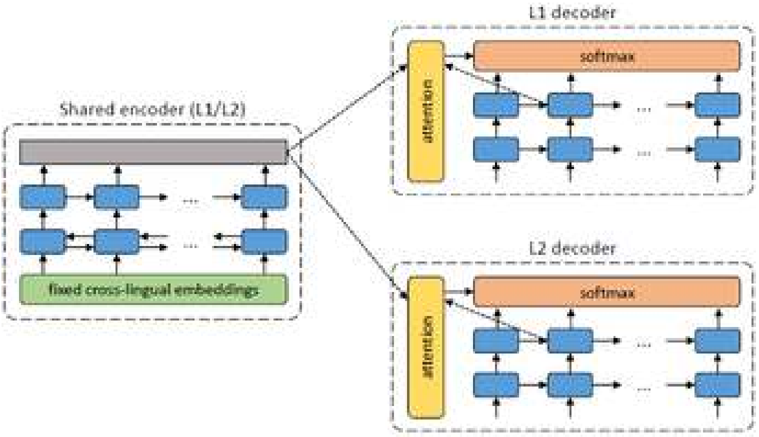


Figure 2.7: High Level Architecture for UNdreaMT tool : from(Artetxe and Labaka, 2018)

Figure 2.7 manifests the architecture of UNdreaMT tool. It consists of three main layers as two decoders for both languages and one shared encoder. This is similar to the standard encoder-decoder architecture with an attention mechanism. Encoder has two layer bidirectional RNN 2.8 and decoder has two-layer RNN. All RNN use GRU cells with 600 hidden units and dimensionality of the embeddings is set to 300.

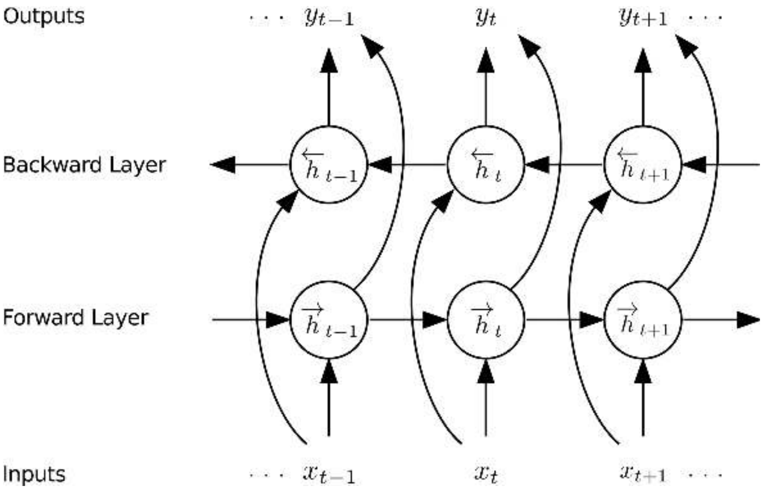


Figure 2.8: Bidirectional Recurrent Neural Network

The system is trained to each sentence with two steps. One is denoising and other step is on-the-fly back translation.

Denoising optimizes the probability of encoding a noised version which means using authentic sentences create it synthetic version of sentences, of the sentence

with the shared encoder and reconstructing it with the L1 decoder
On-the-fly back-translation, which translates the sentence in inference mode which mean encoding it with the shared encoder and decoding it with the L2 decoder. Hereafter, it optimizes the probability of encoding this translated sentence with the shared encoder and recovering the original sentence with the L1 decoder.

Using sequence to sequence encoder-decoder introduced by MASS

MASS[Masked Sequence to Sequence Pre-training for Language Generation] is the one of Open Source tool which creates inspired by BERT. It is a sequence to sequence encoder-decoder based language generation tool which introduced by (Song et al., 2019).

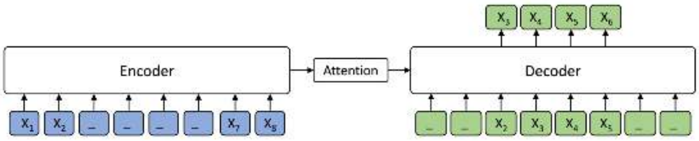


Figure 2.9: MASS high level architecture: from(Song et al., 2019)

The unsupervised approach is taken for addressing the second research question. Availability of monolingual corpora data drives to this approach. MASS is another approach which is different than the UndreaMT framework.
The transformer is chosen (Vaswani et al., 2017) as the basic model structure, which consists of 6-layer encoder and 6-layer decoder with 1024 embedding/hidden size and 4096 feed-forward filter size. For neural machine translation task, the model is required to pre-train on the monolingual data of the source and target languages.

2.1.10 Research for Sinhala and Tamil languages

Studies of (Arukgoda and Weerasinghe, 2018) are well documented, it is also well acknowledged that, the first research for Sinhala-Tamil translation using Neural Machine Translation. In here they mainly used the parallel corpus for their research. In here they have used back translation technique for increasing the parallel corpus.

The main aim of this (Guzman et al., 2019) research paper is to release quality benchmark datasets for low resourced languages pairs English-Sinhala and English-Nepali. They have used Wikipedia pages for collecting data to the evaluation datasets. In model and architectures section, the researches have taken four approaches for their training setting. Although they have mentioned the models regarding the semisupervised and supervised approaches, there is no mentioned model for unsupervised approach. They have mentioned that they used the unsupervised approach with monolingual corpora. In this research, they have used transformer architecture for NMT model in supervised approach. They have preprocessed Sinhala language sentences using NLP library and used Byte Pair Encoding too.

2.1.11 Bilingual Evaluation Understudy (BLEU)

Human evaluation of machine translation weighs in the machine translation. Because it gives an accurate result to the output. But when come to the research, getting human evaluation is not practically applicable because it gets a long time to check the process and we are going to deal with the huge number of sentences then it is very difficult to go through every sentence in the given time. Due to the problem, BLEU (Papineni et al., 2002) score is introduced to getting evaluation Natural Language Processing task. The BLEU score works by counting matching n-grams in the translated sentences to n-grams in the reference text.

BLUE score normally getting their score considering under formula

- Accuracy = (No. of correct words /No. of words)

We assume that BLEU will accelerate the Machine translation in research and development.

2.1.12 Summary

Considering the above literature review, we can see most of the research in morphological rich and low resourced language have taken in both parallel and the monolingual corpora. Then when we are going to focus our research in only monolingual corpora, we need to look at how to analyze morphological richness through preprocessing and post-processing in the research very well. In our research, we are going to find a solution using monolingual data instead of parallel data.

Chapter 3

Research Design

In this chapter, we described the overall design to address each research questions presented in chapter 01. In the first part, we addressed the first research question that ’What is the effect of the Monolingual Corpora on the translation accuracy?’ and the second part we addressed the second research question that ’What is the best approach for achieving high accuracy in Sinhala-Tamil translation using only monolingual corpora?’ using different experiment using monolingual corpora.

3.1 Research Design for First Research Question

The main focus of the experiments was to investigate the performance of translation using both monolingual corpora and parallel corpora data in our first research question. In our preliminary experiments, we estimated some of the models and preprocessing techniques that used to use through the first research question.In this experiment has five main stages: data preprocessing, training, translating, post-processing and evaluating.

3.1.1 Data corpus

Most of the researches for low resourced languages have done using both parallel and monolingual corpora. When coming to the Sinhala and Tamil languages they are morphologically rich languages and low resourced language.

For our experiments, we use a parallel corpus of 25000 sentences which has the length of a sentence between 8 and 12. Table 3.1 shows the details of parallel corpus data. We use 10 M words Sinhala monolingual corpus and 400,000 word Tamil monolingual corpus. Both these corpora are suitable for an open-domain translation as they have been collected from sentences from different domains such as newspaper articles, technical writing and creative writing.

Table 3.1: Corpus of the Parallel Dataset

Corpus Statistics	Sinhala	Tamil
Number of sentence pairs	26,187	26,187
Total Number of Words (T)	262,082	227,486
Vocabulary Size	38,203	54,543
V/T %	14.58	23.98

Table 3.2: Corpus of the Monolingual Dataset

Corpus Statistics	Sinhala	Tamil
Number of sentence pairs	1,067,173	407,578
Total Number of Words (T)	13,158,152	4,178,440
Vocabulary Size	933,153	301,251

Feasibility of collecting monolingual corpora

If we are going to create the parallel corpus, we need good expertise knowledge in both languages and it is getting more time and expensive task. However, there are lot of resources available in publically for both languages. Hence, we focused our direction for using monolingual corpora. After that, we tried to find the feasibility for collecting monolingual corpora if we will need more data during the research process. We wrote the python script for collecting monolingual corpora. Through this script, we were able to collect near to 1000 sentences per day for monolingual corpora. Due to this result our feasibility study got successful conclusion. When the necessity of the monolingual corpora arise then we can use this script for collecting monolingual corpora data. This is based on quantitative research. To examine the effect of the monolingual corpora, we created synthetic parallel corpus using monolingual corpora data.

3.1.2 Data Pre-processing

As we identified earlier, Sinhala and Tamil language have main inheritance properties. Those are Sinhala and Tamil languages consider as low resourced languages due to the lack of resources of linguistics resource and Sinhala and Tamil consider as morphologically rich languages.

The collected data must be refined. The refinement process includes sorting sentences by corpus in both languages and eliminating noise such as special characters. Mainly, we need to apply some treatment to reduce the morphological richness in the language pairs. As a treatment to reduce morphologically richness of languages, researchers have introduced subword segmentation.

Normal word representation can not handle the unseen and rare word well. for that issue, character embedding is one of the solutions to overcome the out-of-vocabulary[OOV]. However, it may too fine-grained any missing some important information. Subword is in between word and character. It is not too fine-grained while able to handle unseen word and rare word. In subword segmentation, the word can be divied into small pieces, until it has a meaning and refines spacing using the POS tagger or segmenter for each language. This allows you to perform additional segments and construct a vocabulary list. As a subword segmentation techniques, we can take two main techniques. One is Byte Pair Encoding(BPE)and the Other one is the workpiece.

(Sennrich et al., 2016) proposed to use Byte Pair Encoding (BPE) to build a subword dictionary. WordPiece is another word segmentation algorithm and it is similar to BPE. At this time, the segmented models learned for the BPE segment should be kept for future use. In our research, we mainly obtained by the two approaches for preprocessing. The first approach for full word form segmentation and a second approach for using BPE.

Full word form segmentation

- **Sinhala** : බුද්ධාගම |[1] නොවන්නට | ඉන්දියාවට | පෙන්වීමට | ඉතිහාසයක් | නොවන්නට | තිබිණැයි | මතයක් | ද | ඇත | .

- **Tamil**: ெ◌ளத்த | மதம் | இல்லாது | இருந்தால் | இந்தியாவில் | சரித்திரம் | என | வழியுறுத்துவதற்கு | சரித்திரம் | ஒன்றும் | இல்லாமல் | போயிருக்கும் | என்ற | கருத்தும் | இருக்கிறது | .

Byte pair encoding

- Sinhala : බු@@ | ද්@@ | ධා@@ | ගම නො වන්නට ඉන්දිය@@ | ◌ාවට පෙ@@ | න්@@ | වීමට ඉතිහාස@@ | යක් නො වන්නට තිබි@@ | ණ@@ | ◌ැ@@ | යි මත@@ | යක් ද ඇත .

- Tamil : ெ◌ள@@ |த்த ம@@ |தம் இல்லா@@ | து இருந்த@@ | ◌ால் இந்தியா@@ | வில் ச@@ ரி@@ | த்திர@@ | ம் என வழ@@ | ◌ிய@@ | ◌ு@@ |று@@ | த்துவ@@ | தற்கு ச@@ | ரி@@ | த்திர@@ | ம் ஒன்ற@@ | ◌ும் இல்லா@@ | மல் போ@@ | ய@@ | ◌ிரு@@ | க்கும் என்ற கருத்த@@ | ◌ும் இருக்கிறது .

3.1.3 High Level Architecture for Research Question 01

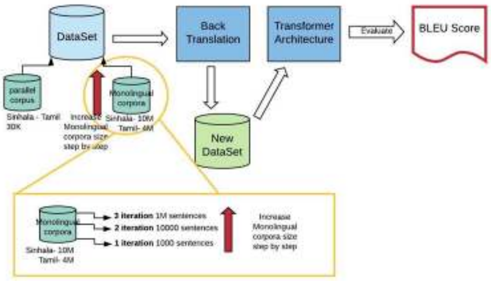


Figure 3.1: High Level Architecture for Research Question 01

We create a dataset including two main different datasets. One is the parallel corpus of Sinhala Tamil and another one is monolingual corpora for each language. After that, we apply the dataset for back-translation approach (Burlot and Yvon, 2019, Graça et al., n.d.) for extending the dataset. At the same time, we increase the size of monolingual corpora without changing the size of the original parallel corpus. After creating the new data set apply the transformer architecture (Guzman et al.,

[1] Subwords boundaries are marked with '|'.

2019, Vaswani et al., 2017) which is the main DNN model in attention-based. Finally, we evaluate the process based on the BLEU score.

3.2 Research Design for Second Research Question 02

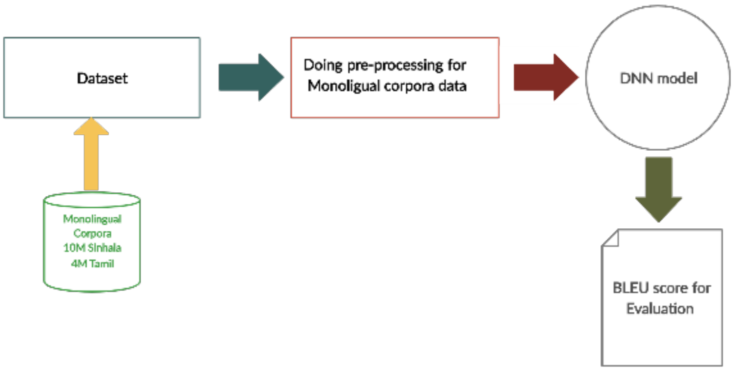

Dataset
Doing pre-processing for Monoligual corpora data
DNN model
Monolingual Corpora 10M Sinhala 4M Tamil
BLEU score for Evaluation

Figure 3.2: High Level Architecture for Research question 02

Our main approach is to find out the capability of the developed translation between Sinhala and Tamil languages pair using only monolingual corpora data. Here Figure 3.2, represents the flow to the high-level diagram of this process. In the first step, we are applying the preprocessing techniques that we used in question one. After doing the pre-processing, we pick the DNN models which are defeated our requirements, to train the dataset.

3.2.1 Research Data-corpora

The Sinhala and Tamil consider as morphologically rich and low resourced language. Through our second research question, we mainly address for acknowledging the problem of owning less number of linguistic resources of these two languages. Therefore our exploration directed for monolingual corpora.

Initial we have described the details of monolingual corpora in table 3.2. Here we supposed to use monolingual corpora data only. We prepare the dataset by removing duplicate sentences and the null strings. After that, we suppose to apply the BPE segmentation to reduce the morphological richness of both languages.

BPE enables the ability to identify rare words. Especially when coming to the monolingual corpora, those two datasets are unlabeled data. Therefore, it is very challengable to recognise the words with the correct translated word. The morphological nature of languages boosts it that difficult to identify the words. Henceforward, we decided to apply BPE segmentation to the words. we use (Sennrich et al., 2016) subword unit implementation for applying BPE to the corpus.

3.2.2 Evaluation

As mentioned is Chapter 1, the quality of the different translation approaches we explore will be measured using the automatic evaluation technique BLEU (Bilingual Evaluation Understudy). We evaluated the results in First research question obtained for the following models

- BLEU score for full-word form (based approach)
- BLEU score for BPE segmentation approach
- BLEU score for different DNN architecture
- BLEU score for back-translation.

In addition to that we do manual semantically analysis for finding the sentences which have same meaning. An error analysis was also done to compare the effect of our treatments at different stages, against the full word-form baseline model as follows.

- 01. Count the number of total words (TotW) and unique words (UniW) in each training (Tr) and testing (Te) datasets.
- 02. Count the number of out-of-vocabulary (OOV) words in the test dataset (as a percentage of test dataset).

3.2.3 Summary

In this chapter, we have discussed the high-level architecture and the overall design to address the two research questions we are focusing on. As the initial step, the methodology to find suitable segmentation for pre-processing in both language. Hereafter, find the performance using both parallel and monolingual corpora for Sinhala- Tamil. This discussion also contains how we designed our research process to treating the challenges of Sinhala–Tamil translation step by step. Chapter discuss the technique of DNN model and dataset increasing methodologies. When coming to the second research question, we address for to monolingual corpora dataset. All the observation and result analysed in the Chapter 5 and 6.

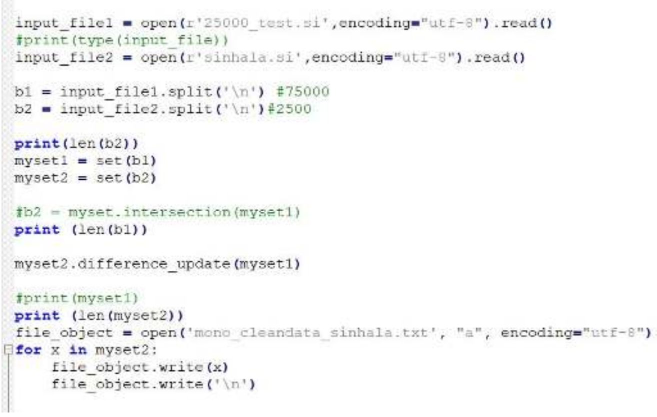
Chapter 4

Implementation

In this chapter, we describe the implementation of the methods that we have discussed in chapter 3.We are going to explain the pre-processing techniques and details of the research tool that we used to our research question.

4.1 Pre-processing Techniques

4.1.1 Removing the duplicated sentences from corpora.



```
input_file1 = open(r'25000_test.si',encoding="utf-8").read()
#print(type(input_file))
input_file2 = open(r'sinhala.si',encoding="utf-8").read()

b1 = input_file1.split('\n') #75000
b2 = input_file2.split('\n')#2500

print(len(b2))
myset1 = set(b1)
myset2 = set(b2)

#b2 = myset.intersection(myset1)
print (len(b1))

myset2.difference_update(myset1)

#print(myset1)
print (len(myset2))
file_object = open('mono_cleandata_sinhala.txt', "a", encoding="utf-8")
for x in myset2:
    file_object.write(x)
    file_object.write('\n')
```

Figure 4.1: Removing duplicate sentences in data set.

4.1.2 Treatment to reduce morphological richness in the Sinhala and Tamil Languages pairs.

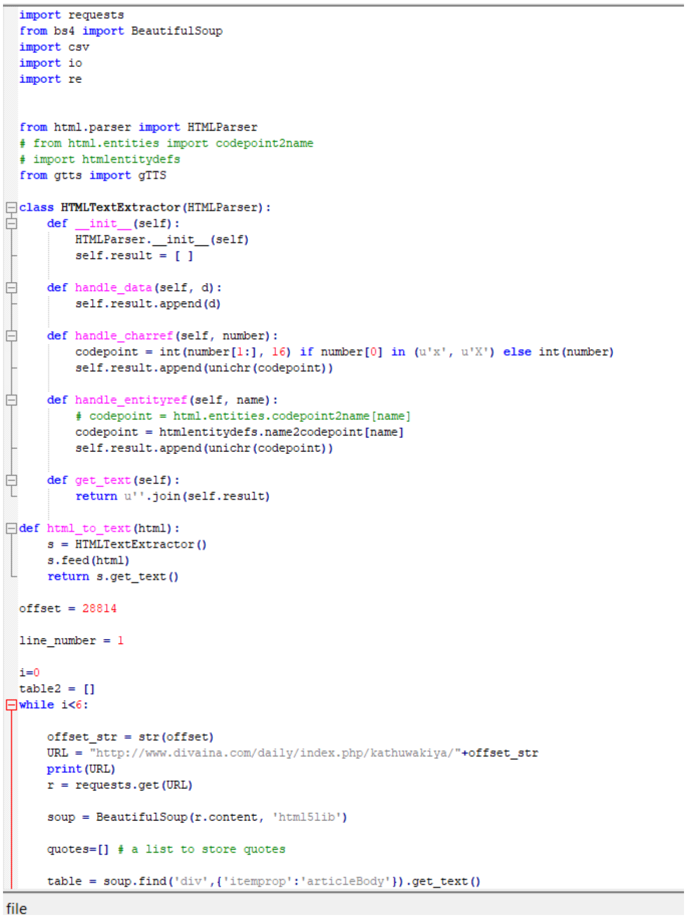
Byte Pair Encoding Technique

We used byte pair encoding for balancing the character - and word-level hybrid representations that make it capable of managing large corpus.Character-level and word-level representations are based on the hyper-parameter (number of merge operation) in BPE. Low value of hyper-parameter would lead to character-level segmentation and very high-level value would lead to word-level segmentation.
The main purpose of using BPE is identifying unknown/rare words in the vocabulary with appropriate sub-word tokens without introducing <unknow> token.

Then We used BPE approach with using different hyper-parameter values for our pre-processing technique.

4.1.3 Data collecting- Web Scraping Code

If we need to collect more monolingual corpora, we did the feasibility study for collecting monolingual corpora.



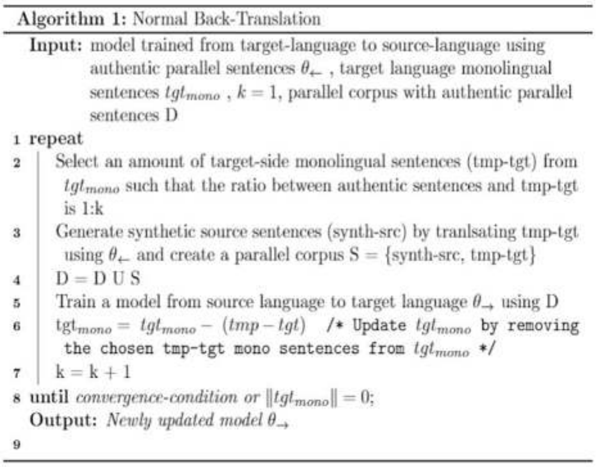
```
import requests
from bs4 import BeautifulSoup
import csv
import io
import re


from html.parser import HTMLParser
# from html.entities import codepoint2name
# import htmlentitydefs
from gtts import gTTS

class HTMLTextExtractor(HTMLParser):
    def __init__(self):
        HTMLParser.__init__(self)
        self.result = [ ]

    def handle_data(self, d):
        self.result.append(d)

    def handle_charref(self, number):
        codepoint = int(number[1:], 16) if number[0] in (u'x', u'X') else int(number)
        self.result.append(unichr(codepoint))

    def handle_entityref(self, name):
        # codepoint = html.entities.codepoint2name[name]
        codepoint = htmlentitydefs.name2codepoint[name]
        self.result.append(unichr(codepoint))

    def get_text(self):
        return u''.join(self.result)

def html_to_text(html):
    s = HTMLTextExtractor()
    s.feed(html)
    return s.get_text()

offset = 28814

line_number = 1

i=0
table2 = []
while i<6:

    offset_str = str(offset)
    URL = "http://www.divaina.com/daily/index.php/kathuwakiya/"+offset_str
    print(URL)
    r = requests.get(URL)

    soup = BeautifulSoup(r.content, 'html5lib')

    quotes=[] # a list to store quotes

    table = soup.find('div',{'itemprop':'articleBody'}).get_text()
```

Figure 4.2: Code for web-scraping

We used figure 4.2 code for collecting monolingual corpora data. Appendix B.1 shows the sample of collected data.

4.2 Back Translation Algorithms implemented

4.2.1 Normal Back-Translation



```
Algorithm 1: Normal Back-Translation
Input: model trained from target-language to source-language using
       authentic parallel sentences $\theta_{\leftarrow}$ , target language monolingual
       sentences $tgt_{mono}$ , $k = 1$, parallel corpus with authentic parallel
       sentences D
repeat
    Select an amount of target-side monolingual sentences (tmp-tgt) from
     $tgt_{mono}$ such that the ratio between authentic sentences and tmp-tgt
     is 1:k
    Generate synthetic source sentences (synth-src) by tranlsating tmp-tgt
     using $\theta_{\leftarrow}$ and create a parallel corpus S = {synth-src, tmp-tgt}
    D = D U S
    Train a model from source language to target language $\theta_{\rightarrow}$ using D
    $\mathrm{tgt}_{mono} = tgt_{mono} - (tmp - tgt)$   /* Update $tgt_{mono}$ by removing
     the chosen tmp-tgt mono sentences from $tgt_{mono}$ */
    k = k + 1
until convergence-condition or $\|tgt_{mono}\| = 0$;
Output: Newly updated model $\theta_{\rightarrow}$
```

Figure 4.3: Code for web-scraping

4.3 BLUE score algorithm

4.3.1 Validation and testing the models

Throughout the experiments conducted using only the 25000 parallel corpora, and 10,000,000 monolingual corpora, the models were validated using a small extracted parallel dataset of 1250 sentences. The perplexity that is given by the OpenNMT framework will help to evaluate the training and validation set accuracy. To test the models, we selected a test-set of 1250 sentences. The test-set consisted of sentences which were mutually exclusive from the training-data. Using the preplexity value we can find out the best model in the training process.

4.4 Research Tool

- We chose OpenNMT (G. Klein et al., 2017) which is Open source frame work for sequence to sequence translation. It is implemented with in Torch with many model configuration.
- Used Sub-wordNMT (Sennrich et al., 2016) for doing preprocessing BPE. This framework contains preprocessing scripts to segment text into subword units.
- Beautiful-Soup web-crawler (a python implementation) was used to collect data.
- Used Python modules sci-py, matplotlib, statsmodel to generate graphs and charts in the evaluation.
- Used the antpc server in UCSC for training.
- Google colab used for building the suitable environment to doing training and evaluation process. Appendix B.2 and B.3 for colab training.

4.5 Summary

This chapter we discussed a data-collection technique, preprocessing technique, machine learning techniques and back-translation techniques we have used to improve translation accuracy. Their contributions and effect on the translations are discussed and analyzed in Chapter 5. We have also made aware of the tools, frameworks, GPU specifications that enabled the implementation of our experiments.

Chapter 5

Results and Analysis

In the chapter, we describe the translation accuracy obtained with each technique discussed in Chapter 3. We elaborate the approach for our research questions by providing the justification and conclusion for observed results. Then the reader can have an idea of how BLUE score varies in the approaches.

5.1 Results obtained from the First Research Question.

Before analysing the Sinhala-Tamil Language pair we used Finnish-English (Fi-En) corpus for getting the idea of OpenNMT Framework work-flow. In chapter 3, we discussed the pre-processing technique using BPE and Deep Neural Network models. To find out the best pre-processing technique and DNN model for research question one, we used authentic 25000 parallel Sinhala-Tamil language pair corpus to train the models. Table 5.1 shows corpus details for using the first step.

Table 5.1: Corpus of the Parallel Dataset

Corpus Statistics	Sinhala	Tamil
Number of sentence pairs	26,187	26,187
Total Number of Words (T)	262,082	227,486
Vocabulary Size(V)	38,203	54,543
V/T %	14.58	23.98

5.1.1 Using full- word form for training

Firstly we used full word form for checking the translation. Under the full-word form pre-processing,we used the table 5.1 data for training.

Deep Neural Model: Long Short Term Memory(LSTM)

Table 5.2: Architecture details:LSTM

Layers	2
Hidden Unites (both encoder and decoder)	512
Training Steps	100 000
Bath Size	64

As shown in Table5.2, the whole LSTM architecture is with two main layers encoder and decoder with 500 hidden units.

Deep Neural Model:Transformer

As we have dissucced in Chapter 3, we came to conclude that the transformer model will be the most effective model for the sequence to sequence machine translation. Therefor Table 5.3 shows that the hyper-parameters that we used for our experiment with the transformer model.

Table 5.3: Architecture details:Transformer

Encoding layers	6 identical layers
Decoding layers	6 identical layers
Training steps	100 000
Label smoothing	0.1

5.1.2 Pre-processing with Byte Pair Encoding.

The main hyperparameter in BPE is merging operation. As we have discussed in Chapter 4, a higher value to merge operation would lead to character level segmentation and low value of the merge operation will lead to word-level.

Researchers have introduced the **30 000** is the good starting point to the corpus including with a large number of rare words. They have observed that this value is suitable for corpus which has a size above 1 million.

If we consider the dataset Sinhala-Tamil languages pair that we are using to our research, those are morphologically rich language. It is self-sustaining with more rare words. Then using BPE, we were able to address the identification of rare words. Our parallel corpus data has 25000 sentence pairs.

At the current moment, we used parallel corpus data to check the identification of the two different model architectures and pre-processing technique. Therefore, the dataset is lower than 1 million, we used lower value **3000** to the merge operation to the dataset concerning our data corpus after experimenting with different merge operations values.

Table 5.4: Original word with its' BPE segmentation

Original Word	BPE segmentation
රියැදුරන්	රියැ@@ දුර@@ න්
ඉංග්‍රීසි	ඉංග්‍රී@@ සි@@
පල්ලිය	පල්@@ ලිය
බුද්ධාගම	බු@@ ද්@@ ධා@@ ගම

Table 5.4, shows that how BPE segment the word. The words represent how BPE segments each word. Through BPE, we can increase our total number of the word in data corpus. BPE can reduce the morphological richness of word. Below example sentence represents the sentences pair that segment with BPE.

Example Sentence for Byte pair encoding

- Sinhala : බු@@ ද්@@ ධා@@ ගම නො වන්නට ඉන්දිය@@ ාවට පෙ@@ න්@@ වීමට ඉතිහාස@@ යක් නො වන්නට තිබි@@ ණ@@ ැ@@ යි මත@@ යක් ද ඇත .
- Tamil : ெள@@ த்த ம@@ தம் இல்லா@@ து இருந்த@@ ால் இந்தியா@@ வில் ச@@ ரி@@ த்திர@@ ம் என வழ@@ ிய@@ ு@@ று@@ த்துவ@@ தற்கு ச@@ ரி@@ த்திர@@ ம் ஒன்ற@@ ும் இல்லா@@ மல் போ@@ ய@@ ிரு@@ க்கும் என்ற கருத்த@@ ு இுக்கிறது .

Table 5.5: Error Analysing in parallel Corpus (Full-Word form)

Description	Sinhala		Tamil	
	UniqW	TotalW	UniqW	TotalW
Training Dataset	35 593	227 447	50 002	197 228
Testing Dataset	5 390	13 866	6 141	10 765
OOV%	19.38	-	29.57	-

Table 5.5 represents how the word counts vary from testing and training data corpus. we can recognise that the total number of words count in training dataset in the Sinhala language is higher than the total number of words in the Tamil language. In addition to that, the testing dataset words count exposes the same result as the training dataset. When analysing the unique word count in both

training and testing dataset, gives the higher value to the Tamil language than the Sinhala language.

This is an important finding in the understanding of the Out-of-Vocabulary (OOV) ratio. We consider the testing dataset for OOV. Table 5.5 gives the value nearly 19% for the unique words are Out-of-vocabulary in the Sinhala language testing datasets and 30% of unique testing words are Out-of-vocabulary in Tamil language testing dataset. It is proved that translation getting lower BLEU score for Sinhala to Tamil rather than Tamil to Sinhala.

Table 5.6: Error Analysing in parallel Corpus (BPE)

Description	Sinhala		Tamil	
	UniqW	TotalW	UniqW	TotalW
Training Dataset	3 158	423 980	3 184	398 203
Testing Dataset	2 868	22 641	2 798	22 035
OOV%	0	-	0	-

Another promising finding was that the OOV ratio with the BPE segmentation. According to Table ed, we can see the Sinhala language has a higher value to the total number of words in training and testing datasets than the Tamil language. Here we can compare the results of the total number of words in training and testing dataset with those of the based methods the full word form. As a result of this comparison, BPE increases the datasets by approximate trice in the dataset of full-word form.

In line with previous studies, we can see that BPE segmentations of both languages gives roughly similar value to the number of unique words in both testing and training dataset and it receives a lower value concerning the table 5.5 values. The promising result is showing the OOV value in BPE. It takes 0% values for both unique and total words. This exposes that the BPE can use for NMT for open-domain translation. Finally, we can conclude that using BPE for the pre-processing technique answers the research problems with handling Out-Of-Vocabulary.

Table 5.7: LSTM Architecture Outputs (BPE)

	Reference Sentences	Output Sentences
01	ඒ ගැන සොයා බලන්නැයි ජනාධිපතිවරයා සරත් අමුණුගම ඇමැතිවරයාට උපදෙස් දුන්නේය .	ජනාධිපතිවරයා ඒ ගැන හොයන්නැයි ඇමැතිවරයා සරත් අමුණුගයට උපදෙස් දුන්නේය .
02	සිළුමිණ සමඟ සම්මුඛ සාකච්ඡාවකට එක් වෙමින් පෙරේරා මහතා මේ ප්‍රකාශය කළේය .	සිළුමිණ සමග සම්මුඛ සාකච්ඡාවකට එක්වෙමින් කරමින් පෙරේරා මහතා මේ ප්‍රකාශය කළේය .
03	මේ සියල්ල ශ්‍රී ලංකා නිදහස් පක්ෂයට සේම ආණ්ඩුවටත් බලපාන දේශපාලන ගැටලු ය .	මේ සියල්ල ශ්‍රී ලංකා නිදහස් පක්ෂයත් විසින් රාජ්‍ය හා සම්මුතිය වේ .

Table 5.8: Transformer Architecture Outputs (BPE)

	Source Sentences	Output Sentences
01	ඒ ගැන සොයා බලන්නැයි ජනාධිපතිවරයා සරත් අමුණුගම ඇමැතිවරයාට උපදෙස් දුන්නේය .	ජනාධිපතිවරයා ඒ ගැන හොයන්නැයි ඇමැතිවරයා සරත් අමුණුකුල උපදෙස් දුන්නේය .
02	සිළුමිණ සමඟ සම්මුඛ සාකච්ඡාවකට එක් වෙමින් පෙරේරා මහතා මේ ප්‍රකාශය කළේය .	සිළුමිණ සමග සම්මුඛ සාකච්ඡාවකට එක්වෙමින් පෙරේරා මහතා මේ ප්‍රකාශය කළේය .
03	මේ සියල්ල ශ්‍රී ලංකා නිදහස් පක්ෂයට සේම ආණ්ඩුවටත් බලපාන දේශපාලන ගැටලු ය .	මේ සියල්ල ශ්‍රී ලංකා නිදහස් පක්ෂයට මෙන්ම ආණ්ඩුවට බලපාන ගැටලුවෙකි .

From the result in Table5.7, shows the output sample sentences using BPE for given test data using the LSTM model.
Table 5.8, Shows the output sample sentences using BPE for given test sentences using the Transformer model.

Table 5.9: BLEU Score of full word form and Byte pair encoding

Architectures	Full-word-form [Tamil-Sinhala]	BPE [Tamil-Sinhala]
LSTM	7.85	9.52
Transformer	8.01	11.18

The BLEU scores obtained for the directions with full-word form and BPE are shown in Table 5.9.

In BLEU score we used for evaluating the machine translation research. Based on the observed valued, we decided that applied BPE pre-processing to data corpus increase the BLEU score concerning the full-word form data corpus in both two models(LSTM, transformer.) Through this observation, we applied BPE for our further experiments.

When selecting the model for moreover experiment in the first research question, we concluded that the transformer model performs well rather than LSTM. Table 5.9 values of BLEU score show that. Table 5.7, 5.8 represents some test sentences in both models. The translations did not relate the semantics of the reference sentences. Only a few words from each sentence were correctly translated but it did not add any value to the underlying meaning of the sentence in BLEU score. Such semantically correct translation outputs are given in Table 5.8 third sentences. We deeply investigated the output corpus for finding the semantically similar sentences. Then we found more sentences in semantically equal in transformer model output. Third sentence in both tables 5.7 and 5.8 elaborated it. You can see the transformer model gave the output sentences approximately similar its' meaning to the reference sentence s' meaning. When we compare the reference sentence with LSTM model output sentence it shows you that both reference and output sentence have two different meaning. Then finally we can conclude that the transformer model supports well for NMT in getting accurate result for semantic way also.
According to the analysis of the training values, we selected BPE for preprocessing techniques and transformer architecture for answering the first research question further experiment.

Before applying the back translation techniques, we investigated the performance while increasing the authentic parallel corpus data. In here first round we checked using 1000 sentences pairs. we trained it using LSTM and Transformer model. Likewise, we increased 1000, 2000, 4000, 5000 ,10 000, 15 000 and 25 000 sentences pairs eventually. At the begin of 1000, it didn't show the promising result due to the lack amount of data. When it eventually reached 25 000 sentences pairs, it ended up the results in representing in table 5.9.

5.2 Result with Back-Translation technique

5.2.1 Normal Back-Translation

In chapter 3, we deeply discussed the step of the back-translation technique. In the beginning, we used the same parallel corpus data 25k sentences pairs in table 3.1 and gradually, expand the dataset using monolingual corpus data table 3.2. Normally, Deep Learning methods show encouraging results when the dataset is increasing. DNNs called as the data hunger models.

Ultimately, we selected the best pre-processing technique and DNN model for employing the back-translation in NMT for the Sinhala-Tamil languages pair.

Table 5.10: BLEU Score for Back translation.

Ratio between authentic parallel sentences : synthetic parallel sentences	Tamil-Sinhala	Sinhala-Tamil
1:1	11.18	4.02
1:2	13.53	4.09
1:3	14.68	

Table 5.10 shows the results of the back-translation technique. The results demonstrate the enormous gap in BLEU score in between 1:1 and 1:2 ratios.

The 1:2 ratio gives the best result in the BLEU score for the Tamil to Sinhala translation so far.

In here, we stop our translation in 1:3 stage, because when comes to Sinhala-Tamil in 1:2 ratio, it gave the BLEU score which similar to the 1:1. It did not show the prominence improvement.

Using only the BLEU score we unable to find the accuracy of the semantic of the sentences. Then we analyzed test data by manually for checking the semantically similar sentence and evaluate the model through the semantic aspect also. Figure 5.1 visualizes the behaviour of BLEU score with increasing size of the data corpus.

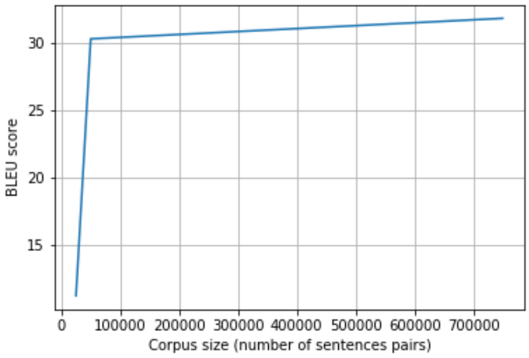


Figure 5.1: BLEU score with corpus size.

5.2.2 Semantic analysing with final models

Model given by 1:2 ratio

Through this method, we created synthetic parallel corpus data using 25000 sentences pairs from the authentic parallel corpus and 50000 sentences pairs for monolingual corpora data. Whole data corpus size is 75000 sentences pairs. For 1:3 ratio, we used 25000 sentences pairs of authentic parallel corpurs and 75000 monolingual corpora data. then whole dataset getting 100 000 sentence pairs

The results of the experiment found in choosing BPE to merge operations, clearly support for this to set our merge operation vale 10000, because of our dataset approximately to 1 million. Table 5.12 represent the example of test sentences which give correct semantic for the reference sentences in final model in back-translation.

Table 5.11: 1:3 Ratio model output sentences

Source Sentences	**Output Sentences**
බුද්ධාගම නො වන්නට ඉන්දියාවට පෙන්වීමට ඉතිහාසයක් නො වන්නට තිබිණැයි මතයක් ද ඇත .	බුදු දහම නැති නම් ඉන්දියාවේ ඉතිහාසය ගැන අවධානය යොමු කිරීමට හැකියාවක් නැති බවයි .
ටයිටැනික් නැවෙන් ගොඩටගත් භාණ්ඩ ප්‍රදර්ශනය කිරීමට විශේෂ ජංගම ප්‍රදර්ශන සේවයක් තිබේ .	ටයිටැනික් නැවෙන් ගොඩටගත් භාණ්ඩ ප්‍රදර්ශනය කිරීමට විශේෂ ජංගම ප්‍රදර්ශන සේවයක් තිබේ .

Experiment with the Out of data set

In here, we translated some sentences which are not in test corpus. The result will shows in table aa.

Table 5.12: Experiment with the sentences of Out of dataset

Source Sentences	**Output Sentences**
එහි ඇති වූ පිපිරීම් හේතුවෙන් වැලි කඳු වැටී ඇති බව ප්‍රදේශවාසීහු පැවසූහ	එහි දී පවතින පිපිරීම නිසා වැලි කඳු කඩා වැටී ඇතැයි ප්‍රදේශවාසීහු පවසති .
මෙහි රඳවා සිටින බොහෝ සරණාගතයින් අනාගතය පිළිබඳ කනස්සල්ල මධ්‍යයේ ආරක්ෂිත රටක නැවත පදිංචි කරවන ලෙස ඉල්ලා සිටිති	මෙකී පාඩම් ගණනාවක් සරණාගත කඳවුරු වල ඇති ශෝක ප්‍රකාශන ලෙස රටක ආරක්ෂාව සැලකිලිමත් විය .

5.2.3 Conclusion

Based on the received values, Sinhala-Tamil translation has confirmed promising result using BPE techniques, back translation and the transformer architecture. Morphologically rich languages can be treated using BPE for reducing the complex nature of the word. BPE reasoned to reduces the ’OOV’ words in the dataset. Using marked data improve the accuracy of the translation and increasing the dataset improves the neural machine translation training. Using Transformer architecture for translation will address the problem of long term dependency.

5.3 Results obtained from the Second Research Question.

In this research question, we exposed result through monolingual corpora data. We have 1 000 000 sentences with Sinhala and 400 000 sentences Tamil sentences. Our aim to achieve some considerable result through this corpora. Table 5.13 shows the details of the monolingual corpora that we are going to used.

Table 5.13: Corpus of the Monolingual Dataset

Corpus Statistics	Sinhala	Tamil
Number of sentence pairs	1,067,173	407,578
Total Number of Words (T)	13,158,152	4,178,440
Vocabulary Size	933,153	301,251

After successfully addressing the first research question, we satisfied with the OpenNMT-py Framework. As a result of our satisfaction, we had a curiosity to investigate deeply in OpenNMT-py for Generative Adversarial Network. We discussed with the authors and contributors of OpenNMT-py for recognising the strength to create GAN architecture using OpenNMT-py framework [1]. we awakened to OpenNMT-py framework is not supporting to the GAN architecture.
Hereafter, we tried the GAN approach with NMT-GAN and UndreaMT[2] frameworks and MASS. Firstly we tried to examined the MASS approach and we observed its' performance for their given dataset.It performed well for their pre-trained and fine-tuned models of English and Finnish. After that we investigated its' architecture and how they input their sentences.When mapped to the Sinhala-Tamil and languages they have mentioned that to create own sourced codes for languages which hasn't pre-trained models. Hereafter we tried to prepare the source code for Sinhala-Tamil , but it didn't work properly. After that we had discussion with the author about how they prepared the source code.Until receiving a answer from him, we examined other approached that we can achieved.

[1] https://opennmt.net/

[2] https://github.com/artetxem/undreamt

In addition to that, we drove our direction to examine whether it is possible to take the unsupervised approach without using GAN and we tried to use encoder-decoder based architecture to training the data in an unsupervised way.

5.3.1 Architecture Detail of UndreaMT

Table 5.14: Architecture details:UndreaMT

Shared Encoding layers	2 layers Bidirectional RNN
Decoding layers	2 identical layers RNN
Training steps	300 000 or 100 000

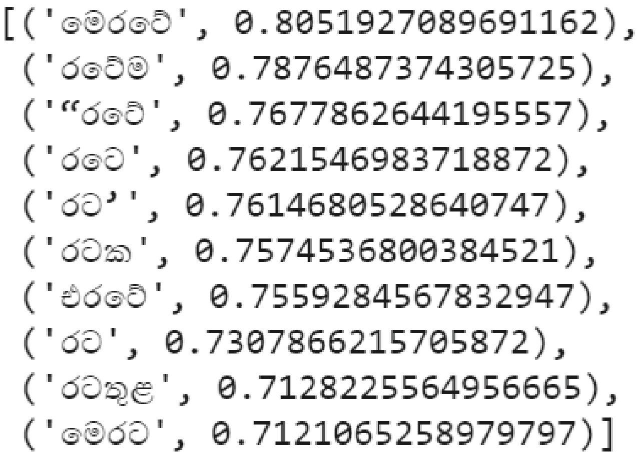
Figure 5.14 represents the details of the architecture. When we used the 300 000 iteration, it took 3 days for the training process. After analysing the time of processing we decided to use 100 000 iteration for train, based on the size of the monolingual corpora.

5.3.2 Data preparation

Before using the raw data, we remove the special characters and symbol in the both data-set. After that we remove the repeating sentences in the data-sets.Our Experiments are mainly dividing into two-part based on the dataset pre-processing technique.

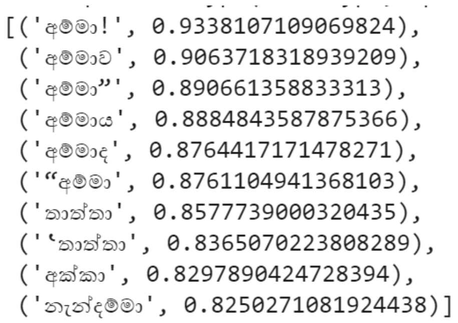
In the first part, full word form segmentation dataset was used for experiments. In the second part, the BPE segmentation technique used for pre-processing.

5.3.3 Result analysis for Full word form



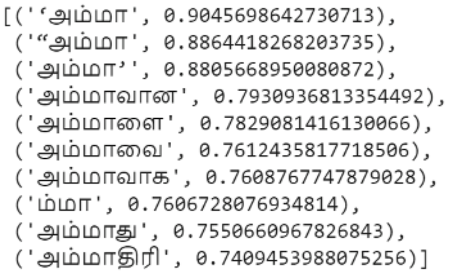
```
[('මෙරටේ', 0.8051927089691162),
 ('රටේම', 0.7876487374305725),
 ('“රටේ', 0.7677862644195557),
 ('රටෙ', 0.7621546983718872),
 ('රට’', 0.7614680528640747),
 ('රටක', 0.7574536800384521),
 ('එරටේ', 0.7559284567832947),
 ('රට', 0.7307866215705872),
 ('රටතුළ', 0.7128225564956665),
 ('මෙරට', 0.7121065258979797)]
```

Figure 5.2: Most similar words represents for word රටෙ Word embedding for the data corpora.



```
[('අම්මා!', 0.9338107109069824),
 ('අම්මාව', 0.9063718318939209),
 ('අම්මා”', 0.890661358833313),
 ('අම්මාය', 0.8884843587875366),
 ('අම්මාද', 0.8764417171478271),
 ('“අම්මා', 0.8761104941368103),
 ('තාත්තා', 0.8577739000320435),
 ('‘තාත්තා', 0.8365070223808289),
 ('අක්කා', 0.8297890424728394),
 ('නැන්දම්මා', 0.8250271081924438)]
```

Figure 5.3: Most similar words for 'අම්මා' in sinhala



```
[('‘அம்மா', 0.9045698642730713),
 ('“அம்மா', 0.8864418268203735),
 ('அம்மா’', 0.8805668950080872),
 ('அம்மாவான', 0.7930936813354492),
 ('அம்மாளை', 0.7829081416130066),
 ('அம்மாவை', 0.7612435817718506),
 ('அம்மாவாக', 0.7608767747879028),
 ('ம்மா', 0.7606728076934814),
 ('அம்மாது', 0.7550660967826843),
 ('அம்மாதிரி', 0.7409453988075256)]
```

Figure 5.4: Most similar words for in tamil

Both of the above figure 5.3 and 5.4 represent the word embedding representation of similar words to 'අම්මා'(Mother) in Sinhala and Tamil languages .

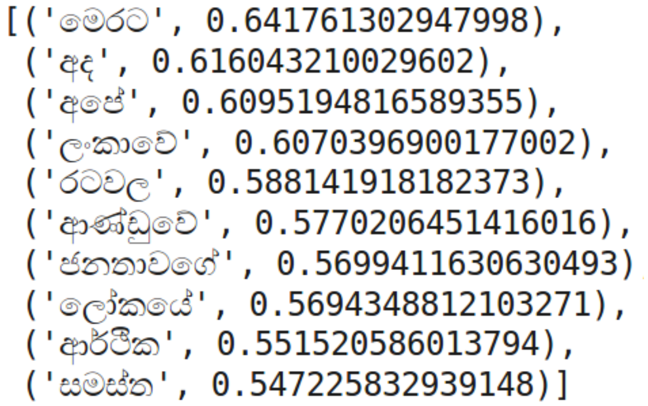
For a instance , figure 5.3 you can see the similar words to 'අම්මා'(Mother) and at the same time it representතාත්තා '(Father) word proving word embedding can identify the pair of male-female relation. In addition to, that it displays the approximately similar words gather around Mother.

Error analysis for the full-word form

Table 5.15: Error analysis for the full-word form

Source Sentences	Output Sentences
බුද්ධාගම නො වන්නට ඉන්දියාවට පෙන්වීමට ඉතිහාසයක් නො වන්නට තිබිණැයි මතයක් ද ඇත .	<OOV> <OOV> <OOV> <OOV> <OOV> ගැන <OOV> <OOV> <OOV> <OOV>
ටයිටැනික් නැවෙන් ගොඩටගත් භාණ්ඩ ප්‍රදර්ශනය කිරීමට විශේෂ ජංගම ප්‍රදර්ශන සේවයක් තිබේ .	<OOV> <OOV> <OOV> කිරීමට <OOV> <OOV> <OOV> <OOV> <OOV>

5.3.4 Using Byte Pair Encoding for segmentation



```
[('මෙරට', 0.641761302947998),
 ('අද', 0.616043210029602),
 ('අපේ', 0.6095194816589355),
 ('ලංකාවේ', 0.6070396900177002),
 ('රටවල', 0.588141918182373),
 ('ආණ්ඩුවේ', 0.5770206451416016),
 ('ජනතාවගේ', 0.5699411630630493)
 ('ලෝකයේ', 0.5694348812103271),
 ('ආර්ථික', 0.551520586013794),
 ('සමස්ත', 0.547225832939148)]
```

Figure 5.5: Most Similar words represents for word රටෙ Word embedding for the data corpora BPE segmentation.

Figure 5.5 represents the word embedding behaviour with the BPE segmentation. BPE segmented word embedding and full word form segmentation represent two different word sets for same word. When comparing the full word form embedding with the BPE segmented word embedding, figure 5.2 represent the words only similar with given word rata. However, when coming to the BPE segmented word embedding, figure 5.5 displays word sets with a given word mostly associated.It shows that the words which are around with the given words. It reasoned to increase the translation accuracy. Hence, the architecture can identify the relative words when translating the sentences.

5.3.5 Error Analysis in BPE segmented word embedding.

In this novel approach, we examined the translation accuracy by checking the synonyms rather than getting a BLEU score. BLEU score can not measure the semantically similar phrases. Then we manually checked the tests data for finding the semantically similar words.

Table 5.16: Error analysis for the full-word form

Source Sentences	**Output Sentences**
ප්‍රේමය විසින් මෙහෙයවනු ලැබූ ඇය ඔහු සමග එක් ඍයෙක පලා ගියා ය .	අන්‍යයෝ ඔහුගේ බිරිඳ හෝ මා මගේ සිත විය <OOV>
ඔවුහු අනුන්ගේ බස් පුන පුනා කියන ගිරවුන් මෙන් වෙති .	අපි ආපසු ගොස් ඒ සියල්ල පිළිබඳ තොරතුරු සොයා ගතිමි <OOV>
ශ්‍රී ලංකාව වැනි රාජ්‍යයක් ගොඩනැඟීම සඳහා ජනමූල නායකයකු අවශ්‍ය වේ .	ශ්‍රී ලංකා ආණ්ඩුවේ මහජන සංවිධානයේ මෙහෙයවීමෙන් යළි පත් විය <OOV>

Table 5.16, displays the result sentences given by the BPE segmented word embedding approach. It is very challenging to generate correct words phase without training the labelled dataset. At a glimpse, it seemed that there is no relationship of these reference sentence and translated output. Hence we examined the synonyms of the sentences with came up with some likelihood to achieve translation using monolingual corpora data.

01 Reference: ශ්‍රී ලංකාවේ ජනමාධ්‍ය පදනම ඔහුට සම්මානයක් ප්‍රදානය කිරීම බෙහෙවින් පැසසිය යුතු කාරණයෙකි .

Translated :ශ්‍රී ලංකාව පාලනය කරන බදු අනුපාත මිල එහි ප්‍රතිඵලයක් ලෙස සැලකේ <OOV>

02 Reference : එහෙත් ආරක්ෂක විෂය සඳහා වෙනමම අමාත්‍යවරයකු අවශ්‍ය බව ඉහත තත්ත්වයන් අනුව වැටහිණි .

Translated : එහෙත් සාධාරණ ලෙස කිසිදු ජීවිතයක් ලබා ගැනීමට හැකියාවක් ලැබෙන අතර සමහර

විට එය වෙනස් විය <OOV>

03 Reference : ශ්‍රී ලංකාවෙ දෙවන නිදහස් අරගලය හැල්ලු කිරීමට ඉඩ නො දෙමු .
Translated: ශ්‍රී ලංකාවෙ ආරක්ෂක බල වර්ජනය අරඹා තිබේ <OOV>

As a novel approach to building a translation for Sinhala and Tamil languages pair only using monolingual corpora shows the promising results of 3 reference sentences. Although Those references sentences received the wrong sentences as output, It s' output sentences have well build sentences based on the POS tagging.This indication illuminates the hope of building the translation using monolingual corpora to morphologically rich and low resourced languages pair.

POS tag explanation example:
අපි ආපසු ගොස් ඒ සියල්ල පිළිබඳ තොරතුරු සොයා ගතිමි

- අපි:Noun(Subject)
- ඒ සියල්ල : Object
- සොයා ගතිමි :Verb

5.3.6 Conclusion

When compared to the full word form with the BPE approach it significantly shows through the results the BPE perform well rather than full word form. Through the second research question, we demonstrated that using only monolingual corpora has the ability to implement a translation between Sinhala-Tamil languages pair.

5.3.7 Overall observation and Discussion

Through our experiments, we observed implement a translation in Tamil to Sinhala performed well rather than implementing the translation Sinhala to Tamil. By reducing the inflexion morphe in both language reason to reduce the morphological richness and reduced the vocabulary size. Therefor address the rare words while translating performs well.

In parallel corpora confirmed that the Tamil language is morphologically richer than the Sinhala language. Because of it V to T ratio almost two times than Sinhala language V to T ratio. When com to the Monolingual corpora Sinhala has V/T ratio is 7.0% and Tamil has 7.2%. Based on the NMT architecture encoder performed well when employing higher morphologically rich language as the source language. Because When the source-side is morphologically richer than the target-side, the encoder tends to encode more information about the sentence, leading to better decoding by the decoder. Overall First research question, we are addressing the problem using parallel and monolingual corpora data. It shows the promising results for Sinhala-Tamil translation. It justified the Tamil to Sinhala translation performed well than Sinhala to Tamil translation.

Doing the second research question, we addressed only monolingual corpora. In here, both V to T ratio is approximately the same, then we examined that the translation has low performance as a translator. As an observation of the experiment, we concluded that there is still availability for Sinhala and Tamil languages can use only monolingual corpora for translation.

Chapter 6

Conclusions

This dissertation is on developing an NMT system for improving the translation between the morphologically rich and low resource language pairs Sinhala and Tamil by using monolingual corpora data. This chapter presents an overall representation of the conclusions drawn from the overall research work conducted by us.

6.1 Conclusions about research questions

The main aim is to develop more sophisticated methods for implementing Sinhala-Tamil translator. For our first goal, we focus on one main problem Sinhala and Tamil are considered as low resourced languages and they have limited parallel corpus data. Most of the research on Natural Language Processing in Neural Machine Translation has achieved an impressive result through parallel corpus dataset. Low resourced languages confront with low performance due to the lack amount of parallel corpus data. Creating parallel corpus for language pair is more expensive and needs the persons who are expert knowledge for both languages. Sinhala and Tamil consider as low resourced languages due to the lack of linguistics references. Conversely, monolingual corpora data is much easier to find than parallel corpus data and many languages with a limited amount of parallel corpus may perform prominent result with monolingual corpora.Inspiring these methodologies we addressed our first research question as 'What is the effect of the Monolingual Corpora on the translation accuracy? ' for examing the performance of translation in between Sinhala and Tamil language with both monolingual and parallel corpora. we examined the morphological richness in both languages and apply the BPE word segmentation for reducing the inflection morpheme. The performance of the BPE and full word form we discussed in Chapter 5. Both of the approch, BPE segmentation perform well for translation. In table 5.6, represents the details of the vocabulary may change after employing the BPE to the dataset. In line with previous studies, we can see that BPE segmentations of both languages gives roughly similar value to the number of unique words in both testing and training dataset and it receives a lower value concerning the table 5.5 values.

The promising result is showing the OOV value in BPE. It takes 0% values for both unique and total words. This exposes that the BPE can use for NMT for open-domain translation. Finally, we concluded that using BPE for the pre-processing technique answers the research problems with handling Out-Of-Vocabulary. After finding the proper segmentation technique, we chose transformer architecture for doing our further experiments after doing many experiments with LSTM model.

Up to this stage, we used only parallel corpora. after finalizing the proper techniques we tried to find a way for increasing the parallel corpus. Back translation was the techniques we used in the increasing the parallel corpus with the help of the monolingual corpora. After coming to the end stage of the first research question we examined that using both parallel and monolingual corpora reasoned to increase the performance in the translation.

As the inspiring, the first research question resulted, we were heading to find a way to develop a translator using only monolingual corpora data. Then our second research question is ”What is the best approach for achieving high accuracy in Sinhala-Tamil translation using only monolingual corpora?”

To address this research question, we experimented in many ways. First, we started with the GAN architecture. But it used parallel corpus data to discriminator model train. Our intention was to use only monolingual corpora data then we drove our direction to adversarial networks by inspiring from GAN. Finally, we came up with the architecture using the encoder-decoder approach. When compared to the full word form with the BPE approach it significantly shows through the results the BPE perform well rather than full word form.

Through the second research question, we demonstrated that using only monolingual corpora has the ability to implement a translation between Sinhala-Tamil languages pair.

Finally we concluded that, the due to the morphological richness in Tamil language than the Sinhala Language Tamil to Sinhala translation perform well rather than to Sinhala to Tamil. we discussed it in Chapter 05.

6.2 Limitations

The considerable limitation that we are come a cross to research is limiting the iterations of training based on the fixed size considering the dataset size and used the fixed size merge operation to the BPE according to the corpus size.

6.3 Implications for further research

In our research we focused on mainly semi-supervised and unsupervised approach for addressing the morphological rich and low resourced languages. Through our research we were able to give a possible clue to develop translator using monolingual corpora only. There are many ways we have discuss in our chapter 3. As future works anyone can improve a method suggested by us and try out the another methods such as GAN, MASS for improving translation accuracy for Sinhala-Tamil languages pair. In adition to that , further experiment can be done using wordPiece segmentation methods. WordPiece [1]. is another word segmentation algorithm and it is similar with BPE. Schuster and Nakajima introduced WordPiece by solving Japanese and Korea voice problem in 2012. Basically, WordPiece is similar with BPE and the difference part is forming a new subword by likelihood but not the next highest frequency pair

Many Asian language word cannot be separated by space. Therefore, the initial vocabulary is larger than English a lot. You may need to prepare over 10k initial word to kick start the word segmentation. From Schuster and Nakajima research, they propose to use 22k word and 11k word for Japanese and Korean respectively.

[1] https://medium.com/@makcedward/how-subword-helps-on-your-nlp-model-83dd1b836f46

Bibliography

A. Imankulova, T. S. and Komachi, M. (2017), 'Improving low-resource neural machine translation with filtered pseudo-parallel corpus'.

A. Karakanta, J. D. and Genabith, J. (2017), 'Neural machine translation for low-resource languages without parallel corpora'.

Artetxe, M. and Labaka (2018), 'Unsupervised neural machine translation'.

Arukgoda, A. and Weerasinghe, A. (2018), 'Improving sinhala – tamil translation through deep learning techniques'.

B. Zoph, D. Yuret, J. M. and Knight, K. (n.d.), 'Transfer learning for low-resource neural machine translation'.

Burlot, F. and Yvon, F. (2019), 'Using monolingual data in neural machine translation: a systematic study'.

Currey, A., Miceli Barone, A. V. and Heafield, K. (2017), 'Copied monolingual data improves low-resource neural machine translation'.

D. Bahdanau, K. C. and Bengio, Y. (2014), 'Neural machine translation by jointly learning to align and translate'.

Fadaee, M., Bisazza, A. and Monz, C. (2017), 'Data augmentation for low-resource neural machine translation'.

G. Klein, Y. K., Deng, Y., Senellart, J. and Rush, A. (2017), 'Opennmt: Open-source toolkit for neural machine translation'.

Gangi, M. and Federico, M. (2019), 'Can monolingual embeddings improve neural machine translation'.

Graça, M., Kim, Y., Schamper, J., Khadivi, S. and Ney, H. (n.d.), ‘Generalizing back-translation in neural machine translation’.

Gulcehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H. and Bengio, Y. (2019), ‘On using monolingual corpora in neural machine translation’.

Guzman, F., Chen, P.-J., Ott, M., Pino, J., Lample, G., Koehn, P., Chaudhary, V. and Ranzato, M. (2019), ‘Two new evaluation datasets for low-resource machine translation: Nepali-english and sinhala-english’.

Hans, K. and Milton, R. (2016), ‘Improving the performance of neural machine translation involving morphologically rich languages’, *arXiv preprint arXiv:1612.02482* .

Ian J. Goodfellow, J. P.-A., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014), ‘Generative adversarial nets’.

Lample, G., Conneau, A., Ranzato, M., Denoyer, L. and Jégou, H. (2018), ‘Word translation without parallel data’.
URL: *https://openreview.net/forum?id=H196sainb*

Lample, G., Denoyer, L. and Ranzato, M. (2018), ‘Unsupervised machine translation using monolingual corpora only’, *ICLR* .

M. Artetxe, G. L. and Agirre., E. (2018), ‘A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings’, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* **Volume 1: Long Papers**.

M. Luong, H. P. and Manning, C. D. (2015), ‘Effective approaches to attention-based neural machine translation’.

Nguyen, T. Q. and Chiang, D. (2017), ‘Transfer learning across low-resource, related languages for neural machine translation’.

Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J. (2002), ‘Bleu: a method for automatic evaluation of machine translation’.

Passban, P. (2017), ‘Machine translation of morphologically rich languages using deep neural networks’.

Renduchintala, A., Shapiro, P., Duh, K. and Koehn, P. (2019), ‘Character-aware decoder for translation into morphologically rich languages’.

R.Pushpananda, Weerasinghe, R. and M.Niranjan (2014), ‘Sinhala-tamil machine translation: Towards better translation quality’.

Sennrich, R., Haddow, B. and Birch, A. (2016), ‘Neural machine translation of rare words with subword units’.

Sennrich, R., Haddow, B. and Birch, A. (n.d.), ‘Improving neural machine translation models with monolingual data.’.

Song, K., Tan, X., Qin, T., Lu, J. and Liu, T. (2019), ‘MASS: masked sequence to sequence pre-training for language generation’, *CoRR* **abs/1905.02450**.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I. (2017), ‘Attention is all you need’.

Weerasinghe, R. (2004), ‘A statistical machine translation approach to sinhala tamil language translation’.

Zhang, J. and Zong, C. (2016), ‘Exploiting source-side monolingual data in neural machine translation’.

Zhang, Z., Liu, S., Li, M., Zhou, M. and Chen, E. (2018), ‘Bidirectional generative adversarial networks for neural machine translation’.

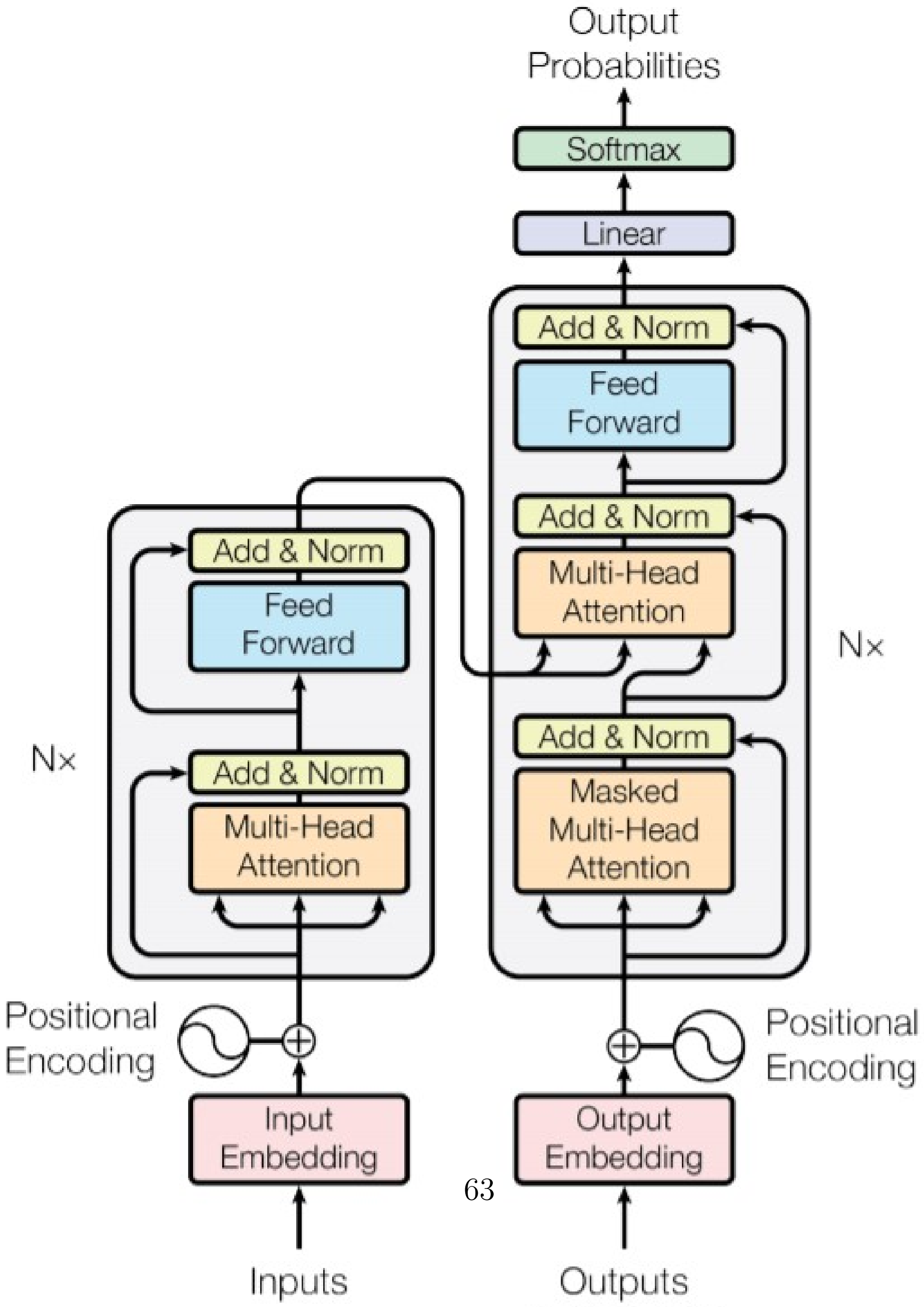
Zoph, B., Yuret, D., May, J. and Knight, K. (2016), ‘Transfer learning for low-resource neural machine translation’.

Appendices

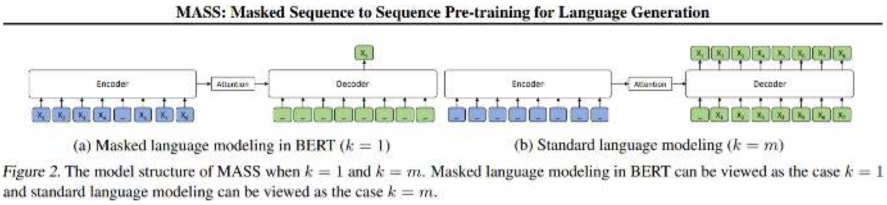
Appendix A

Diagrams

A.1 Transformer Architecture



A.2 MASS architecture



MASS: Masked Sequence to Sequence Pre-training for Language Generation

(a) Masked language modeling in BERT ($k = 1$)

(b) Standard language modeling ($k = m$)

Figure 2. The model structure of MASS when $k = 1$ and $k = m$. Masked language modeling in BERT can be viewed as the case $k = 1$ and standard language modeling can be viewed as the case $k = m$.

Figure A.2: MASS:Pretraing architecture: from(Song et al., 2019)

A.3 GAN architecture

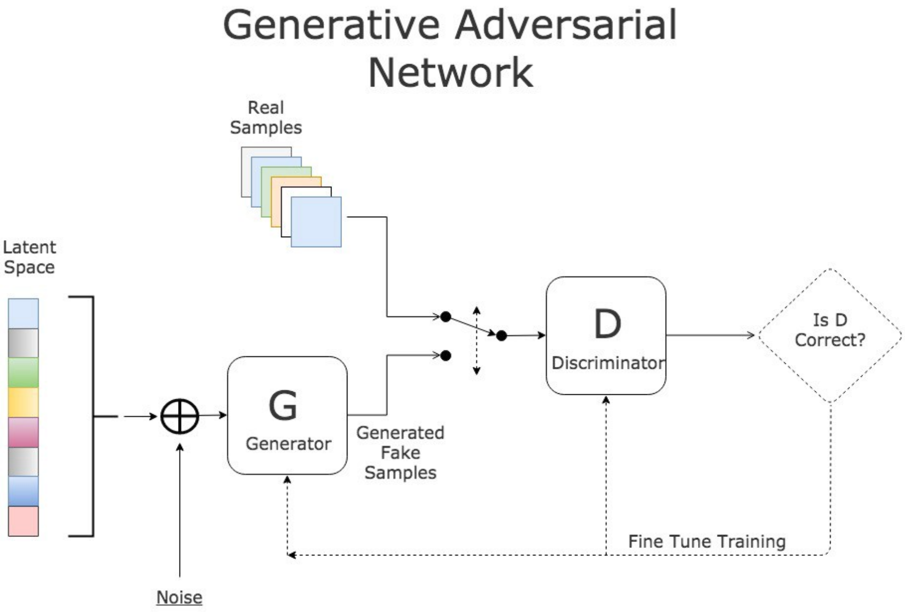
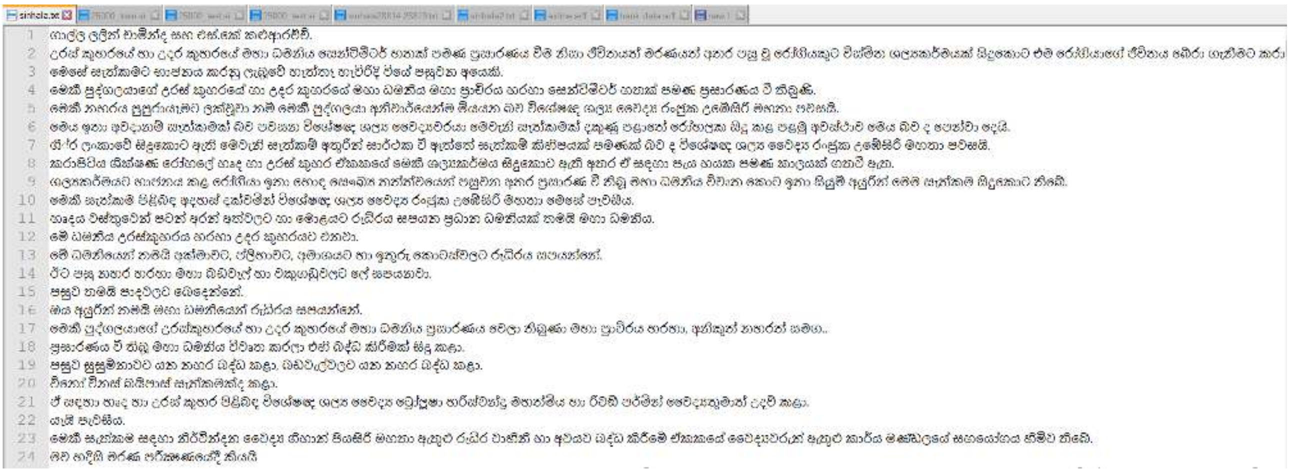


Figure A.3: Genarative Adversarial Network architecture

Appendix B

Interfaces

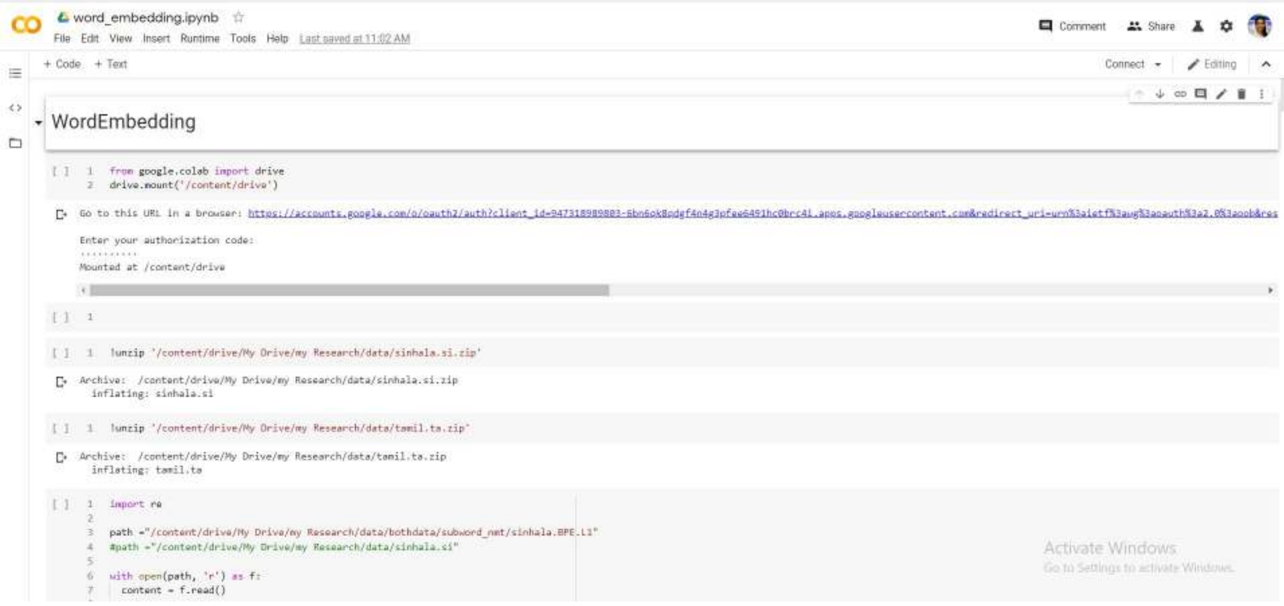
B.1 Collected data sample



```
ගාල්ල ලලිත් චාමින්ද සහ එස්.කේ කළුආරච්චි.
උරස් කුහරයේ හා උදර කුහරයේ මහා ධමනිය සෙන්ටිමීටර් හතක් පමණ ප්‍රසාරණය වීම නිසා ජීවිතයත් මරණයත් අතර පසු වූ රෝගියකුට විස්මිත ශල්‍යකර්මයක් සිදුකොට එම රෝගියාගේ ජීවිතය බේරා ගැනීමට කරා
මෙසේ සැත්කමට භාජනය කරනු ලැබුවේ හැත්තෑ හැවිරිදි වියේ පසුවන අයෙකි.
මෙකී පුද්ගලයාගේ උරස් කුහරයේ හා උදර කුහරයේ මහා ධමනිය මහා ප්‍රාචීරය හරහා සෙන්ටිමීටර් හතක් පමණ ප්‍රසාරණය වී තිබුණි.
මෙකී තත්වය පුපුරායෑමට ලක්වුවා නම් මෙකී පුද්ගලයා අනිවාර්යෙන්ම මියයන බව විශේෂඥ ශල්‍ය වෛද්‍ය රංජුක උමේසිරි මහතා පවසයි.
මෙය ඉතා අවදානම් සැත්කමක් බව පවසන විශේෂඥ ශල්‍ය වෛද්‍යවරයා මෙවැනි සැත්කමක් දකුණු පළාතේ රෝහලක සිදු කළ පළමු අවස්ථාව මෙය බව ද පෙන්වා දෙයි.
ශ්‍රී ලංකාවේ සිදුකොට ඇති මෙවැනි සැත්කම් අතුරින් සාර්ථක වී ඇත්තේ සැත්කම් කිහිපයක් පමණක් බව ද විශේෂඥ ශල්‍ය වෛද්‍ය රංජුක උමේසිරි මහතා පවසයි.
කරාපිටිය ශික්ෂණ රෝහලේ හෘද හා උරස් කුහර ඒකකයේ මෙකී ශල්‍යකර්මය සිදුකොට ඇති අතර ඒ සඳහා පැය හයක පමණ කාලයක් ගතවී ඇත.
ශල්‍යකර්මයට භාජනය කළ රෝගියා ඉතා හොඳ සෞඛ්‍ය තත්ත්වයෙන් පසුවන අතර ප්‍රසාරණ වී තිබූ මහා ධමනිය විවෘත කොට ඉතා සියුම් අයුරින් මෙම සැත්කම සිදුකොට තිබේ.
මෙකී සැත්කම පිළිබඳ අදහස් දක්වමින් විශේෂඥ ශල්‍ය වෛද්‍ය රංජුක උමේසිරි මහතා මෙසේ පැවසීය.
හෘදය වස්තුවෙන් පටන් අරන් අත්වලට හා මොළයට රුධිරය සපයන ප්‍රධාන ධමනියක් තමයි මහා ධමනිය.
මේ ධමනිය උරස්කුහරය හරහා උදර කුහරයට එනවා.
මේ ධමනියෙන් තමයි අක්මාවට, ප්ලීහාවට, අමාශයට හා ඉතුරු කොටස්වලට රුධිරය සපයන්නේ.
ඊට පසු නහර හරහා මහා බඩවැල් හා වකුගඩුවලට ලේ සපයනවා.
පසුව තමයි පාදවලට බෙදෙන්නේ.
ඔය අයුරින් තමයි මහා ධමනියෙන් රුධිරය සපයන්නේ.
මෙකී පුද්ගලයාගේ උරස්කුහරයේ හා උදර කුහරයේ මහා ධමනිය ප්‍රසාරණය වෙලා තිබුණා මහා ප්‍රාචීරය හරහා, අනිකුත් නහරත් සමඟ..
ප්‍රසාරණය වී තිබූ මහා ධමනිය විවෘත කරලා එක් බද්ධ කිරීමක් සිදු කළා.
පසුව සුසුම්නාවට යන නහර බද්ධ කළා. බඩවැල්වලට යන නහර බද්ධ කළා.
වීනෝ විනස් බයිපාස් සැත්කමක්ද කළා.
ඒ සඳහා හෘද හා උරස් කුහර පිළිබඳ විශේෂඥ ශල්‍ය වෛද්‍ය ශෙලූෂා හරිස්චන්ද්‍ර මහත්මිය හා ටිවඩි පර්මින් වෛද්‍යතුමාත් උදව් කළා.
යැයි පැවසීය.
මෙකී සැත්කම සඳහා නිර්වින්දන වෛද්‍ය ශිහාන් පියසිරි මහතා ඇතුළු රුධිර වාහිනී හා අවයව බද්ධ කිරීමේ ඒකකයේ වෛද්‍යවරුන් ඇතුළු කාර්ය මණ්ඩලයේ සහයෝගය හිමිව තිබේ.
මව හදිසි මරණ පරීක්ෂණයේදී කියයි
```

Figure B.1: Collected data using web-scraping

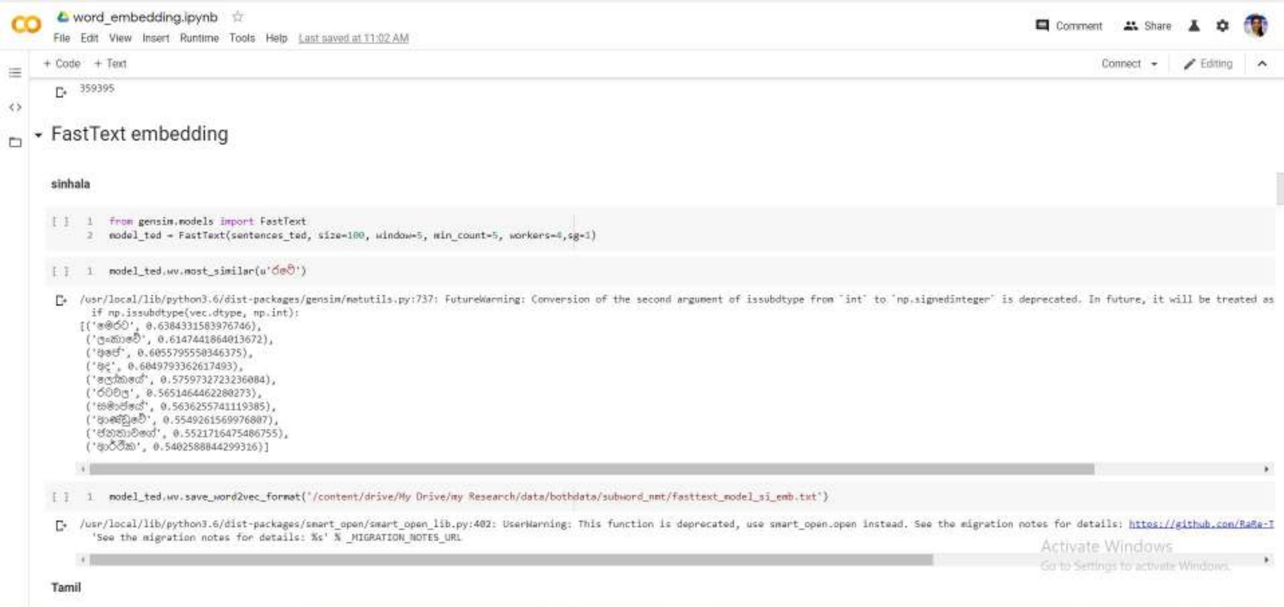
B.2 Google colab interface-01



```
[ ] from google.colab import drive
    drive.mount('/content/drive')

Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6ok8pdgf4n4g3pfee6491hc0brc4l.apps.googleusercontent.com&redirect_uri=urn%3aietf%3awg%3aoauth%3a2.0%3aoob&res

Enter your authorization code:
··········
Mounted at /content/drive

[ ]

[ ] !unzip '/content/drive/My Drive/my Research/data/sinhala.si.zip'

Archive:  /content/drive/My Drive/my Research/data/sinhala.si.zip
  inflating: sinhala.si

[ ] !unzip '/content/drive/My Drive/my Research/data/tamil.ta.zip'

Archive:  /content/drive/My Drive/my Research/data/tamil.ta.zip
  inflating: tamil.ta

[ ] import re

    path ="/content/drive/My Drive/my Research/data/bothdata/subword_nmt/sinhala.BPE.L1"
    #path ="/content/drive/My Drive/my Research/data/sinhala.si"

    with open(path, 'r') as f:
      content = f.read()
```

Figure B.2: Google colab using

B.3 Google colab interface-02



```
359395

[ ] from gensim.models import FastText
    model_ted = FastText(sentences_ted, size=100, window=5, min_count=5, workers=4,sg=1)

[ ] model_ted.wv.most_similar(u'රටේ')

/usr/local/lib/python3.6/dist-packages/gensim/matutils.py:737: FutureWarning: Conversion of the second argument of issubdtype from `int` to `np.signedinteger` is deprecated. In future, it will be treated as
  if np.issubdtype(vec.dtype, np.int):
[('සමරට', 0.6384331583976746),
 ('ලංකාවේ', 0.6147441864013672),
 ('ඉතේ', 0.6055795550346375),
 ('ඉද්', 0.6049793362617493),
 ('[illegible]', 0.5759732723236084),
 ('රටවල', 0.5651464462280273),
 ('සමාජයේ', 0.5636255741119385),
 ('ආණ්ඩුවේ', 0.5549261569976807),
 ('ජනතාවගේ', 0.5521716475486755),
 ('ආර්ථික', 0.5402588844299316)]

[ ] model_ted.wv.save_word2vec_format('/content/drive/My Drive/my Research/data/bothdata/subword_nmt/fasttext_model_si_emb.txt')

/usr/local/lib/python3.6/dist-packages/smart_open/smart_open_lib.py:402: UserWarning: This function is deprecated, use smart_open.open instead. See the migration notes for details: https://github.com/RaRe-T
  'See the migration notes for details: %s' % _MIGRATION_NOTES_URL
```

Figure B.3: Google colab using