NEURAL MACHINE TRANSLATION FOR SINHALA –TAMIL

Thesis · April 2018

CITATIONS
0

READS
1,257

2 authors:

Surangika Ranathunga
Massey University
122 PUBLICATIONS **888** CITATIONS
SEE PROFILE

Pasindu Tennage
École Polytechnique Fédérale de Lausanne
10 PUBLICATIONS **63** CITATIONS
SEE PROFILE

NEURAL MACHINE TRANSLATION FOR SINHALA - TAMIL

P.N.Tennage (130584U)
M.W.D.P.Sandaruwan (130534T)
J.K.M.M.Thilakarathne (130597L)
A.N.Herath (130199T)

Degree of Bachelor of Science

Department of Computer Science and Engineering

University of Moratuwa
Sri Lanka

December 2017

NEURAL MACHINE TRANSLATION FOR SINHALA – TAMIL

P.N.Tennage (130584U)
M.W.D.P.Sandaruwan (130534T)
J.K.M.M.Thilakarathne (130597L)
A.N.Herath (130199T)

Thesis submitted in partial fulfillment of the requirements for the degree Bachelor of Science in Computer Science and Engineering

Department of Computer Science and Engineering

University of Moratuwa
Sri Lanka

December 2017

DECLARATION

I declare that this is my own work and this thesis/dissertation2 does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

…… P.N.Tennage

…… M.W.D.P.Sandaruwan

…… A.N.Herath

…… J.K.M.M.Thilakarathne

The above candidates have carried out the research for the Bachelor of Science thesis under my supervision.

Supervised By

……

Dr. Surangika Ranathunga

Coordinated By

……

Dr. Charith Chitraranjan

ABSTRACT

Project Title: Neural Machine Translation for Sinhala – Tamil

Authors: P.N.Tennage M.W.D.P.Sandaruwan

A.N.Herath J.K.M.M.Thilakarathne

Supervisors : Dr. Surangika Ranathunga

Prof. Sanath Jayasena

Prof. Gihan Dias

Coordinator: Dr. Charith Chitraranjan

Neural Machine Translation (NMT) is the current state of the art machine translation technique that superseded Statistical Machine Translation (SMT), which was the most widely used machine translation technique for more than two decades. NMT exhibited an edge over SMT when translating between morphologically rich languages. Yet the applicability of NMT for under resourced languages is still debatable. Improving NMT performance for under resourced languages is an ongoing research area in machine translation domain. Sinhala and Tamil are under-resourced morphologically rich languages. Translation between the language pair utilizing neural machine translation is a novel research area, where no published literature is found. In this research, we focus on creating an NMT system for Sinhala and Tamil. We experiment on domain specific translation, focusing the domain of official government documents of Sri Lanka. We introduce novel techniques and enhanced existing techniques that improved NMT performance for under resourced languages in general, and for the selected language pair in particular. Experimental results show that the novel techniques that we introduce are capable of improving NMT performance for Sinhala and Tamil while achieving an overall 5.42 BLEU and 2.16 BLEU gains for Sinhala to Tamil and Tamil to Sinhala models respectively. Most of these techniques can be applied to other language pairs as well.

Key words: Neural machine translation, POS tagging, Morphological analysis, Data augmentation, Word phrases, Encoder Decoder

ACKNOWLEDGMENT

We would like to acknowledge with greatest gratitude the help and guidance we received to conduct this research, from these respected persons. We would like to show our deepest gratitude to project supervisor Dr Surangika Ranathunga for the thorough guidance and the consistent assistance we received throughout this research. Our gratitude goes to Professor Sanath Jayasena and Professor Gihan Dias, project supervisors for guiding us in the research through numerous consultations.

In addition, we like to thank Dr Uthayasanker Thayasivam, who introduced us to the methodology of work.

We like to extend our gratitude to members of the Si-Ta research group for the support they extended. Especially the postgraduate research students of Si-Ta research group, who have made valuable comments and suggestions on this project, which gave us an inspiration to improve our research.

The authors are grateful to members of the National Languages Processing Centre at University of Moratuwa for their significant contribution in developing the basic linguistic resources, and the Department of Official Languages of Sri Lanka for providing corpus data needed to carry out the research.

We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in completing this research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BiRNN	Bidirectional recurrent neural network
BLEU	bilingual evaluation understudy
BPE	Byte Pair Encoding
CFG	Context-free grammar
FST	Finite State Transducers
MT	Machine Translation
NMT	Neural Machine Translation
NPLM	Neural Probabilistic Language Model
OOV	out-of-vocabulary
POS	Parts of Speech
RBMT	Rule Based machine translation
RNN	Recurrent Neural Networks
SMT	Statistical machine translation
WER	Word error rate

LIST OF APPENDICES

APPENDIX A: Benchmark Results

APPENDIX B: Adding Word Phrases Results

APPENDIX C: Monolingual Training Data Results

APPENDIX D: Data Augmentation Results

APPENDIX E: Neural Machine Translation for Sinhala and Tamil Languages research paper

APPENDIX F: Handling Rare Word Problem using Synthetic Training Data for Sinhala and Tamil Neural Machine Translation research paper

1 INTRODUCTION

Machine Translation (MT), also known as automated translation, is the process of using software to translate text or speech from one language to another. Research in machine translation started more than 65 years ago.

Early translation techniques focused on word to word substitutions between the two languages. This technique alone failed to produce accurate translations thus giving way to the inauguration of a wide new research area. The first most noted research area is the Rule – Based machine translation (RBMT) that generates the output based on morphological, syntactic, and semantic analysis of both the source and the target language [1]. However, RBMT techniques proved to be less accurate due to the difficulty in incorporating rule interactions in big systems, ambiguity, and idiomatic expressions.

New trend in MT turned to the use of corpora (a large and structured set of text used to do statistical analysis and hypothesis testing, checking occurrences or validating linguistic rules within a specific language territory) to ensure translation of whole phrases of text to their closest counterparts in the target language.

Statistical machine translation (SMT), the most widely studied machine translation method dominated the field of MT for nearly two decades. It proposes to generate translations based on statistical models with parameters derived from the analysis of bilingual and monolingual corpora [2]. However, existing SMT systems are far perfect and their performance is greatly domain dependent. SMT techniques fail to provide accurate translations between language pairs with significant grammatical differences such as translation between Asian and European languages.

Thus, the emerging research in MT has turned towards neural machine translation (NMT). NMT is a simple new architecture that aims at building a single neural network that can be jointly tuned to maximize the translation performance. This neural network is trained using deep learning techniques. The models proposed recently for neural machine translation often belong to a family of encoder-decoders and consist of an encoder that encodes a source sentence into a set of annotation vectors from which the decoder generates translation [3].

Research and practice of NMT are only at their beginning stage. Despite being relatively new, it has already shown promising results, achieving state-of-the-art performance for various language pairs [4]. Its potential to overcome many of the aforementioned weaknesses of conventional phrase-based SMT systems has made popular translation providers such as Google Translator to turn to NMT [5]. NMT has proven to provide accurate translations for morphologically rich languages, than SMT.

Despite the fact that NMT provides intelligible outputs, NMT requires a very large parallel corpus to train the network. This requirement hinders the applicability of NMT for language pairs that lack the luxury of having a large parallel corpus. Extensive research has been carried out on improving NMT performance using a small parallel corpus while addressing rare word problem and out of vocabulary problem [4].

2 RESEARCH PROBLEM

2.1 Overview and Motivation

Currently the most popular translation tool that is accepted worldwide is the Google Translator [5]. Yet its translation performance for Sinhala and Tamil is considerably low.

Sinhala and Tamil languages are under resourced due to lack of sufficiently large parallel corpora. Research groups in the country have tried to address this issue by building language pair specific translation models between the two languages using methods such as statistical machine translation [6]. Due to the unavailability of large parallel corpora, these translation models have not yet reached the stage of proliferation demonstrated by other language translations (such as translation between European languages) [7] [8].

NMT has bypassed SMT for most language pairs. There is no published literature on the applicability of NMT for Sinhala and Tamil languages. Being low resourced is one of the major factors that hinders NMT performance. Improving NMT for under resourced languages is an ongoing research area with proven success.

2.2 Problem Statement

In this research, we focus on how to use Neural Machine Translation for low-resourced languages, focusing on Sinhala and Tamil.

3 RESEARCH OBJECTIVES AND AIMS

3.1 Overall Objective

Objective of this research is to identify effective approaches of improving domain specific NMT performance for under resourced languages, specifically focusing on the language pair Sinhala and Tamil. We focus on the domain of official government documents of Sri Lanka.

3.2 Specific Aims

1. Identify and implement the basic NMT system.
2. Design and implement effective approaches for improving NMT for under resourced languages.
3. Implement NMT improvements specific to Sinhala-Tamil translation.

4 LITERATURE REVIEW

4.1 Machine Translation

The first task when building a machine translation system is to collect pairs of source sentences and their corresponding translations. (X_n, Y_n) will be used to represent a pair of source and corresponding translation, respectively. D is the data set with N pairs. With the training data D in hand, it is possible to score a model by looking at how well the model works on the testing data. The score, which is called the log-likelihood of the model, is the average of the log-likelihood of the model on each pair of sentences. With the probabilistic interpretation of the machine translation model, the log-likelihood of the model on each pair is simply how high a log-probability (**$\log p(y^n|x^n, \Theta)$**) the model assigns to the pair. Θ is the set of parameters that defines the model.

The overall score of the model on the training data is **$L(\Theta, D) = \Sigma \log p(y^n|x^n, \Theta)$.**

If the log-likelihood L is low, the model is not giving enough probability mass to the correctly translated pairs, meaning that it is wasting its probability mass on some wrong translations. Thus, it is important to find a configuration of the model, or the values of the parameters that maximize this log-likelihood, or score. In machine learning, this is known as a maximum likelihood estimator.

SMT was the state-of-the-art MT system for more than 2 decades. SMT system is a combination of feature functions that are combined to maximize the translation performance. Introduction of NMT paved way to get the advantage of deep learning techniques for machine translation. NMT is an end to end system, which tunes its parameters to maximize translation performance.

4.2 Statistical Machine Translation

The core of SMT is a log-linear model, where the logarithm of the true **p(y|x)** is approximated with a linear combination of many features. A large part of the research comes down to finding a good set of feature functions. In this approach of statistical machine translation, often the only thing left to machine learning is to find a set of coefficients that balance among different features, or to filter or re-rank a set of potential translations decoded from the log-linear model [9]. Figure 1 shows the high level architecture of SMT.

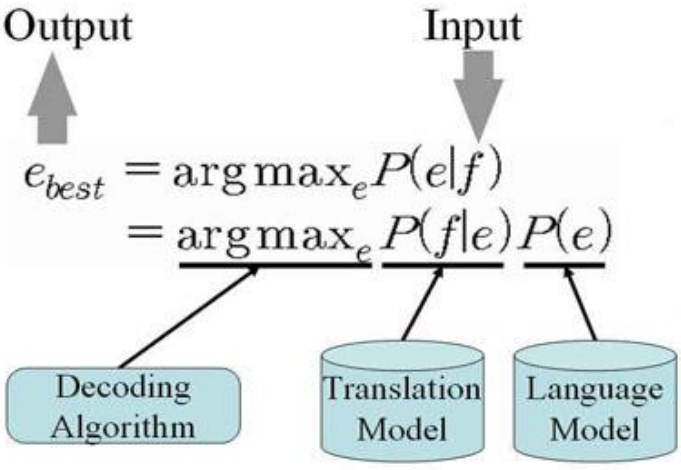

Figure 4.1 : Statistical Machine Translation: Statistical Machine Translation
Source: http://nlp.postech.ac.kr/research/previous_research/smt/images/math.jpg

f: source sentence

e: target sentence.

The standard approach of computing **pr(e|f)** is to use Bayes' theorem.

$$\mathrm{pr}(e|f) = \frac{\mathrm{pr}(f|e)\mathrm{pr}(e)}{\mathrm{pr}(f)}.$$

Because **f** is fixed, the maximization over **e** is thus equivalent to maximizing, **pr(e|f) = pr(f|e)pr(e)**

Inferring models for **pr(f|e)** and **pr(e)** and then using those models to search for **e** that maximizes **pr(f|e)pr(e)** is the approach that is used in SMT.

Machine translation using SMT can be broken up into three problems:

1. Language model which estimates **pr(e)**
2. Translation model which estimates **pr(f|e)**
3. Decoder which searches for **e** that maximizes the product **pr(f|e)pr(e)**

4.2.1 Translation Model

Two notions derived from alignments are particularly useful in building a translation model.

1. Fertility - Defined as the number of target side words that are generated by a given source side word.
2. Distortion - In many sentences, the source side word and its corresponding target side word or words appear in the same part of the sentence. Such words are translated roughly undistorted, while words which move a great deal have high distortion.

Translation model is built using some simple parameters related to fertility and distortion.

1. The fertility probability **pr(n|e)**: the probability that the source word **e** has fertility **n**.
2. The distortion probability **pr(t|s,l)**: the probability that a source word at position **s** corresponds to a target word at position **t** in a target side sentence of length **l**.
3. The translation probability **pr(f|e):** one for each source side word **f** and target side word **e**.

4.2.2 Language Model

Source sentence **e** can be broken up into words **e = e_1, e_2, e_3,.. e_n**. Then it's possible to write the probability for **e** as a product of conditional probabilities.

$$\mathrm{pr}(e) = \prod_{j=1}^{m} \mathrm{pr}(e_j|e_1, \ldots, e_{j-1})$$

The challenge in building a good language model **pr(e)** is that there are so many distinct conditional probabilities that need to be estimated. The most drastic assumption that is used in language modelling is to assume that the probability of seeing a word is independent of what came before it,

$$\mathrm{pr}(e_i|e_1, \ldots, e_{i-1}) = \mathrm{pr}(e_i)$$

Thus,

$$\mathrm{pr}(e) = \prod_{j=1}^{m} \mathrm{pr}(e_j).$$

Probabilities **pr(e_j)** can be estimated by taking a very large corpus of source side text, and counting words. The problem is that this model is not very realistic. A more realistic model is the bigram model, which assumes that the probability of a word occurring depends only on the word immediately before it.

$$\mathrm{pr}(e_j|e_1,\ldots,e_{j-1}) = \mathrm{pr}(e_j|e_{j-1})$$

Most widely used model is the trigram model, which assumes that the probability of a word occurring depends only on the two words immediately before it [10].

$$\mathrm{pr}(e_j|e_1,\ldots,e_{j-1}) = \mathrm{pr}(e_j|e_{j-2},e_{j-1}).$$

4.2.2.1 NPLM - Neural Probabilistic Language Model

A word sequence on which the model will be tested is likely to be different from all the word sequences seen during training (known as curse of dimensionality). Traditional but very successful approaches based on n-grams obtain generalization by concatenating very short overlapping sequences seen in the training set. Problems with traditional approach include,

1. It is not considering contexts farther than 1 or 2 words.
2. It is not considering the "similarity" between words.

NPLM is developed to deal with these issues. Figure 4.2 shows the high-level architecture of NPLM.

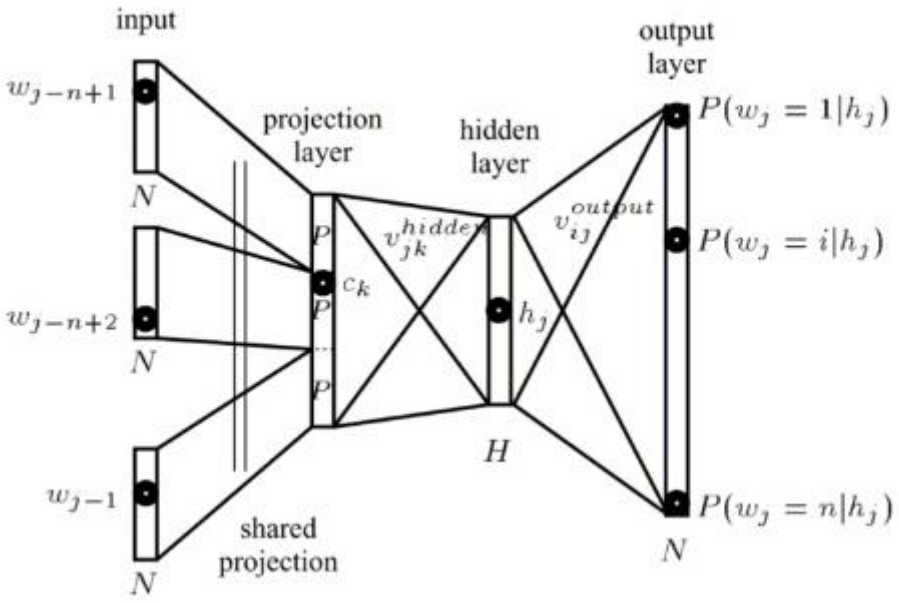

Figure 4.2 : NPLM Architecture

Source: https://image.slidesharecdn.com/latin-150313140222-conversion-gate01/95/representation-learning-of-vectors-of-words-and-phrases-12-638.jpg?cb=1426255492

NPLM learns a distributed representation for words which allows each training sentence to inform the model about an exponential number of semantically neighboring sentences. The model learns simultaneously,

1. A distributed representation for each word: distributed word feature vector (a real-valued vector in R^m)

2. The probability function for word sequences, expressed in terms of these representations.

Generalization is obtained because a sequence of words that have never been seen before gets high probability if it is made of words that are similar (in the sense of having a nearby representation) to words forming an already seen sentence.

4.2.3 Decoder

Both language model and translation model are defined as mathematical formulae that given possible translation, assigns a probabilistic score to it. Task of Decoder is to find the best scoring translation according to these formulae.

Heuristic search methods are used in the decoding stage.

There exist 2 types of errors in decoding process.

1. Search error: Failure to find best translation. This is a consequence of heuristic function which is unable to explore the entire search space.
2. Model error: Highest possible translation might not be the best. (Not a problem of decoding).

4.2.4 Statistical Machine Translation Systems

4.2.4.1 MOSES

Moses is a statistical machine translation system that allows to automatically train translation models for any language pair using a collection of translated texts (parallel corpus) [11]. Once a model is trained, an efficient search algorithm quickly finds the highest probability translation among the exponential number of choices.

4.2.4.1.1 Features

1. Moses offers two types of translation models: phrase-based and tree-based
2. Moses features factored translation models, which enable the integration of linguistic and other information at the word level.
3. Moses allows the decoding of confusion networks and word lattices, enabling easy integration with ambiguous upstream tools, such as automatic speech recognizers or morphological analyzers
4. The Experiment Management System makes using Moses much easier.

The translation model should be trained using a parallelly aligned corpus and the language model should be trained on a corpus that is in domain with the translation task.

MOSES comes with pre-configured language modeling toolkits.

1. KenLM
   KenLM is a language modeling toolkit that is simultaneously fast and low memory. KenLM is distributed with Moses and compiled by default.
2. NPLN
   NPLM is a neural network language modeling toolkit. It uses a neural network to compute tri-gram probabilities.

3. Bilingual-LM
   A neural network language model that uses a target-side history as well as source-side context, is implemented in Moses as Bilingual LM. It uses NPLM as the back-end.

4.3 Neural Machine Translation

Neural machine translation does not rely on pre-designed feature functions. The goal of NMT is to design a fully trainable model of which every component is tuned based on training corpora to maximize its translation performance.

Figure 4.3 shows the high-level architecture diagram of NMT.

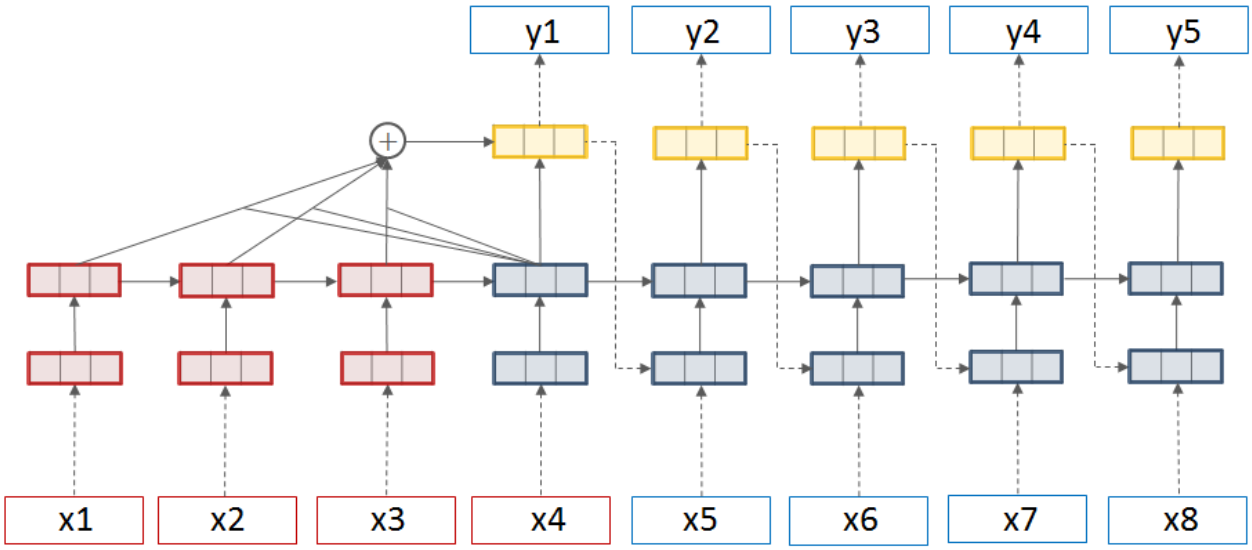

Figure 4.3 : Neural Machine Translation
source: http://opennmt.net/simple-attn.png

Given a source sequence X = {x_1, x_2, x_3, x_4 … x_T} of word indices, the NMT model computes the conditional probability of Y = {y_1, y_2, y_3, y_4.... $y_{T'}$} - (T and T' in X and Y are not fixed, i.e. the sentence length can vary).

4.3.1 Recurrent Neural Networks

To deal with variable-length input and output, recurrent neural networks (RNN) are used [12]. Widely used feed-forward neural networks, such as convolutional neural networks, do not maintain internal state other than the network's own parameters. Whenever a single sample is fed into a feed-forward neural network, the network's internal state is computed from scratch and is not influenced by the state computed from the previous sample. On the other hand, a RNN maintains its internal state while reading a sequence of inputs, which in this case will be a sequence of words, there by being able to process an input of any length [13].

Figure 4.4 shows inputs and outputs of a RNN network with 3 units.

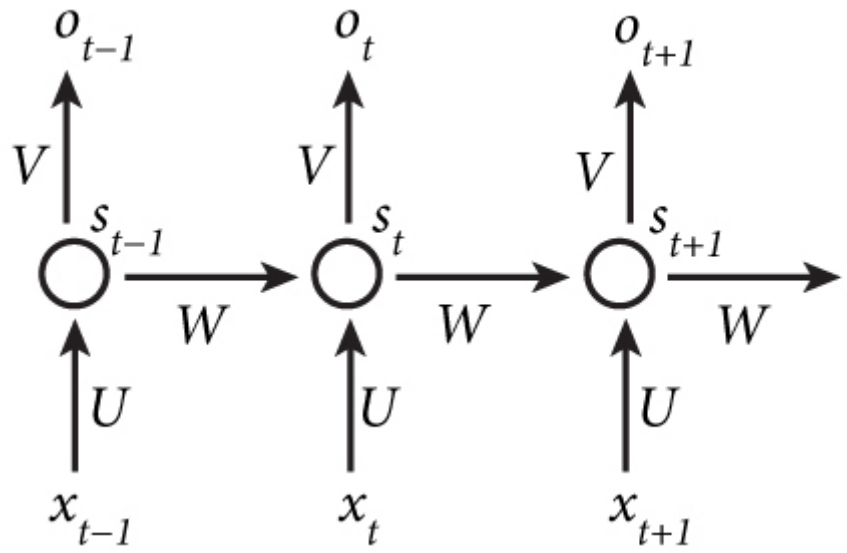

Figure 4.4 : Recurrent Neural Network
source: http://d3kbpzbmcynnmx.cloudfront.net/wp-content/uploads/2015/09/rnn.jpg

The main idea behind RNNs is to compress a sequence of input symbols into a fixed dimensional vector by using recursion. If at step t there is a vector h_{t-1} which is the history of all the preceding symbols, then the RNN will compute the new vector, or its internal state, h_t which compresses all the preceding symbols $\{x_1, x_2, x_3, x_4, x_{T-1}\}$ as well as the new symbol x_t using, $\mathbf{h_t = shy(x_t, h_{t-1})}$. ($h_0$ is all-zero vector).

The recurrent activation function **shy** is often implemented as a simple affine transformation followed by an element-wise nonlinear function, $h_t = \mathbf{tanh(Wx_t + Uh_{t-1} + b).}$

4.3.2 Encoder Decoder Architecture

Figure 4.5 shows the 2 main stages of encoder decoder approach.

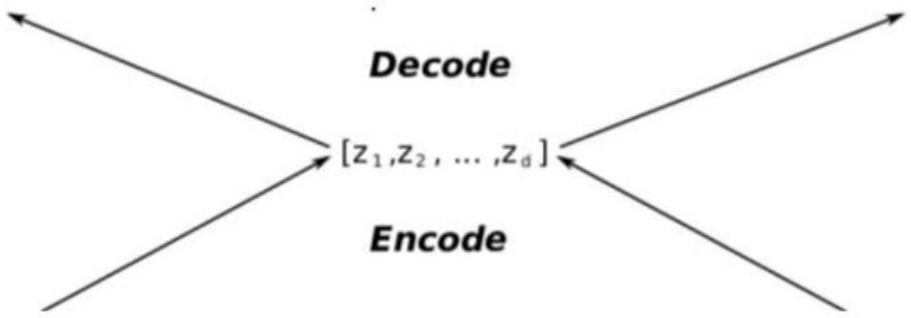

Figure 4.5 : Encoder-Decoder approach
source: https://devblogs.nvidia.com/wp-content/uploads/2015/06/figure1_encoder-decoder1-300x126.png

4.3.2.1 Encoder

In short, recurrent activation function is applied recursively over the input sequence, or sentence, until the end when the final internal state of the RNN, h_T is the summary of the whole input sentence [14]. First, each word in the source sentence is represented as a one-hot vector, or 1-of-K coded vector.

Every word is equidistant from every other word, meaning that it does not preserve any relationships among them.

The encoder linearly projects the 1-of-K coded vector w_i with a matrix E which has as many columns as there are words in the source vocabulary and as many rows required. This projection $\mathbf{S_i = EW_i}$ results in a continuous vector for each source word, and each element of the vector is later updated to maximize the translation performance.

In mathematical notation the process of summarization can be represented as $h_i = shy0(s_i, h_{i-1})$, where h_0 is an all-zero vector. After the last word's continuous vector s_T is read, the RNN's internal state h_T represents the summary of the whole source sentence.

Figure 4.6 shows a detailed layered diagram of encoder decoder architecture.

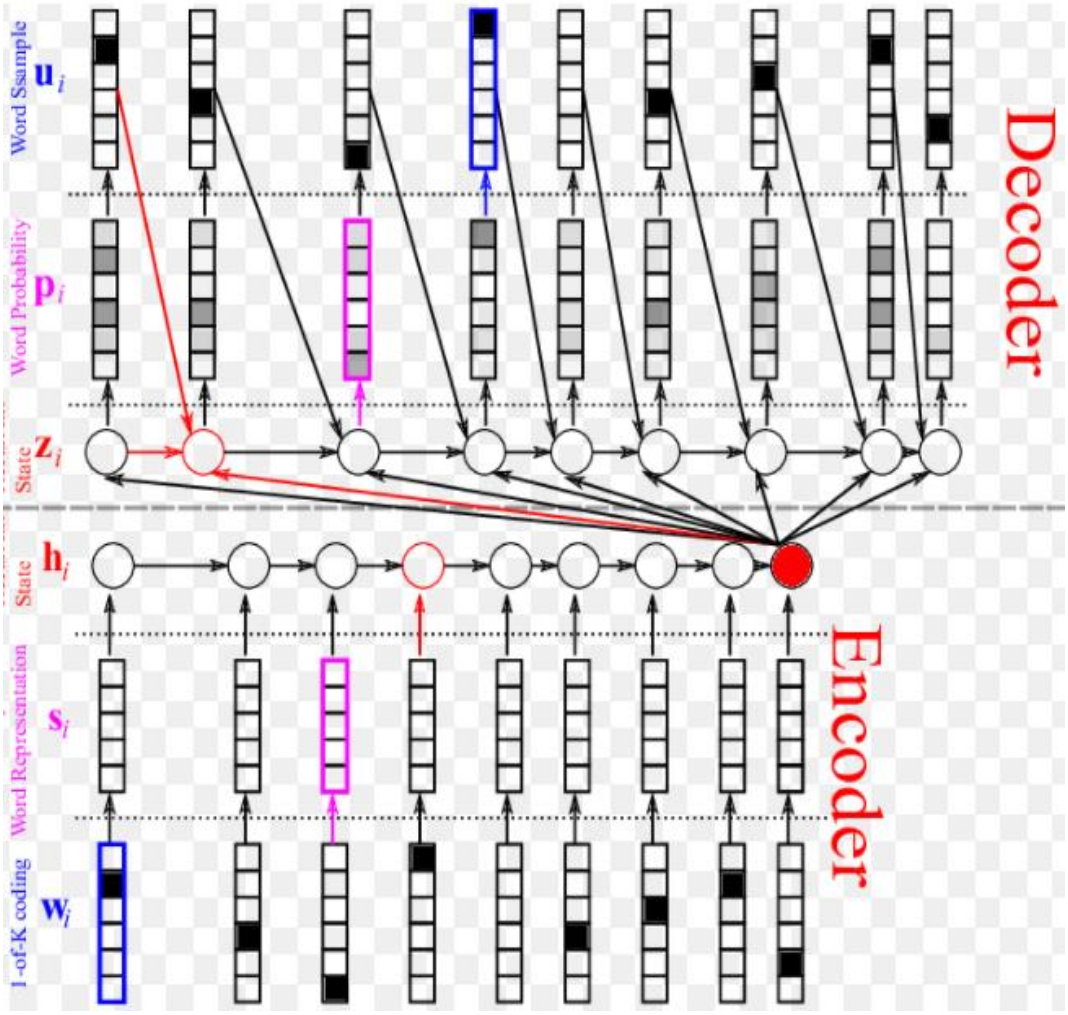

Figure 4.6 : Encoder Decoder Architecture
source: https://devblogs.nvidia.com/wp-content/uploads/2015/06/Figure2_NMT_system.png

4.3.2.2 Decoder

Decoder is essentially the encoder flipped upside down. It computes the RNN's internal state z_i based on the summary vector h_T of the source sentence, the previous word u_{i-1} and the previous internal state z_{i-1} using $\mathbf{z_i = shy0'(h_T, u_{i-1}, z_{i-1})}$ [14].

With the decoder's internal hidden state $\mathbf{z_i}$ ready, it's possible to score each target word based on how likely it is to follow all the preceding translated words given the source sentence. This is done by assigning a probability $\mathbf{p_i}$ to each word. Each word k is scored, given a hidden state $\mathbf{z_i}$ such that $\mathbf{e(k) = w_k^T z_i + b_k}$. ($\mathbf{w_k}$ and $\mathbf{b_k}$ are the (target) word vector and a bias, respectively).

The dot product is larger when the target word vector $\mathbf{w_k}$ and the decoder's internal state $\mathbf{z_i}$ are similar to each other, and smaller otherwise. A dot product gives the length of the projection of one vector onto another; if they are similar vectors (nearly parallel) the projection is longer than if they were different (nearly perpendicular). Hence this mechanism scores a word high if it aligns well with the decoder's internal state.

Once the score of every word is computed, scores are turned into proper probabilities using softmax normalization.

$$p(w_i = k | w_1, w_2, \ldots, w_{i-1}, h_T) = \frac{\exp(e(k))}{\sum_j \exp(e(j))}.$$

4.3.3 Maximum Likelihood Estimation

First, a parallel corpus **D** must be prepared. Each sample in the corpus is a pair ($\mathbf{X_n}$, $\mathbf{Y_n}$). Each sentence is a sequence of integer indices corresponding to words, which is equivalent to a sequence of one-hot vectors. Given any pair from the corpus, the NMT model can compute the conditional log-probability of $\mathbf{Y_n}$ given $\mathbf{X_n}$, using **log P(Yn | Xn , theta)**.

Thus, the log-likelihood of the whole training corpus is given by,

$$\mathcal{L}(D, \theta) = \frac{1}{N} \sum_{n=1}^{N} \log P(Y^n | X^n, \theta).$$

The Log-likelihood function can be maximized using stochastic gradient descent.

4.3.4 Attention Mechanism

In the encoder-decoder architecture, the encoder compresses the input sequence as a fixed-size vector from which the decoder needs to generate a full translation. The fixed-size vector, which is called a context vector, must contain every single detail of the source sentence. Intuitively, this means that the true function approximated by the encoder must be extremely nonlinear and complicated. Furthermore, the dimensionality of the context vector must be large enough that a sentence of any length can be compressed.

Bidirectional recurrent neural network (BiRNN) consists of a forward recurrent neural network (RNN) and a separate backward RNN. The forward and backward RNNs read the source sentence in forward and backward directions, respectively. Figure 4.7 shows the inputs and outputs of 3 bidirectional recurrent neural networks.

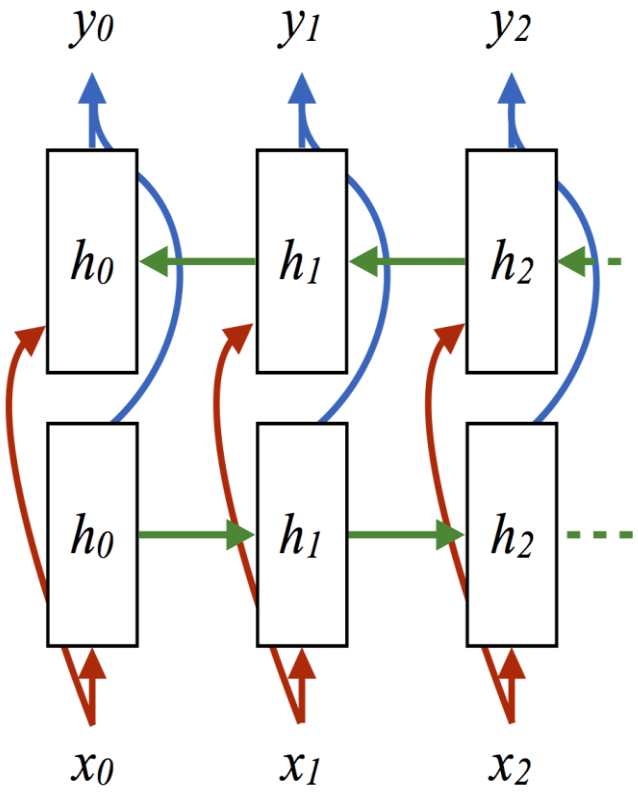

Figure 4.7 : Bidirectional recurrent neural networks
source: http://blog.jacobandreas.net/figures/monference_bdrnn.png

Hidden state from the forward RNN: $\mathbf{h_{jf}}$

Hidden state from the backward RNN: $\mathbf{h_{jb}}$

A RNN summarizes a sequence by reading one symbol at a time. $\mathbf{h_{jf}}$ of the forward RNN summarizes the source sentence up to the j-th word beginning from the first word, and h_{jb} of the backward RNN up to the j-th word beginning from the last word. In other words, $\mathbf{h_{jf}}$ and $\mathbf{h_{jb}}$ together summarize the whole input sentence. This summary at the position of each word, however, is not the perfect summary of the whole input sentence. Due to its sequential nature, a recurrent neural network tends to remember recent symbols better. The further away an input symbol is from **j,** the less likely the RNN's hidden state, either $\mathbf{h_{jf}}$ or $\mathbf{h_{jb}}$, remembers it perfectly. The annotation vector, which we use to refer to the concatenation of $\mathbf{h_{jf}}$ and $\mathbf{h_{jb}}$ represents the current word $\mathbf{w_j}$ best. Figure 4.8 shows the architecture diagram of the attention mechanism.

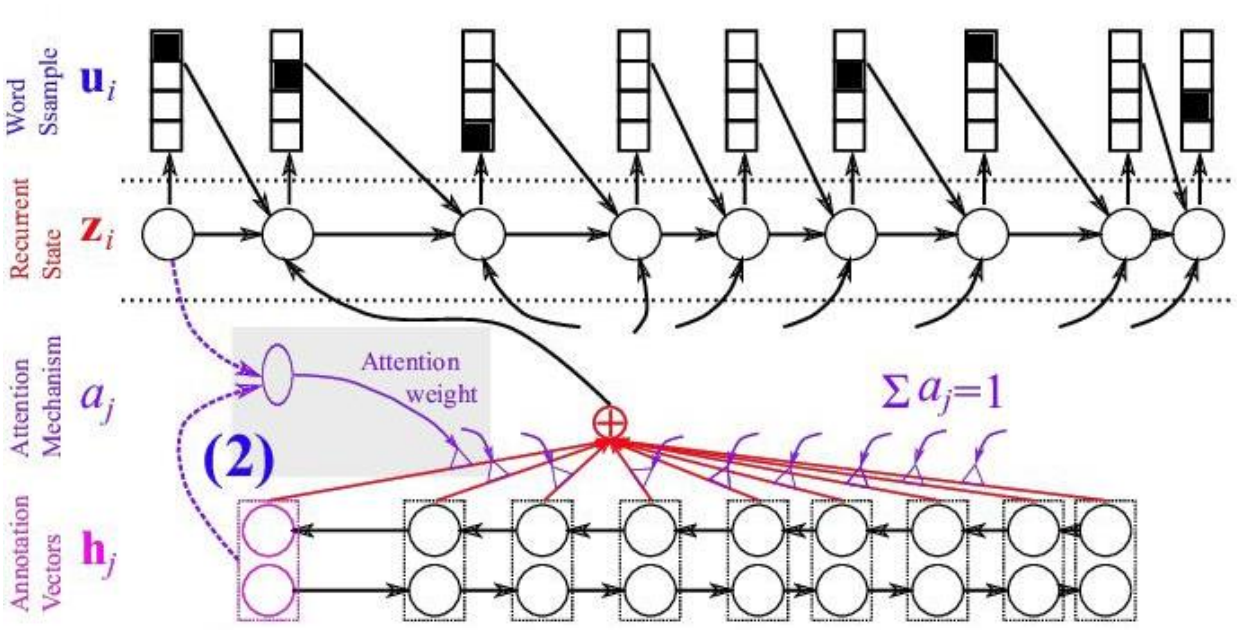

Figure 4.8 : Attention Mechanism

source: https://devblogs.nvidia.com/wp-content/uploads/2015/07/Figure4_attention_2-624x352.png

With this variable-length representation of a source sentence, the decoder now needs to be able to selectively focus on one or more of the context-dependent word representations, or the annotation vectors, for each target word.

When translating the given source sentence, where first **i-1** target words ($\mathbf{y_1, y_2 \ldots y_{i-1}}$) are already translated, there is a need to decide which target word to write as the $\mathbf{i^{th}}$ target word. In this case, it should be decided as to which source word should get translated this time.

A typical translator looks at each source word $\mathbf{x_j}$ (or its context-dependent representation $\mathbf{h_j}$), and considers it together with the already translated words $\mathbf{(y_1, y_2, .., y_{t-1})}$ and decides whether the source word $\mathbf{x_j}$ has been translated (equivalently, how relevant the source word $\mathbf{x_j}$ is for the next target word). It repeats this process for every word in the source sentence.

The small neural network, which is called the attention mechanism, takes as input the previous decoder's hidden state $\mathbf{z_i}$ (what has been translated) and one of the source context-dependent word representations $\mathbf{h_j}$. The attention mechanism is implemented as a neural network with a single hidden layer and a single scalar output $\mathbf{e_j}$. This is applied to every source word. Once the relevance score of every source word is computed, softmax normalization is used.

$$\alpha_j = \frac{\exp(e_j)}{\sum_{j'} \exp(e_{j'})}.$$

From this probabilistic perspective, attention weight can be defined as the probability of the decoder selecting the $\mathbf{j^{th}}$ context-dependent source word representation out of all **T** source words. Then, the expected context-dependent word representation under this distribution is computed using,

$$c_i = \sum_{j=1}^{T} \alpha_j h_j = \mathbb{E}_{\alpha'_j}[h_j].$$

This expected vector $\mathbf{c_i}$ summarizes the information about the whole source sentence, however, with different emphasis on different locations/words of the source sentence.

4.3.5 Sizes of datasets used in NMT

It has been identified that online available large corpora such as WMT have been used in many research, which has given a benchmark for comparison. There is a direct effect on parallel corpus size on the BLEU score. Table 4.1 demonstrates the important research papers with the sizes of the parallel corpuses used, the language pair and the obtained BLEU scores.

Table 4.1: Sizes of datasets used in NMT

	Paper	Language Pair	Original Corpus Size	Trained Corpus	BLEU score
1	Japanese-to-English Machine Translation Using Recurrent Neural Networks	Japanese - English	Tanaka corpus (publicly available) 150,000 sentence-pairs collection	1000 sentence-pairs (Very poor results)	0.73
2	Improved Neural Machine Translation with SMT Features (OOV problem)	Chinese - English	Billion Chinese words & 2.3 billion English words	Same	36-38
3	Translation of Patent Sentences with a Large Vocabulary of Technical Terms Using Neural Machine Translation	Japanese - Chinese	2.8 M pairs	same	55 range
4	Japanese- English machine translation of recipe text	Japanese - English	Cookpad Japanese English corpus	same	28.09
5	Domain adaptation and attention based unknown word replacement in chinese to japanese NMT	Chinese - Japanese	672, 315 pairs	same	46.7
6	Neural Machine Translation for rare words with sub word units	English - German, English to Russian	4.2 million, 2.6 million	same	25.3

7	On the Properties of Neural Machine Translation: Encoder–Decoder Approaches	English - French	348M words	Same	-
8	Effective Approaches to Attention-based Neural Machine Translation	English - German	4.5M sentences pairs (116M English words, 110M German Words).	same	25.9
9	Character based neural machine translation.	English-Portuguese English-French	600k sentence pairs 20k sentence pairs	same	19.29, 15.45
10	Sequence to Sequence Learning with Neural Networks	English - French	WMT'14 English to French dataset.	12M sentences consisting of 348M French words and 304M English words	34.8
11	Minimum Risk Training for Neural Machine Translation	Chinese-English	2.56M pairs of sentences with 67.5M Chinese words and 74.8M English words		
12	Improving Neural Machine Translation Models with Monolingual Data	English-German	WMT 2015 parallel 4,200,000 sentences	Same	
13	Linguistic Input Features Improve Neural Machine Translation	English-German	4.2 million sentence pairs	Same	

4.3.6 Neural Machine Translation System Implementations

Table 4.2 shows the details of existing NMT implementations.

Table 4.2: NMT implementations

Name	Maximum BLEU Score	Size of the corpus	
		Number of parallel sentences	Number of words
Stanford neural machine translation system	31.4	4.5M sentence pairs from WMT dataset + 200K sentence Pairs from IWSLT 2015 data set	Top 50K frequent words for each language
Stanford neural machine translation system - For low resource language pairs	26.4	133K sentence pairs	17K and 7.7K for English and Vietnamese
Edinburgh neural machine translation system	En -> De 34.2 / De -> En 38.6	English - German 4.2M parallel sentences	
Montreal neural machine translation system	En -> Cs 18.3 / Cs -> En 23.3	English - Czech 0.6M parallel sentences	

4.3.6.1 OpenNMT

OpenNMT is an open source (MIT licensed) neural machine translation framework utilizing the Torch/ PyTorch mathematical toolkit [15]. This system prioritizes goal of supporting NMT researches while maintaining competitive performance. OpenNMT is a complete library for training and deploying neural machine translation models.

The system is a successor to seq2seq model developed at Harvard, and has been completely rewritten for ease of use, readability, and generalizability. It includes vanilla NMT models along with support for attention, gating, stacking, input feeding, regularization, beam search and all other options necessary for state-of-the-art performances. Just as the SMT community benefited greatly from toolkits like Moses [11], OpenNMT toolkit is to support the NMT community in academia and industry. There are a lot of additional options on top of the baseline model. OpenNMT has an active open community welcoming both academic and industrial requests and contributions.

Table 4.3 depicts results obtained for English to German translation by a baseline model generated by OpenNMT system.

Table 4.3: OpenNMT English to German translation results

Corpus	Training Parameters	Server Details	Score
WMT15 - Translation Task (German-English) 4.5 M parallel sentence pairs.	Default options: 2 layers, RNN 500, WE 500, input feed 13 epochs	Intel(R) Core(TM) i7-6800K CPU @ 3.40GHz, 256Gb Mem, trained on 1 GPU TITAN X (Pascal)	NIST=5.5376 BLEU=17.02

4.3.6.2 Nematus

Nematus was developed by research teams from University of Edinburgh, Middle East Technical University New York University, Heidelberg University and University of Zurich. Nematus has been used to build top-performing submissions to shared translation tasks at WMT and IWSLT, and has been used to train systems for production environments. It implements and an attentional encoder–decoder architecture.

While the system has been tested for German-English model, the system has proven to be unsupportable for language pair of Sinhala and Tamil since during training, it requires the data set to be converted to a file format named as ConLL, which is a format separating each word into different phrases. This is a functionality supported by the Stanford Parser for languages that are close to English. Sinhala and Tamil which are built on a different morphology cannot be parsed by the above tool. Currently there are no tools supporting the above phrasing of Sinhala Tamil languages.

4.4 Translation Evaluation

4.4.1 BLEU Score

BLEU (bilingual evaluation understudy) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another [16]. Quality is the correspondence between a machine's output and that of a human: "the closer a machine translation is to a professional human translation, the better it is" – this is the central idea behind BLEU. BLEU was one of the first metrics to claim a high correlation with human judgements of quality, and remains one of the most popular automated and inexpensive metrics.

Scores are calculated for individual translated segments—generally sentences—by comparing them with a set of good quality reference translations. Those scores are then averaged over the whole corpus to reach an estimate of the translation's overall quality. Intelligibility or grammatical correctness are not considered.

BLEU is designed to approximate human judgement at a corpus level, and performs badly if used to evaluate the quality of individual sentences. BLEU's output is always a number between 0 and 1. This value indicates how similar the candidate text is to the reference text, with values closer to 1 representing more similar texts. Few human translations will attain a score of 1, since this would indicate that the candidate is identical to one of the reference translations. For this reason, it is not necessary to attain a score of 1 as there are more opportunities to match and adding additional reference translations will increase the BLEU score.

BLEU uses a modified form of precision to compare a candidate translation against multiple reference translations. The metric modifies simple precision since machine translation systems have been known to generate more words than are in a reference text.

BLEU has frequently been reported as correlating well with human judgement, and remains a benchmark for the assessment of any new evaluation metric. There are however many criticisms that have been voiced. It has been noted that although in principle it is capable of evaluating translations of any language, BLEU cannot in its present form deal with languages lacking word boundaries.

It has been argued that although BLEU possess significant advantages, there is no guarantee that an increase in BLEU score is an indicator of improved translation quality. There is an inherent, systemic problem with any metric based on comparing with one or a few reference translations: in real life, sentences can be translated in many ways, sometimes with no overlap. Therefore, the approach of comparing by how much any given translation result by a computer differs from just a few human translations is considered flawed.

4.4.2 NIST Score

NIST is a method for evaluating the quality of text which has been translated using machine translation. Its name comes from the US National Institute of Standards and Technology [17].

It is based on the BLEU metric, but with some alterations. Where BLEU simply calculates n-gram precision adding equal weight to each one, NIST also calculates how informative a n-gram is. When a correct n-gram is found, the rarer that n-gram is, the more weight it will be given.

NIST also differs from BLEU in its calculation of the brevity penalty insofar as small variations in translation length do not impact the overall score as much.

4.4.3 Human Evaluation

There exist two large scale evaluation studies that have had significant impact on the field—the ALPAC 1966 study and the ARPA study [18].

4.4.4 Word Error

The Word error rate (WER) is a metric based on the Levenshtein distance, where the Levenshtein distance works at the character level, WER works at the word level [18]. It was originally used for measuring the performance of speech recognition systems, but is also used in the evaluation of machine translation. The metric is based on the calculation of the number of words that differ between a piece of machine translated text and a reference translation.

A related metric is the Position-independent word error rate (PER), which allows for re-ordering of words and sequences of words between a translated text and a references translation.

4.5 Machine Translation for Sinhala and Tamil

In this section, we first discuss the basic linguistic resources that have been explored in Sinhala Tamil machine translation. Then we discuss the different approaches of Sinhala Tamil machine translation that have been explored in literature.

4.5.1 Basic Linguistic Tools

Morphological analysis plays a key role in effective functioning of a parser of any machine translation system. Further, morphological analyzers are useful as supportive software tools to coin terms for a given language. Hettige et al. [20] have reported on the first morphological analysis system for Sinhala language. The paper has described how the morphological analyzer can detect grammatical information of a given Sinhala word, and generation of all possible forms of the given word. It has also presented how a language specialist can use the proposed system to device Sinhala terms that are agreeable with Sinhala grammar. The proposed model has been able to retrieve morphological features of words with a good accuracy.

Development of the computational model of grammar for a highly inflected language is a complex task and it is also essential to develop rule-based machine translation systems. Hettige et al. [21] have presented a computational model of grammar for Sinhala language by considering the morphology and the syntax of the Sinhala language. Finite State Transducers (FST) and Context-free grammar (CFG) have been used to describe the computational grammar for Sinhala. The grammar has been tested through the English to Sinhala Machine Translation System. The translation system has been able to successfully translate English sentences with simple or complex subjects and objects with most commonly used patterns of the tenses including active and passive voice forms.

4.5.2 Translation Systems

Jeyakaran et al [22] have investigated the applicability of Kernel Ridge Regression technique to Sinhala-Tamil translation with a small data set. For this, the translation problem has been viewed as a string-to-string mapping among source and target language features. The Kernel Ridge Regression model, two novel solutions for pre-image problem and the decoder have been explored in this research. Compared to SMT approaches, this method has produced low accurate results due to string to string mapping. Yet, this research has paved way for new Sinhala and Tamil machine translation research.

Silva et al. [23] have presented an Example Based Machine Translation System which can be used for English to Sinhala translations mainly to be used in the government domain. The System has used a bilingual corpus of English - Sinhala aligned at sentence level, as the knowledge base of the System. Given a source phrase, the System retrieves the English sentences and the corresponding Sinhala sentences in which the input phrase is found (Intra Language Matching). Then the System performs a scoring algorithm on the retrieved Sinhala sentences to find the most occurring Sinhala phrase in the set, which is most likely to be the best candidate translation for the phrase (Inter-Language Matching). Compared to English to Sinhala SMT, that has been proposed later, use of example based machine translation has produced results with less accuracy and fluency.

Hettige et al. [24] have presented a theoretical-based approach to English-Sinhala machine translation through the concept of Varanagema (conjugation) in Sinhala Language. The theory of Varanagema in Sinhala language handles major language primitives including nouns, verbs and prepositions. The concept of Varanagema also drastically reduces the number of word forms to be stored in the dictionaries of the machine translation system. The design, implementation and results of test versions of English-Sinhala machine translation system have been presented in this research. Even though this research has stated acceptable translation results, theoretical based approaches have been proved inefficient in general machine translation tasks.

Out-of-vocabulary, handling proper nouns and technical terms are some major issues which are common to all machine translation systems. Hettige et al. [25] have proposed a transliteration approach to machine translation from English to Sinhala. They have used finite state automaton to develop transducers for English to Sinhala transliteration. This approach can transliterate the text in original English and Sinhala words that are written using English letters. The transliteration system has been developed using SWI-PROLOG and prolog server page (PSP). English WorldNet and Sinhala Chatbot have been used to test the transducers and reasonable results have been achieved. One major drawback of this method is its inability to be used as an end to end translation system. This method can only be used as a supplement to an already existing end to end translation system.

A basic SMT system has been modelled and implemented by Sripirakas et al. [26], with the preparation of parallel corpora from parliament order papers. Sripirakas et al. [26] demonstrate on the preliminary system, devoid of various parameter refinements and actual design and evaluation strategies. Language Model, Translation Model and Decoder Configurations have been done consistent with recent literature. To facilitate the improvement of output quality, MERT technique has been integrated to tune the decoder. To stay away from sole dependence on BLEU, two other automatic metrics namely TER and NIST have been utilized for the evaluation in different aspects. In addition, directions to future research have also been recognized and specified for the refinements of this system. This method is the first known SMT system that uses Sinhala language. With respect to the corpus size they have used, the results seem to have an edge over previous rule based approaches of Sinhala Tamil machine translation.

A Research done by Weerasinghe et al. [27] has attempted to cross the language family divide to compare the performance of machine translation techniques on Asian languages. Their work reports on Statistical Machine Translation experiments carried out between language pairs of the three major languages of Sri Lanka: Sinhala, Tamil and English. Results indicate that the current models perform significantly better for the Sinhala-Tamil pair than the English-Sinhala pair. This in turn appears to confirm the assertion that these techniques work better for languages that are not too distantly related to each other.

Recently a research has been carried out by Farhath et el. [28] on the development of a Sinhala-to-Tamil SMT system for official government documents. This system has been developed with emphasis given to domain adaptation. Performance of the system has been evaluated with the static integration of three types of lists, namely, a list of government organizations and official designations, a glossary related to government administrations and operations, and a general bilingual dictionary to the translation model of the SMT system. The proposed method has shown the applicability of lists for domain adaptation, even though the observed BLEU score gains are not in acceptable levels.

4.6 Improving Neural Machine Translation

4.6.1 Using large vocabulary

Despite NMT's recent success, neural machine translation has its limitation in handling a larger vocabulary, as training complexity as well as decoding complexity increase proportionally to the number of target words. Jean et al. [29] have proposed a method that allows to use a very large target vocabulary without increasing training complexity, based on importance sampling. They have shown that decoding can be efficiently done even with the model having a very large target vocabulary by

selecting only a small subset of the whole target vocabulary. The models trained by the proposed approach have been empirically found to outperform the baseline models with a small vocabulary as well as the LSTM-based neural machine translation models. Furthermore, using the ensemble of a few models with very large target vocabularies, they have achieved the state-of-the-art translation performance on the English->German translation and almost as high performance as state-of-the-art English->French translation system. Main outcome of this research is the method of using large vocabularies in training NMT models. For languages which have large vocabulary due to morphological richness, this method provides a good approach to train the NMT model with sufficient fluency.

Haitao et al. [30] have alleviated this issue by introducing a sentence-level or batch-level vocabulary, which is only a very small subset of the full output vocabulary. For each sentence or batch, the research has only predicted the target words in its sentence-level or batch-level vocabulary. Thus, they have reduced both the computing time and the memory usage. Proposed method has simply considered the translation options of each word or phrase in the source sentence, and picked a very small target vocabulary for each sentence based on a word-to-word translation model or a bilingual phrase library learned from a traditional machine translation model. This method solves the vocabulary size barrier that is present in morphologically rich languages' machine translation.

4.6.2 Handling out of vocabulary words

A significant weakness in conventional NMT systems is their inability to correctly translate rare words (words that rarely appeared in the parallel corpus) and out of vocabulary words (words that did not appear in the parallel corpus): end-to-end NMTs tend to have relatively small vocabularies with a single **unk** symbol that represents every possible out-of-vocabulary (OOV) word. Luong et al. [31] have implemented an effective technique to address this problem. An NMT system has been trained on data that is augmented by the output of a word alignment algorithm, allowing the NMT system to emit, for each OOV word in the target sentence, the position of its corresponding word in the source sentence. This information is later utilized in a post-processing step that translates every OOV word using a dictionary. Using this strategy the translation model has been able to provide acceptable level of BLEU score gain.

4.6.3 Handling rare word problem

Sennrich et al. [32] have introduced a simpler and more effective approach for rare word problem, making the NMT model capable of open-vocabulary translation by encoding rare and unknown words as sequences of sub-word units. This has been based on the intuition that various word classes are translatable via smaller units than words, for instance names (via character copying or transliteration), compounds (via compositional translation), and cognates and loanwords (via phonological and morphological transformations). They have discussed the suitability of different word segmentation techniques, including simple character n-gram models and a segmentation based on the byte pair encoding compression algorithm, and empirically shown that sub-word models improve over a back-off dictionary baseline. This method has produced promising results for machine translation of morphologically rich languages.

To control the computational complexity, NMT must employ a small vocabulary, and massive rare words outside the vocabulary are all replaced with a single **unk** symbol. Besides the inability to translate rare words, this kind of simple approach leads to much increased ambiguity of the sentences

since meaningless **unks** break the structure of sentences, and thus hurts the translation and reordering of the in-vocabulary words. To tackle this problem, Xiaoqing et al. [33] have proposed a novel substitution-translation-restoration method. In the substitution step, the rare words in a testing sentence has been replaced with similar in-vocabulary words based on a similarity model learnt from monolingual data. In translation and restoration steps, the sentence has been translated with a model trained on new bilingual data with the rare words replaced, and finally the translations of the replaced words have been substituted by that of original ones. The proposed method has produced good results for European languages. Due to massive fertility, applicability of this method for morphologically rich languages is questionable.

4.6.4 Character based NMT

Ling et al [34] have introduced a neural machine translation model that views the input and output sentences as sequences of characters rather than words. Since word-level information provides a crucial source of bias, their input model compose of representations of character sequences into representations of words (as determined by whitespace boundaries), and then these have been translated using a joint attention/translation model. In the target language, the translation has been modeled as a sequence of word vectors, but each word has been generated one character at a time, conditional on the previous character generations in each word. As the representation and generation of words have been performed at the character level, their model has been capable of interpreting and generating unseen word forms. Authors have discussed that this method can provide state-of-the-art results for languages which have large variation in word forms.

Luong et al. [4] have presented a novel word-character solution to achieve open vocabulary NMT. The research has focused on building hybrid systems that translate mostly at the word level and consult the character components for rare words. The character-level recurrent neural networks compute source word representations and recover unknown target words when needed. According to their analysis, the two-fold advantage of such a hybrid approach is that it is much faster and easier to train than traditional character-based models.

Lee et al. [35] have introduced a NMT model that maps a source character sequence to a target character sequence without any segmentation. They have employed a character-level convolutional network with max-pooling at the encoder to reduce the length of source representation, allowing the model to be trained at a speed comparable to sub-word-level models while capturing local regularities. Proposed character-to-character model has outperformed a baseline with a sub-word-level encoder on WMT'15 translation tasks. It has been demonstrated that it is possible to share a single character-level encoder across multiple languages by training a model on a many-to-one translation task. In this multilingual setting, the character-level encoder has significantly outperformed the sub-word-level encoder on all the language pairs.

4.6.5 Neural Machine Translation for Low Resourced Languages

The quality of a NMT system depends substantially on the availability of sizable parallel corpora. For low-resource language pairs this is not the case, which results in poor translation quality. Inspired by work in computer vision, Fadaee et al. [36] have proposed a novel data augmentation approach that targets low-frequency words by generating new sentence pairs containing rare words in new, synthetically created contexts. Experimental results on simulated low-resource settings have shown

that the proposed method improves translation quality a significant amount. One drawback of this method is the reduced quality of generated sentences.

Target-side monolingual data plays an important role in boosting fluency for phrase-based statistical machine translation. Sennrich et al. [37] has investigated the use of monolingual data for low resourced languages NMT. Compared NMT models with separately trained language models, they have noted that encoder-decoder NMT architectures already have the capacity to learn the same information as a language model, and explored strategies to train with monolingual data without changing the neural network architecture. By pairing monolingual training data with an automatic back-translation, they have treated it as additional parallel training data, and have obtained substantial improvements.

Gulcehre et al. [38] have investigated how to leverage abundant monolingual corpora for neural machine translation. Compared to a phrase-based and hierarchical baseline, they have obtained considerable improvement on the low-resource language pairs, and on the focused domain specific tasks. While the proposed methods have been initially targeted toward tasks with less parallel data, they have shown that it can also be extended to high resource languages.

4.6.6 Using SMT Features in NMT

Sennrich et al. [39] have shown that the strong learning capability of NMT models do not make linguistic features redundant; they can be easily incorporated to provide further improvements in performance. They have generalized the embedding layer of the encoder in the attentional encoder--decoder architecture to support the inclusion of arbitrary features, in addition to the baseline word feature. They have added morphological features and syntactic dependency labels as input features. They have found that linguistic input features improve model quality according to three metrics: perplexity, BLEU and CHRF3. For languages with large features, this method provides significant improvements in translation quality.

5 METHODOLOGY

In this research, we first implemented a basic NMT system. We then explored the new techniques that can be used to improve NMT performance for under resourced languages. Then we modified existing techniques of improving NMT performance for under resourced languages, to apply them for Sinhala and Tamil language pair.

These techniques can be categorized into preprocessing and post processing. Preprocessing refers to the alteration of training set and the test set before they are used for training or testing. Post processing refers to modification of test results after they are translated by the NMT system.

5.1 Benchmark System

To find the benchmark NMT performance of Sinhala and Tamil, two models were trained using the sentence aligned parallel corpus. Figure 5.1 shows the NMT training process. The NMT system takes one sentence pair at a time and adjusts its parameters to maximize the log likelihood. This process involves three main steps.

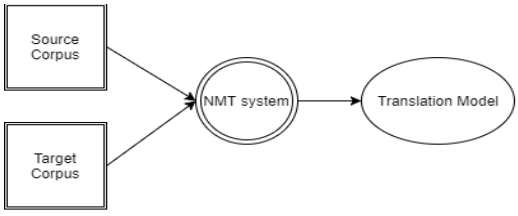

Figure 5.1 : NMT Training

As the first step, a parallel corpus consisting of source side file and the target side file are given as inputs to the NMT system. Using the encoder decoder architecture and attention model, NMT system outputs a translation model which can be used to test the system. Training involves learning the weights of the underlying deep learning network. To avoid model overfitting, we provide a small sentence aligned parallel corpus called the validation set. Validation set ensures that the model parameters are not over trained. When training the system, it periodically checks the model performance using the validation set. If the validation set results decline, that is a clear indication of model overfitting. Hence at this step, the model training is stopped. Figure 5.2 shows a diagram of NMT testing process.

Figure 5.2 : NMT Testing

Using the model trained in the above step, it's possible to translate the test set. As a best practice, we make the intersection between the test set and the training set to a null set. Figure 5.3 shows the translation evaluation process.

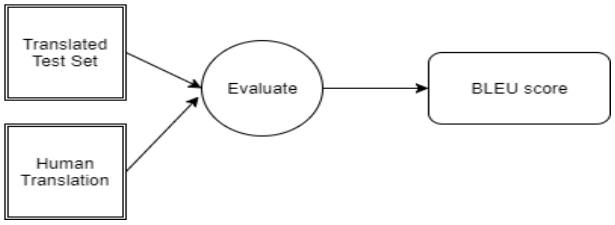

Figure 5.3 : Translation Evaluation

As the last stage, we measure the translation performance of the system using the translated document from the above set and the reference translation that was manually translated by a domain expert. Depending on the trigram matching a score is assigned to each sentence, from which the overall model score is calculated.

5.2 Adding word phrases

This is a novel method that we present to improve NMT performance for under resourced languages. We consider a word phrase as a combination of one or more words that have a specific meaning when taken together. Four types of word phrases that are in domain with this translation task - named entities, domain specific terms, government designations and frequently used phrases, were extracted and added to the training corpus. These word phrases were integrated statically into the NMT system. Figure 5.4 shows the work process of this method.

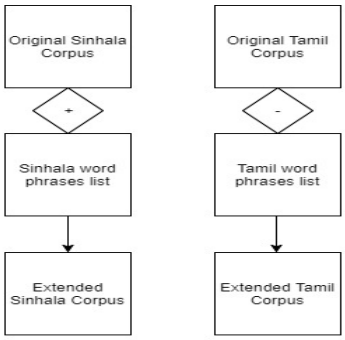

Figure 5.4 : Extending the parallel corpus with word phrases

Using the extended parallel corpora, two models were trained for Sinhala to Tamil and vice versa.

5.3 Use of Monolingual Training Data

Monolingual data are especially helpful if parallel data are sparse, or if there is a poor fit for the translation task, for instance because of domain mismatch. Techniques that can be used to improve the quality of NMT using monolingual data have been identified by Sennrich et al. [37]. According to the authors, adding synthetic target side monolingual data where the source side data are generated using automatic back translation produces better results compared to using dummy source sentences. Hence, we use the synthetic source sentence method to increase the performance of NMT. Back translation of target side monolingual data was done using our extended corpus based system (model that used word phrases) itself.

As the initial step, target side monolingual data are extracted from different sources that are in domain with this machine translation task. New extended corpus is created by adding the target side

monolingual data to the target side original corpus and synthetic source data to the original source side corpus. The resulting corpus is called the extended corpus. Figure 5.5 and 5.6 show this process.

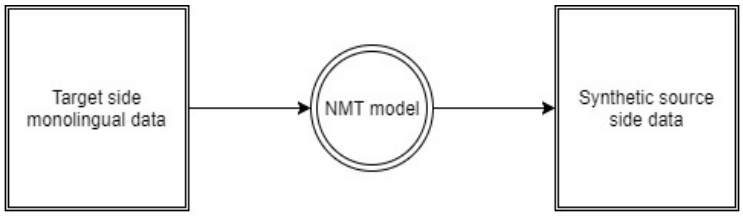

Figure 5.5 : Synthetic Source side data generation

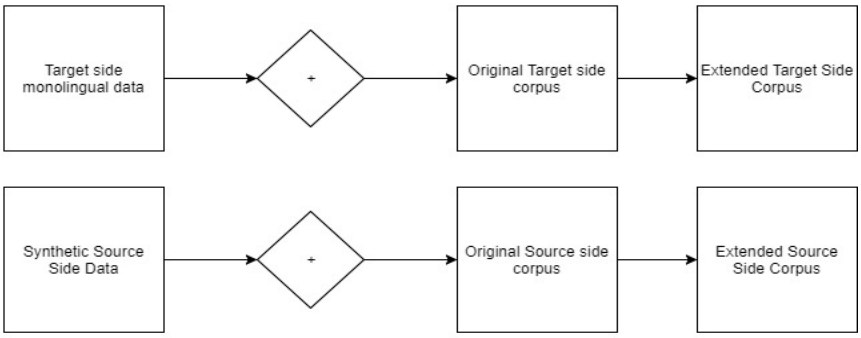

Figure 5.6 : Creating Extended corpus

Finally, two models are trained, Sinhala to Tamil and Tamil to Sinhala, using the extended corpus that was created.

5.4 Data Augmentation

Being an under-resourced language, Sinhala and Tamil NMT falls short of reaching state-of- the-art performances. Limited corpus size directly influences the rare word problem. Being morphologically rich languages, there exist many inflections for each word in both Sinhala and Tamil languages. Hence having many rare words in the corpus is inevitable.

We present two novel sentence pruning techniques based on Parts of Speech (POS) tagging and morphological analysis to remove synthetic sentence pairs that do not preserve language semantics. Compared to Fadaee et al.'s [36] method, POS tagging method and morphological analyzing improve the quality of translation by a significant amount according to our experiments.

For initial synthetic sentences generation, we use the technique used by Fadaee et al. [36]. Initially we obtain a list of rare words by considering the unique words and their counts. Words that appear only R (rare word threshold) times or less are considered as rare words.

In the below expressions, s_i and t_i denote the i^{th} word in the source sentence and target sentence, respectively. Each word in the source sentence is iterated through and substituted by r. Trigram language probability (probability assigned to a group of adjacent 3 words by the language model according to their relative occurrence in the monolingual corpus that was used to train the language model) around r is checked thereafter. If the i^{th} source word is substituted,

Original language probability $\mathbf{p_1 = LM\ (s_{i-1}, s_i, s_{i+1})}$

Synthetic language probability $\mathbf{p_2 = LM (s_{i-1}, r, s_{i+1})}$

if (p_2 >= M*p_1): this is a valid source substitution (M is fluency threshold).

To generate the target side synthetic sentence, we need to substitute the translation of r to the word in the corresponding original target sentence that is aligned to the word that we removed from the source sentence. Statistical approach of automatic word alignment [40] is used to accomplish this task. Using automatic word alignment, it is possible to get the index of the target word that is aligned with the source word that was removed. To get the translation of a rare word r, phrase tables that are generated using word alignment are used.

For a given word e, there exist several possible translations f according to the generated phrase tables. To find the exact translation, we use a two-way translation probability as follows.

translation(e) = argmax f possible translations (p(f|e)*p(e|f))

where,

p (f| e) : Probability of f being the translation of e.

p (e| f) : Probability of e being the translation of f.

If there exists a target side word q corresponding to r, with two-way translation probability greater than T (translation threshold), we select it as a viable translation for r.

q is substituted to the word that is aligned to the word that was removed in source side. If the trigram language probability around that word is greater than M times the original trigram language probability, then we select it as a correct target word substitution. A synthetic sentence pair that satisfies all these conditions is added to the synthetic parallel corpus. To reduce distortion of the meaning, only a single rare word substitution per sentence was allowed. Use of language modeling ensures the fluency of synthetic sentences whereas use of the translation modeling ensures the correspondence between source sentence and target sentence.

Algorithm 1 depicts the initial rare word substitution.

```
dictionary augmented_sentences = {}
For each r in rare_words:
    For each sentence s in corpus:
        For s_i in s:
            p_1 = LM (s_{i-1}, s_i, s_{i+1})
            p_2 = LM (s_{i-1}, r, s_{i+1})
            if(p_2>= M*p_1)
                t = corresponding target side sentence of s
```

t_j = word corresponding to s_i //can be found using word alignment

q = translation of r //can be found using word alignment

$p_1' = LM\ (t_{j-1}, t_i, t_{j+1})$

$p_2' = LM\ (t_{j-1}, q, t_{j+1})$

if($p_2' >= M * p_1'$):

augmented_source = $s_1\ s_2 \ldots\ldots s_{i-1}\ r\ s_{i+1}\ \ldots s_n$, augmented_target = $t_1\ t_2 \ldots\ldots t_{j-1}\ q\ t_{j+1}\ \ldots t_m$

augmented_sentences(augmented_source) = augmented_target

Algorithm 1: Initial Data Augmentation

Table 5.1 depicts an example synthetic sentence pair.

Table 5.1: Example synthetic sentence pairs

Original Sentence Pair	Synthetic Sentence Pair
එසේ පවරා දෙනු ලැබුවේ කවරෙකුටද (/* esea pavaraa denu lAbuwea kavarekuTada*/) - (It was assigned to whom?)	එසේ පවරා දෙනු ලැබුවේ ඔබටද (/*esea pavaraa denu labuwea oba Tada */) (It was assigned to you?)
அவ்வாறு யாருக்கு **ஒப்படைக்கப்பட்டுள்ளது**? (/*avvaaRu yaarukku oppataikkappattuLLadhu*/) (It was assigned to whom?)	**அவ்வாறு யாருக்கு** **உங்களுக்கும்**? (/*avvaaRu yaarukku ungkaLukkum*/) (It was to whom and to you)

5.4.1 POS Tagging

We further increased the quality of synthetic training data by checking the POS tag of each rare word that is substituted. Initially, the original parallel corpus is POS tagged. Then using the methodology proposed in section 7.4, all possible synthetic sentence pairs are generated. Then the synthetic parallel sentences are also POS tagged.

Here,

s_i = word that was removed from source sentence.

t_i = word that was removed from target sentence.

r = rare word that was introduced to source sentence.

t = translation of r that was introduced to target side.

Algorithm 2 depicts the POS tagged bases rare word substitution.

```
if (POS tag of s_i == POS tag of r) and (POS tag of t_i == POS tag of t):
    keep the augmented sentence pair
else:
    remove the augmented sentence pair
```

Algorithm 2: POS tag based pruning algorithm

5.4.2 Morphological Analysis

To further preserve language semantics, we use morphological features. In this research, we pay attention to morphological features of Sinhala nouns only, since most of the rare words are noun word forms. We use two morphological features of Sinhala nouns,

1. Count (වචනය /*wachanaya*/)
2. Case (විභක්තිය /*wibhaktiya*/)

Count can take three values,

1. Definite singular (DS)
2. Indefinite singular (IS)
3. Definite plural (DP)

Case is a suffix that is added to a stem to derive nouns in different meanings. Sinhala language consists of 9 cases [41].

1. ප්‍රථමා (/*prathamaa*/) - Nominative
2. කර්ම (/*karma*/) - Accusative
3. කර්තෘ (/*kartru*/) – Auxiliary
4. කරණ (/*karaNa*/) - Instrumental
5. සම්ප්‍රදාන (/*sampradaana*/) - Dative
6. අවදි (/*awadi*/) - Abalative
7. සම්බන්ධ (/*sambandha*/) - Possesive
8. ආධාර (/*aadhaara*/) - Locative
9. ආලපන (/*aalapana*/) – Vocative

Synthetic parallel corpus that was generated in section 7.4.1 is further improved using morphological features. For a given word, there exists a variable number of case - count combinations. In this

approach, we check whether the case - count combinations of the word that was removed has an intersection with the case - count combinations of the word that is introduced synthetically. We consider it as a semantic preserving sentence pair if there exists an intersection of at least one element.

5.5 Test set splitting

One limitation that hinders the performance of NMT is the inability to translate sentences that have length greater than a certain threshold (usually 30 in practice). In this method, we try to address this issue by splitting the test sentences using language modeling probabilities.

We define the term sentence-splitting as the result of splitting a sentence. A sentence-splitting is expressed as a list of sub-sentences that are portions of the original sentence. A sentence splitting includes a portion or several portions. We use a 3-gram Language Model to generate sentence-splitting candidates.

5.5.1 Probability Based on 3-gram Language Model

The probability of a sentence can be calculated by a language model trained using a corpus. The probability of a sentence-splitting, Prob, is defined as the product of the probabilities of the sub-sentences that result in splitting the sentence. Figure 5.7 depicts a sample sentence splitting.

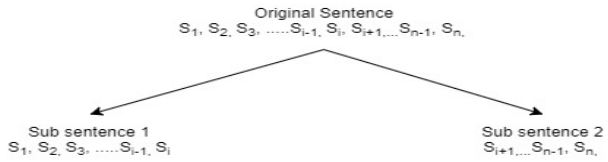

Figure 5.7 : Example Sentence Splitting

P_1 = Probability of original sentence

$P_1 = LM (S_1, S_2, S_3 \ldots S_{i-1}, S_i, S_{i+1}, S_{i+2} \ldots S_n)$

$= Pr(S_1|SOS, SOS)*Pr(S_2|S_1, SOS)*Pr(S_3|S_1, S_2)*..*Pr(S_n|S_{n-1}, S_{n-2})$

P_2 = Probability of sub sentence 1

$P_2 = LM (S_1, S_2, S_3 \ldots S_{i-1}, S_i)$

$= Pr(S_1|SOS, SOS)*Pr(S_2|S_1, SOS)*Pr(S_3|S_1,S_2)*..*Pr(S_i|S_{i-1}, S_{i-2})$

P_3 = Probability of sub sentence 2

$P_3 = LM (S_i, S_{i+1}, S_{i+2} \ldots S_{n-1}, S_n)$

$= Pr(S_i|SOS, SOS)*Pr(S_{i+1}|S_i, SOS)*Pr(S_{i+2}|S_{i+1}, S_i)*..*Pr(S_n|S_{n-1}, S_{n-2})$

To judge whether a sentence is split at a position, we compare the probabilities of the sentence-splitting before and after splitting. When calculating the probability of a sentence and a sub-sentence, we put pseudo words at the head and tail of the sentence to evaluate the probabilities of the head word. This causes a tendency for the probability of the sentence-splitting after adding a splitting position to be lower than that of the sentence-splitting before adding the splitting position. Therefore, when we find

a position with a probability greater than a certain threshold, it is plausible that the position divides the sentence into sub-sentences.

If (p_1 <= $M*p_2*p_3$) then the sentence can be split at the i^{th} position, and the original sentence can be replaced with sub sentences 1 and 2.

Figure 5.8 shows the summary of the test set splitting method.

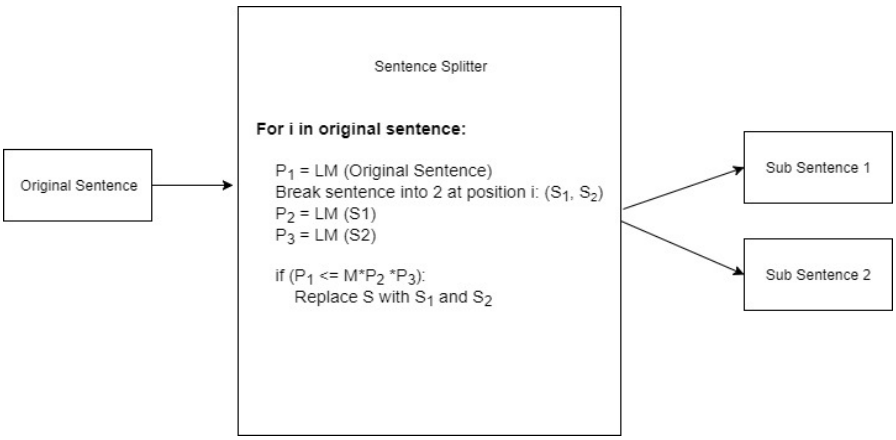

Figure 5.8 : Test Set Splitting

5.6 Transliteration

Sinhala and Tamil have two different alphabets but similar phonetic systems. Yet it is not very accurate to map Sinhala alphabet to Tamil or Tamil alphabet to Sinhala, because in both alphabets there are more than one letter that map to a single sound, where the ideal letter depends on the context.

E.g.: (ந, ன, ண/ න, ණ - /Na/), (ன, ச/ ස - /Sa/).

Also, there are letters in each language that do not have an exact mapping in the other language. E.g. (ඳ /nda/), (ඥ /gna/). Hence, we used English as the common alphabet, which greatly eliminated the above difficulties. We developed a tool that uses a rule based approach to transliterate each alphabet to English (Accuracy: 99.6%). The transliterated parallel corpus and test corpus were trained and tested.

5.7 Byte Pair Encoding

Byte Pair Encoding (BPE) is a popular data segmentation technique. It begins by treating each word as a sequence of characters and iteratively combining most commonly occurring character pair into one. The algorithm stops after a controllable number of operations, or when no token pair appears more than once. It has proven to be an efficient technique to address the rare word problem and the OOV problem. Sennrich et al. [32] have shown that BPE methodology could be adapted specially to improve translation quality of under-resourced, morphologically rich language pairs. Algorithm 3 shows the Byte Pair Encoding python code, as presented in Sennrich et al. [32].

```
def get_stats(vocab):
  pairs = collections.defaultdict(int)
  for word, freq in vocab.items():
    symbols = word.split()
    for i in range(len(symbols)-1):
      pairs[symbols[i],symbols[i+1]] += freq
  return pairs

def merge_vocab(pair, v_in):
  v_out = {}
  bigram = re.escape(' '.join(pair))
  p = re.compile(r'(?<!\S)' + bigram + r'(?!\S)')
  for word in v_in:
    w_out = p.sub(''.join(pair), word)
    v_out[w_out] = v_in[word]
  return v_out

vocab = {'l o w </w>' : 5, 'l o w e r </w>' : 2,
         'n e w e s t </w>':6, 'w i d e s t </w>':3}
num_merges = 10
for i in range(num_merges):
  pairs = get_stats(vocab)
  best = max(pairs, key=pairs.get)
  vocab = merge_vocab(best, vocab)
  print(best)
```

Algorithm 3: Byte Pair Encoding

5.7.1 Byte Pair Encoding (BPE) – Independent Models

We separately used BPE to segment each corpus in to its most commonly occurring sub-words. The independently encoded models ensured that each sub-word unit exists in the training corpus of the respective language.

Following section shows the steps of generation of BPE model and NMT training.

Step 1: Byte Pair encoding of both source side and target side of parallel corpus.

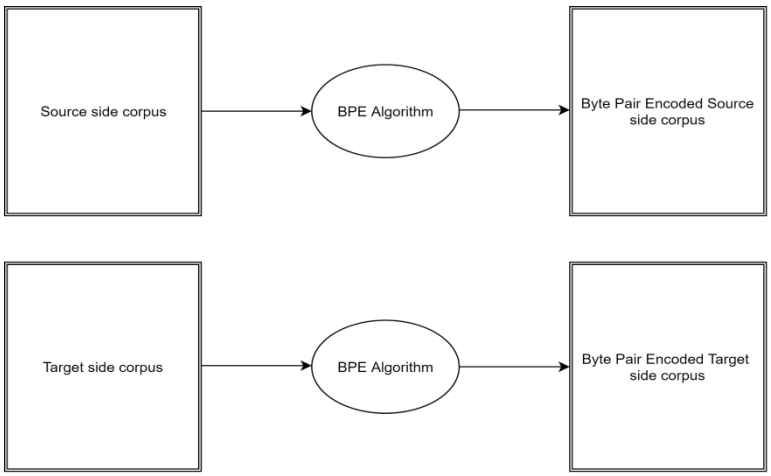

Figure 5.9 : Byte Pair Encoding

Step 2: Training an NMT model using Byte Pair Encoded Corpus.

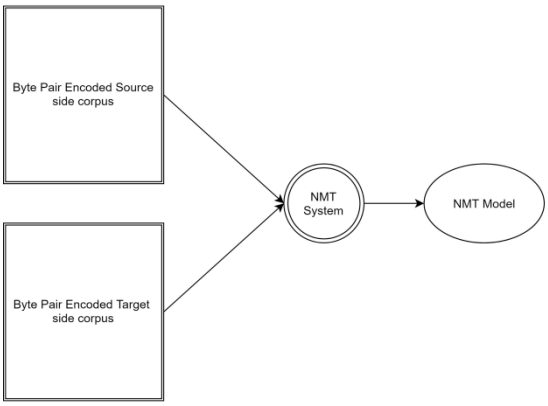

Figure 5.10 : Training NMT Model

Step 3: Byte pair encoding the source side of test set.

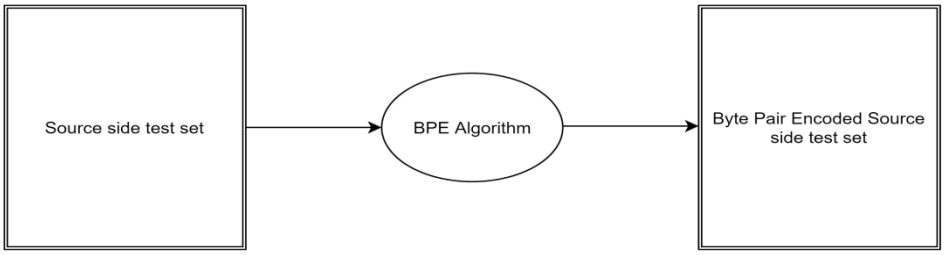

Figure 5.11 : BPE Test Set

Step 4: Translate the byte pair encoded source side test set using the model

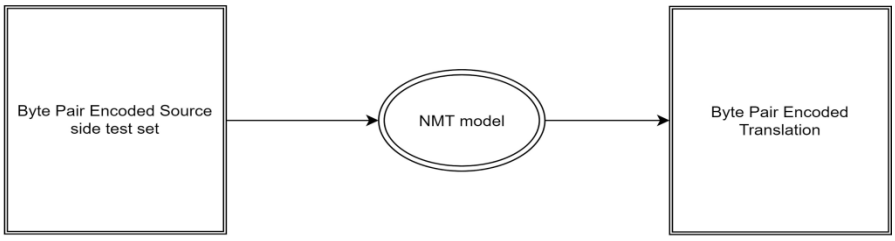

Figure 5.12 : Translation of BPE test set

Step 5: Convert the Byte Pair Encoded translation into word based test set

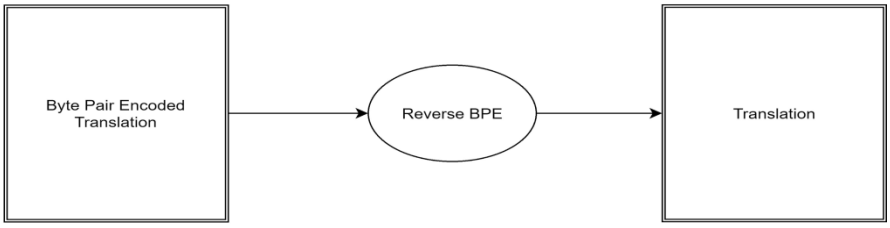

Figure 5.13 : Reversed BPE

Step 6: Evaluate the translation

5.7.2 Byte Pair Encoding – Joint Model

We used the transliterated corpus with the English alphabet to create a joint BPE model. Using a joint model with a single alphabet has proven to be successful for other language pairs with dissimilar alphabets [32]. The joint model ensures that words are segmented in the same manner in both corpuses, though each sub-word unit may not be present in each of the original corpus. The test set too was transliterated. And the translated text from the NMT model was transliterated back into the respective language as a post processing step.

5.8 Parts of Speech Tagging

Being an end to end process, NMT does not rely on feature functions. Yet linguistic input features have additional data that will help to come up with a better translation model. Addressing this fact, we used POS tags as input features in our experiments.

Initially both Sinhala and Tamil sides of parallel corpus were POS tagged. Then two models were trained, Sinhala to Tamil and Tamil to Sinhala.

Figure 5.14 shows the additional steps that we followed in this section.

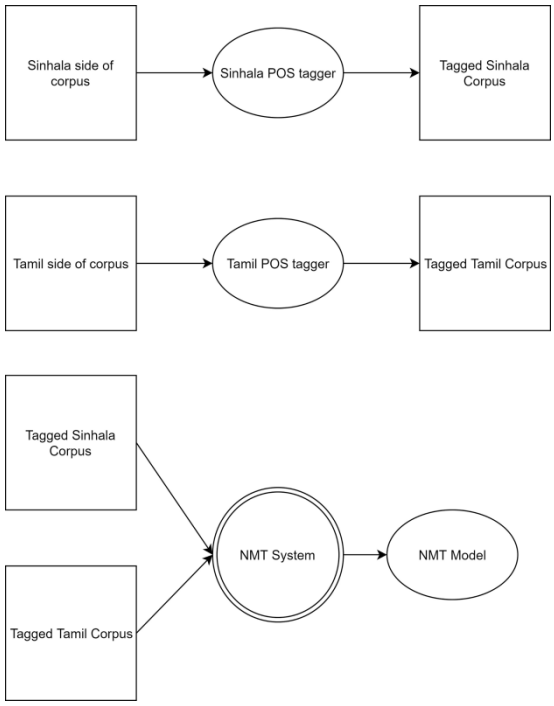

Figure 5.14 : Parts of Speech Tagging

Since the model is trained using POS tagged words, both validation set and the testing set also need to be POS tagged.

5.9 SMT NMT Pipelining

Initially the original parallel corpus is used to train a SMT system. To train a language model, we used a large target side monolingual corpus. Then using this pure SMT model, we translated the original source side of the parallel corpus. We rely on the basis that SMT system can produce translations of the sentences, that it has already seen in the training process. The output of this process is considered as synthetic target side of the parallel corpus. Since SMT does not limit its vocabulary size to a certain limit, we believe that this method will reduce the overall rare word problem, to a considerable level.

Figure 5.15 depicts this process in a diagram.

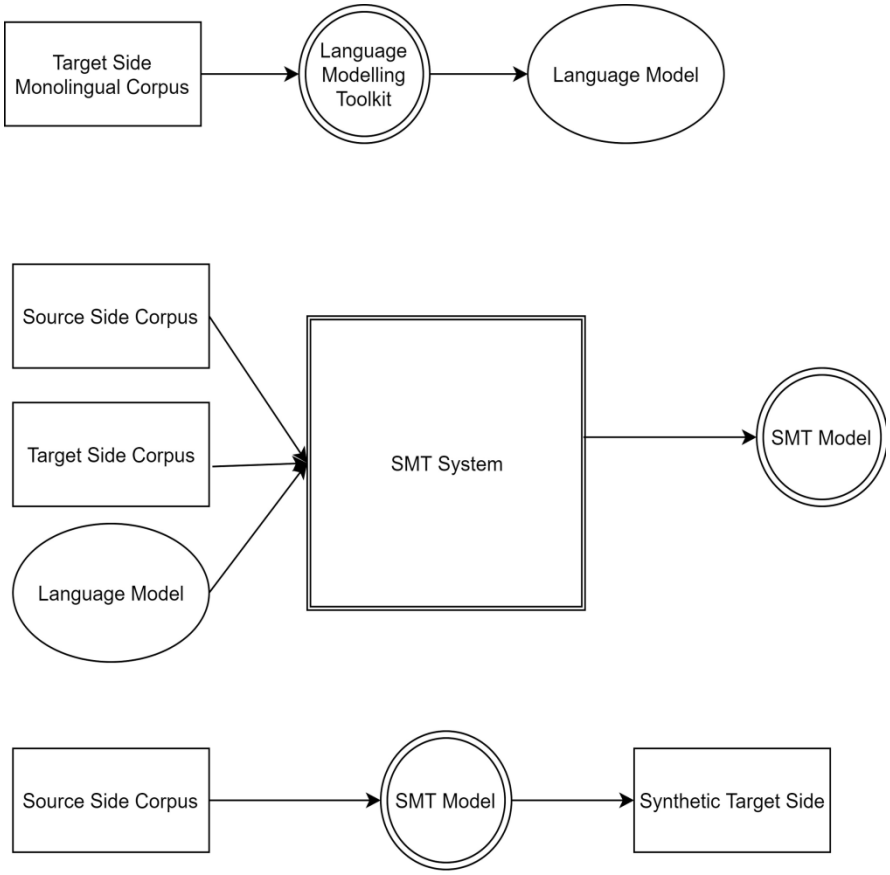

Figure 5.15 : SMT NMT pipelining

Using the synthetic and original target side corpora, we train an NMT system. Since both source and target sides are same, we believe that it would increase the model performance. To evaluate this model, it is necessary to perform the same sequence of actions for both validation set and the test set.

5.10 Use of Word Similarity

Despite the advantages of NMT, NMT models have a major drawback in handling rare words. To control the computational complexity, which grows proportional to target vocabulary size, most NMT systems limit the vocabulary to contain only 30k to 80k most frequent words in both the source and target side and convert rare words into a single **unk** symbol. An obvious problem of this approach is that NMT model cannot learn the translation of rare words. If a source word is outside the source vocabulary or its translation is outside the target vocabulary, the model will not be able to generate proper translation for this word during testing. Another problem is that masking rare words with meaningless **unk** increases the ambiguity of the sentence.

To solve the above problems, Xiaoqing et al. [33] have proposed a novel rare word replacement method based on similarity. During training, word alignment is induced from bilingual corpus. And each aligned word pair which contains rare word either on the source side or the target side is replaced with

similar in-vocabulary words, where the similarity model is learned from a large monolingual corpus. Then this new bilingual corpus with rare words replaced is used to train a NMT model. During testing, the rare words in input sentence are also replaced with similar in-vocabulary words. After translation, a post-processing step is adopted to recover the translation of rare words.

5.10.1 Replacing Rare Words with Similar words

Xiaoqing et al. [33] have proposed to replace rare words in training and testing data with in vocabulary words similar to them. The data processing diagram is shown in Figure 5.16.

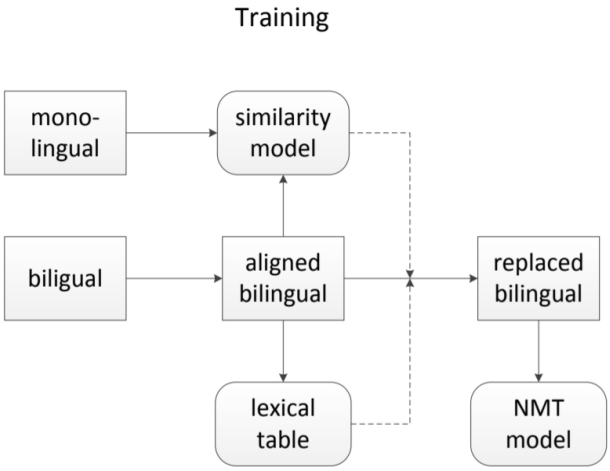

Figure 5.16 : Rare Word Substitution

In the training phase, we first learn a similarity model from a monolingual corpus, which is used to evaluate the similarity between words. We also need to learn word level alignment for sentence pairs in the bilingual corpus. As a byproduct, a lexical translation table can be derived from the aligned bilingual corpus. In our experiments, we only reserve the translation with the highest probability for each word in the table. Then the aligned word pairs which contain rare words are replaced with in-vocabulary words similar to them. Finally, a NMT model is learned from the new bilingual corpus.

In the testing phase, the rare words in testing sentence are first replaced with similar in-vocabulary words. Then the sentence after replacement is translated by the NMT model obtained in the training phase. With the help of the lexical translation model, the translations of those rare words are substituted back into the generated target sentence to obtain the final result. This process is shown in figure 5.17.

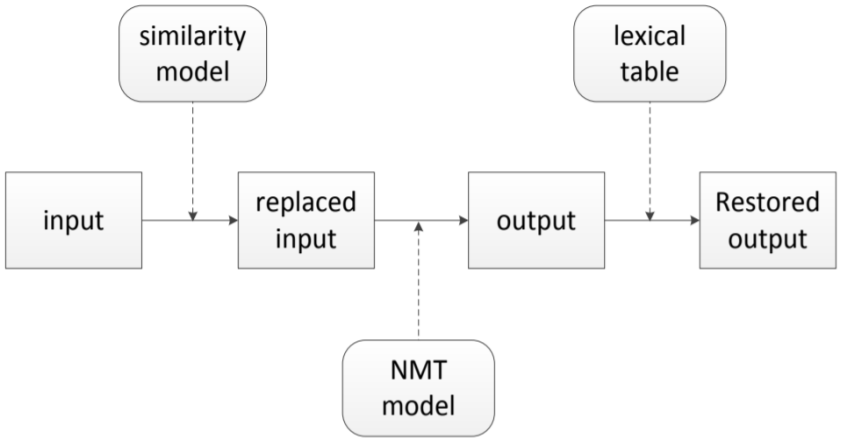

Figure 5.17 : Testing with rare word substituted model

Being morphologically rich languages, Sinhala and Tamil NMT do not come up with the expected performance as mentioned in Xiaoqing et al's [33] work. Hence, we focus on improving the quality of word substitution.

When substituting a word instead of a rare word, we consider the POS tag similarity and morphological features similarity. These features contain important details about the word that is substituted, hence provide better substitution quality.

To improve the gain of this method, we use this method for handling unknown words. Unknown words are the words that are not seen in the training corpus, yet appear in the test set. Figure 5.18 depicts the overall process of rare word and unknown word substitution.

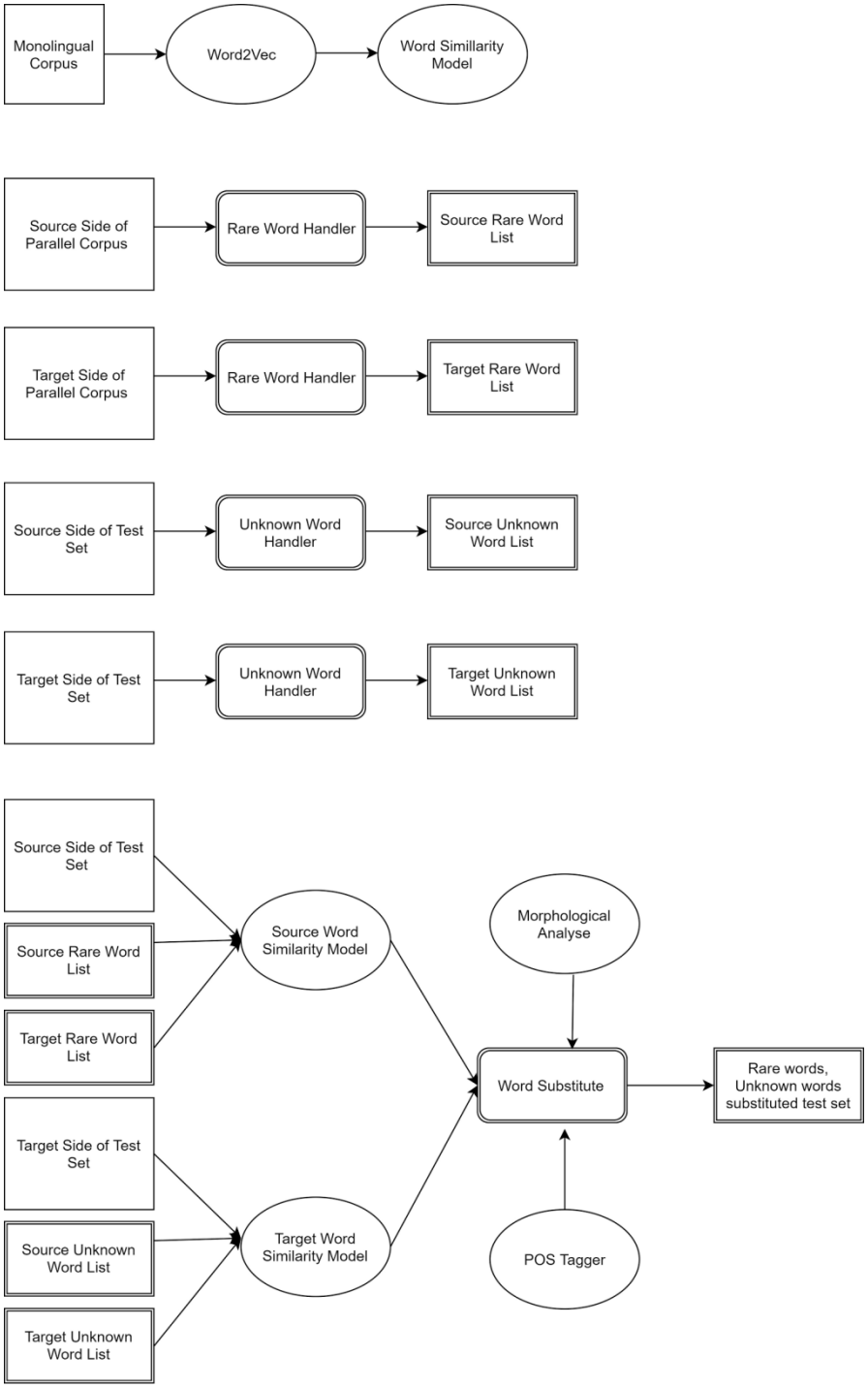

Figure 5.18 : Rare words and unknown words substitution

6 EXPERIMENTAL SETUP

6.1 Data

6.1.1 Si-Ta Dataset

The parallel corpus used in Si-Ta research is created by the extended research team in collaboration with the Department of Official Languages of Sri Lanka. It features the government documents, annual reports, gazette papers, order papers, parliament documents etc. Human translators have ensured the validity of the translation materials used in the data set. The domain is official government documents of Sri Lanka.

The size of the data set is 19, 153 parallel sentences, and the same data set is currently used in other projects of Si-Ta research group [28].

Parallel corpus was divided into 3 parts: training set (14653 sentence pairs), validation set (4000 sentence pairs) and testing set (500 sentence pairs).

6.1.2 Monolingual Data

Monolingual data are used mainly for three different tasks.

1. Language model training
2. Word2Vec model training
3. Target side in domain monolingual data

We used three sources to collect the required monolingual data.

1. News Corpus
   Using publicly available news items in Sinhala and Tamil, we extracted monolingual data by cleaning them.
2. Wiki Dump
   Tamil and Sinhala monolingual corpora were created using the Wikipedia dumps. Initially Sinhala and Tamil Wikipedia dumps were downloaded which were in xml format. The xml tags were removed using "Wiki extractor". The process took nearly 1 hour to create 10 files of extracted data (6800 such files were created). After removing xml tags, extracted files needed to be tokenized and then further cleaned to remove unwanted foreign characters. For tokenizing, tokenizer implemented by researchers in the language research group was used and for further cleaning a script was written using python.
3. Parliament Order Papers
   Set of order papers were cleaned and used as monolingual data.

For the evaluation of Sinhala to Tamil NMT, we use human evaluation and BLEU score measures.

6.1.3 Tools

1. Multi-bleu.perl script - Comes pre-installed with Moses. Given the translated and reference documents it calculates the BLEU score.
2. TildeMT - An online tool which computes different statistical measures for the translation, including BLEU score [42].

6.2 Benchmark System

Parallel corpus was divided into 3 parts: training set, validation set, and testing set. Each dataset consisted of parallel source and target data containing one sentence per line with tokens separated by a space. Validation files were used to evaluate the convergence of the training. To make an unbiased test data set, it was necessary to take the relevant ratios of sentence pairs from different sources.

6.2.1 Pre-Processing

Both Sinhala and Tamil languages contain one or more symbols per character, unlike English. Due to this characteristic of Sinhala and Tamil, existing tokenization tools were not able to tokenize the text, since they identified a single character as two characters. Hence a tokenizer that was specifically developed for Sinhala and Tamil was used in this research.

6.2.2 System Setup

The open source NMT system OpenNMT [15] was used for the experiments. OpenNMT supports standard encoder - decoder architecture with attention mechanism. To evaluate the quality of the translation, Bilingual Evaluation Understudy (BLEU) metric [16] was used. We used the percentage BLEU score values in this research (0 to 100 range).

Using the above parallel corpus, two benchmark systems were trained: Sinhala to Tamil, and Tamil to Sinhala. Training involved two different steps: preprocessing and model training. After completing the preprocessing step, two dictionaries (source dictionary and target dictionary) were generated to index mappings. Using these two dictionaries and the serialized file, a model was trained with 2-layer Long Short-Term Memory with 500 hidden units on both encoder and decoder. Since most of the operations inside the network were numeric and easily parallelizable, NVIDIA TESLA C2070 with GPU memory 5.5 GB was used to speed up the process.

6.3 Adding Word Phrases

Four types of word phrases that are in domain with this translation task were extracted and added to the training corpus.

1. Set of named entities-11,561 pairs
2. Set of common domain specific terms and phrases- 19,861 pairs
3. Set of government designations- 5,291 pairs
4. Frequently used phrases that are used in government documents (Letter heads, salutations etc.) - 610 pairs.

Tables 6.1, 6.2, 6.3 and 6.4 depict example word phrases for each category.

1. A set of named entities

Table 6.1: Example named entities

Sinhala	Tamil
බන්ධනාගාරය	சிறையில்

නගර සභාව	நகராட்சி மன்றம்
ඉංගිරිය	இங்கிரிய
පාර්ලිමේන්තුව	பாராளுமன்றத்தில்

2. A set of common domain specific terms and phrases

Table 6.2: Example domain specific terms and phrases

Sinhala	Tamil
අපද්‍රව්‍ය කළමනාකරණ ව්‍යාපෘතිය	கழிவு மேலாண்மை திட்டம்
ජල සැපයුම් අංශය	நீர் வழங்கல் பிரிவு
පුහුණුව	பயிற்ச
සුබසාධන මණ්ඩලය	நலன்புரி வாரியம்

3. A set of government designations

Table 6.3: Example government designations

Sinhala	Tamil
මුරකරු	கவனிப்பவர்
ඉංජිනේරු	மூத்த பொறியாளர்
කොමසාරිස්	ஆணையாளர்
ජෙනරාල් අධ්‍යක්ෂ	பொது இயக்குனர்

4. Frequently used phrases that are used in government documents.

Table 6.4: Example frequently used phrases

Sinhala	Tamil
වැඩපත සම්බන්ධ පරික්ෂාව	பட்டறை சோதனை
මගේ අංකය	என் எண்
නානායක්කාර මෙනවිය	மிஸ். நானாயக்கார
ලිපිනය	முகவரி

To find the effect of number of word phrases for the BLEU score, a comprehensive analysis was carried out. We trained separate models for Sinhala to Tamil, and Tamil to Sinhala by adding 5000 more-word phrases to the initial training dataset each time. Experiments were carried out for 5k, 10k, 15k, 20k, 25k, 30k, 35k, 40k, 45k and 47k number of word phrases.

6.4 Monolingual Training Data

Target side in-domain monolingual data were extracted using official letters. Maximum sentence length was set to 30 to avoid the performance issues in NMT systems when sentence length is large. 10, 000 target side parallel sentences for each language were extracted and back translated. Model that was trained using the extended corpus that included word phrases was used to back translate. Since the generated synthetic source side data had translation errors, we restricted number of synthetic sentence pairs to be less than the number of sentence pairs in the original parallel corpus, to make sure that the overall quality of resulting parallel corpus remains acceptable. Two models were trained for Sinhala to Tamil and Tamil to Sinhala separately.

6.5 Data Augmentation

6.5.1 Initial data augmentation

Out of 15383 number of unique words, 6421 words appeared only once in the Sinhala language side whereas corresponding values for Tamil side was found to be 31186 and 17238, respectively. Hence, we chose rare word threshold R to be 1. Considering the tradeoff between the number of sentences generated and semantic preservation of synthetic data, we chose fluency threshold M to be 2 and translation threshold T to be 0.9.

6.5.2 POS tagging

Both original and synthetic parallel corpora that were generated in the previous section were POS tagged. We used the POS tagger developed by Fernando et al. [43] for Sinhala, and the POS tagger developed by the Computational Linguistic Research Group [44] for Tamil.

6.5.3 Morphological Analysis

We considered morphological features only when generating the Sinhala side of the parallel corpus. We used Helabasa - Noun Analyzer [45] to retrieve Sinhala morphological features. When training the translation model for each technique, we appended the synthetic corpus generated from that technique to our original corpus in one to one ratio and trained separate models.

6.6 Test Set Splitting

We initially trained a language model using the monolingual news corpus. Using the trained language model, we checked each test sentence for the possible splitting. To reduce the distortion of the resulting testing set, we introduced only a single splitting position for each sentence. When splitting the sentences according to the probability values we observed some notable issues which lead to erroneous sentence splitting.

1. Initials of names which are separated by a dot (.) were chosen as possible splitting points. To overcome this issue we manually corrected such erroneous occurrences in the resulting testing set.
2. English letters and numbers that were present in the testing set were chosen as possible splitting positions. To overcome this issue, we purposely ignored English letters and numbers when checking for the possible splitting positions.

We choose M to be 0.07, the probability factor, to address the tradeoff between number of test set splitting and the quality of resulting test set. Higher values of M result in high quality test set, but with very few number of splitting points, whereas low M value results in many number of test set splitting with reduced quality.

6.7 Transliteration

We transliterated both the source side and target side of the parallel corpora into English. The resulting corpora were used to train the NMT model. Both validation and test set had to be transliterated as well. To evaluate the model, the resulting machine translation had to be re transliterated into native language.

Table 6.5 and 6.6 show the letter mappings from Sinhala to English and Tamil to English respectively.

Table 6.5: Mappings of Sinhala and English characters

Sin-Eng	Sin-Eng	Sin-Eng	Sin-Eng	Sin-Eng
['අ'] = 'a'	['ෛ'] = '-I'	['ද'] = 'da'	['ශ'] = 'Sa'	['ී'] = '-'
['ආ'] = 'aa'	['ක'] = 'ka'	['හ'] = 'ha'	['ෂ'] = 'sha'	['ෘ'] = '-ru'
['ඇ'] = 'A'	['බ'] = 'ba'	['න'] = 'na'	['ෆ'] = 'fa'	['ඖ'] = 'au'

['ஈ'] = 'Aa'	['஗'] = 'ga'	['ங'] = 'Na'	['ஜ்'] = 'GNa'	['ஹ'] = 'nnda'
['ஒ'] = 'i'	['ம'] = 'ma'	['ப'] = 'a'	['ஸ்'] = 'KNa'	['ஶ'] = 'nndha'
['ஓ'] = 'ie'	['வ'] = 'ca'	['ள'] = 'La'	['க்ஷ'] = 'jha'	['ஹ'] = 'nnga'
['ெ'] = 'u'	['ய'] = 'ya'	['ை'] = '-ie'	['ள'] = 'Lu'	['ஷ'] = 'Ka'
['ூ'] = 'uu'	['ஜ'] = 'ja'	['ே'] = '-ei'	['ளு'] = 'Luu'	['ஸ'] = 'Ga'
['ஏ'] = 'e'	['ர'] = 'ra'	['ொ'] = '-oe'	['ஃ'] = '-A'	['ஜ'] = 'cha'
['ஏ'] = 'ea'	['ற'] = 'Ta'	['ீ'] = '-uu'	['ை'] = '-i'	['ஷ'] = 'Tha'
['ஶ்ரீ'] = 'I'	['ல'] = 'la'	['ௌ'] = '-au'	['ா'] = '-aa'	['ஐ'] = 'Da'
['ஔ'] = 'o'	['ஷ'] = 'Da'	['்'] = '\\n'	['ஃ'] = '-Aa'	['ஸ'] = 'tha'
['ெ'] = '-e'	['வ'] = 'wa'	['்ஹ'] = '\h'	['ர ்'] = '-R'	['ஷ'] = 'dha'
['ு'] = '-u'	['ந'] = 'ta'	['ஷ'] = '\N'	['் ய'] = 'Ya'	['ஏ'] = 'a'
['ொ'] = '-o'	['ஸ'] = 'sa'	['ஶ'] = 'Ba'	['் ர'] = 'ra'	['ஹ'] = 'bha'

Table 6.6: Mappings of Tamil and English characters

Ta-Eng	Ta-Eng	Ta-Eng	Ta-Eng
['ஔ'] = 'au'	['൫'] = '-5'	['ி'] = '|i'	['க'] = 'ka'
['ஐ'] = 'ai'	['൬'] = '-6'	['ீ'] = '|ii'	['ஸ'] = '-sa'
['அ'] = 'a'	['൭'] = '-7'	['ௌ'] = '|au'	['ர'] = 'ra'
['ஆ'] = 'aa'	['அ'] = '-8'	['ை'] = '|ai'	['ற'] = 'Ra'
['எ'] = 'e'	['൯'] = '-9' # x	['ா'] = '|aa'	['ட'] = 'ta'
['ஏ'] = 'ee' # E	['இ'] = 'i'	['்'] = '|'	['ப'] = 'a'
['ஈ'] = 'ii' # I	['உ'] = 'u'	['ஞ'] = 'nja'	['ம'] = 'ma'
['ஊ'] = 'uu' # U	['ஒ'] = 'o'	['ங'] = 'nga'	['ய'] = 'ya'
['ஓ'] = 'oo' # O	['ஃ'] = 'q'	['ஷ'] = 'sha'	['ன'] = 'na'

['கூ'] = '-1000'	['ெ'] = '|e'	['ந'] = 'wa'	['ண'] = 'Na'
['ற'] = '-100'	['ே'] = '|ee'	['த'] = 'dha'	['ல'] = 'la'
['ப'] = '-10'	['ொ'] = '|o'	['ச'] = 'sa'	['ள'] = 'La'
['உ'] = '-2'	['ோ'] = '|oo'	['ழ'] = 'za'	['வ'] = 'va'
['ஹ'] = '-3'	['ு'] = '|u'	['ஜ'] = 'ja'	
['ச'] = '-4'	['ூ'] = '|uu'	['ஹ'] = '-ha'	'

6.8 Byte Pair Encoding

6.8.1 Byte Pair Encoding (BPE) – Independent Models

We separately used BPE to segment each corpus in to its most commonly occurring sub-words. The independently encoded models ensured that each sub-word unit exist in the training corpus of the respective language.

In BPE, each word is initially segmented as a character sequence. For BPE-based systems, we varied the number of BPE merge operations from 1,000 to 17,000 for Sinhala and from 1000 to 26,000 for Tamil. Instead of using a fixed vocabulary cutoff, we used the full vocabulary; to ensure the model still learns how to deal with unknown words.

After making the Byte pair encoded version of the parallel corpus, 2 NMT models were trained. Since the model is trained to work at sub-word level, both validation set and the testing set had to be Byte Pair encoded.

To test the accuracy of the model, the machine translated output was reverse byte pair encoded to the native representation.

6.8.2 Byte Pair Encoding – Joint Model

We used the transliterated corpus with the English alphabet to create a joint BPE model. The joint model ensures that words are segmented in the same manner in both corpuses, though each sub-word unit may not be present in each of the original corpus. The test set too was transliterated. And the translated text from the NMT model was transliterated back into the respective language as a post processing step.

6.9 Parts of Speech Tagging

In our experiments, we associated each word with one POS tag. If the POS tag for a word is unknown, we use a unique tag ‘NOTAG’. The POS tagged corpus was used to train the model. Each 3 sets, training set, validation set and testing set were POS tagged in the same manner.

6.10 SMT NMT Pipelining

Initially a language model was trained for each language Sinhala and Tamil, using the monolingual data that we had collected. Using the respective target side language model and the original parallel corpus, corresponding SMT models were generated.

The source side of the training set was translated using the trained SMT model. Since the model was trained using the same data set, the accuracy of the resulting target side sentences was in good quality. Two NMT models were trained using the resulting target side corpus as the source side and original target side corpus as the target side.

6.11 Word Similarity

Initially we trained word similarity models using word2vec for each language Sinhala and Tamil. Resulting models were used to find the most similar word or a set of words, for a given word. A list of rare words and unknown words were prepared for each language. Each rare word and the unknown word that is present in the test set were replaced by a similar, non-rare and seen word that is in the training set.

In our experiments, we did not alter the translated words. We rely on the assumption that if two source words are similar to each other, their corresponding translation should be the same.

7 RESULTS AND DISCUSSION

Table 7.1 shows the BLEU scores obtained for each method. Rows in red color show the novel methods that we are proposing, while the rows in blue and purple represent the methods that we have improved and methods that are applied as it is.

Table 7.1: BLEU scores

Method	Tamil to Sinhala	Sinhala to Tamil
Benchmark	6.84	6.78
Monolingual Training Data	10.27	6.81
SMT-NMT Pipelining	6.78	6.56
Data Augmentation	11.84	8.94
Test Set Splitting	6.92	6.89
Byte Pair Encoding	10.47	6.78
Parts of speech tagging	6.93	6.81
Word Similarity	6.89	6.82
Transliteration	8.36	6.79
Adding word phrases	12.26	7.46

7.1 Benchmark Model

Compared to machine translation tasks that use large corpus size, the BLEU scores that we obtained are comparatively low. Yet, considering the corpus size we used to train the system, this result seems to have an edge over general machine translation tasks. Since we used a domain specific corpus when training the system, this result is acceptable. Unlike general machine translation tasks, the language flow is quite uniform and the vocabulary is small in domain specific translation. Hence the underlying neural network does not have to concern about many language patterns, while training the model.

Table 7.2 and 7.3 show example translations obtained from this method.

Table 7.2: Benchmark Sinhala to Tamil translations

	Source Sentence	Reference Translation	Machine Translation
1	ශ්‍රී ලංකා රූපවාහිනී සංස්ථාව	இலங்கை ரூபவாஹினிக் கூட்டுத்தாபனம்	இலங்கை ரூபவாஹினிக் கூட்டுத்தாபனம்
2	අධ්‍යාපන අමාත්‍යාංශයේ අතිරේක ලේකම්වරුන්	கல்வி அமைச்சின் மேலதிக செயலார்கள்	கல்வி அமைச்சின் செயலாளர் செயலாளர்கள்
3	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி அமைச்சுடன் பட்டப் பாடநெறி (ஆங்கில தர கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)

Table 7.3: Benchmark Tamil to Sinhala translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් □□□□ය නිලධරයන්ට ගෙවන ගාස්තු .	සිවිල් හානියට පත් කිරීම් පිළිබඳ □□□□ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතුය .	එසේම දෙපාර්තමේන්තු ප්‍රධානියා දෙපාර්තමේන්තු ප්‍රධානියා වෙත යැවිය යුතු ය .

3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ.	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තිය සංශෝධනය කර ඇත .

Words that appear in blue color are the correctly translated words. When analyzing the above outputs, it can be noticed that although the translations are incorrect, the grammatical sentence structure has been preserved in many cases. For an instance, "එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තිය සංශෝධනය කර ඇත" has a clear meaning, although it is not the expected translated with respect to reference translation. One notable aspect found in the translations is the model's ability to correctly output numerical values. The model has not been able to correctly translate named entities.

BLEU score does not account for the actual word meaning. This leads to reduced BLEU score. When analyzing the sample translations, we can observe this fact. Word "එලෙසම" and "එසේම" in the second example have identical meaning. But when computing the BLEU score, these words are treated as completely different lexical terms. This leads to produce a low BLEU score. Being morphologically rich languages, having subtle differences in words is unavoidable.

7.2 Adding Word Phrases

Adding word phrases has increased the BLEU score by 0.68 for Sinhala to Tamil translation, and by 5.4 for Tamil to Sinhala translation. This BLEU score gain is due to two main factors.

Firstly, adding word phrases increases the corpus size. It should be noted that the word phrases that were added are not complete sentences, but contain only 2-3 words per phrase. Second reason for the BLEU score gain is the nature of domain-specific language translation behavior. In this research, we used official government documents as our domain. Word phrases included a significant amount of named entities that are widely used in official government documents. Even though there is no explicit language model in NMT, the decoder considers the last translated word when assigning the probability to the next translated word. There is a high chance that a named entity appears only once in the original training corpus. Hence, a low probability would be applied for the correct next word, due to low presence of two adjacent words in the corpus. Adding word phrases helps to increase this probability, thus reducing rare word problem.

Table 7.4 and 7.5 show example translations obtained from this method.

Table 7.4: Word phrases example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	අධ්‍යාපන අමාත්‍යාංශයේ අතිරේක ලේකම්වරුන්	கல்வி அமைச்சின் மேலதிக செயலார்கள்	கல்வி அமைச்சின் மேலதிக செயலாளர்கள்

2	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)
3	රාජ්‍ය භාෂා කොමසාරිස්.	அரசகருமமொழிகள் ஆணையாளர் .	அரசகருமமொழிகள் ஆணையாளர் .

Table 7.5: Word Phrases Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛ□□ය නිලධරයන්ට ගෙවන ගාස්තු .	සමෘද්ධි වන්දි ගෙවීම සඳහා කටයුතු සඳහා වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .	එසේම දෙපාර්තමේන්තු ප්‍රධානියා විසින් නිවාඩු අනුමත කළ යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වැනි වගන්තියේ විධිවිධාන අදාළ වේ .

Compared to benchmark model, this method shows significant improvement in handling named entities. Out of all approaches we tested, this is the method which gave the highest improvement. When we consider the errors in the above examples, it can be noticed that error nous translated segments (වැනි, එසේම, ප්‍රධානියා විසින්) have very close meaning to that of reference translation.

The factor that long sentences produce less BLEU score is evident even in this model. Yet, when we consider the BLEU score for each sentence, it is evident that majority of the translations have reached a good level of fluency. Out of all the improvements that we carried out, this method produced the highest gain with respect to the benchmark model.

Figure 7.1 shows the graph that depicts BLEU score against the number of word phrases. When the number of word phrases is increased by 5000, the increase in BLEU score is 0.0723 BLEU points in average for Sinhala to Tamil translation and 0.5744 BLEU points for Tamil to Sinhala translation. Hence, we can conclude that increasing number of word phrases makes the NMT results more accurate.

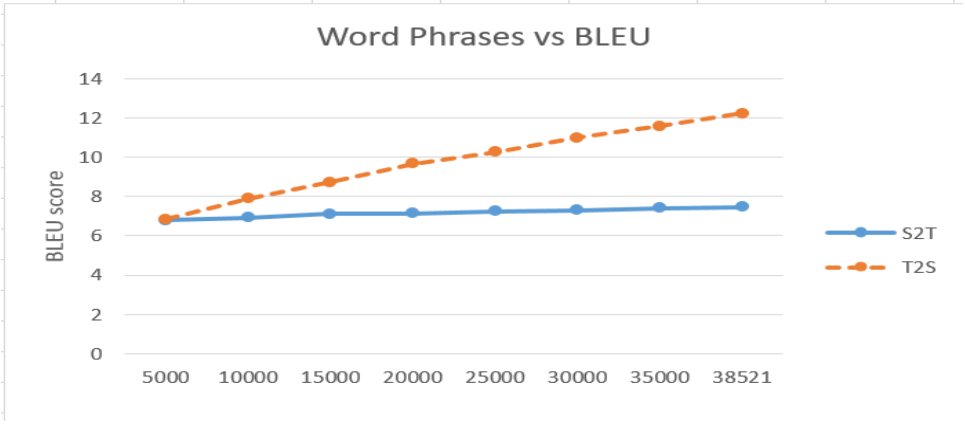

Figure 7.1 : BLEU score vs number of word phrases

7.3 Monolingual Training Data

Use of target side monolingual data improved the translation quality by 0.13 for Sinhala to Tamil and 3.43 for Tamil to Sinhala. A central theoretical expectation is that monolingual target-side data improve the model's fluency, and its ability to produce natural target-language sentences.

This BLEU score gain is due to two factors. Target side monolingual data play a vital role in language modeling in SMT. Being an end to end process, NMT does not have a separate language model. Yet in the decoder, NMT system considers the previously translated word when predicting the new translation. Hence when in-domain target side monolingual data are added to the training corpus by automatically back translating them to source side, NMT system can take advantage of language specific features in the target side.

Second major reason for BLEU score gain is the increased corpus size. Even though the quality of back translated source side is low, compared to our original parallel corpus, the model output was increased due to the increased number of sentences in the parallel corpus.

Table 7.6 and 7.7 show example translations obtained from this method.

Table 7.6: Monolingual training data example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	අමාත්‍යාංශය සහ අනුබද්ධ ආයතන වල නිලධාරීන් 30 ක් පමණ සහභාගි විය .	அமைச்சு மற்றும் அதன் உள்ளக நிறுவனங்களின் உத்தியோகத்தர்கள் 30 பேர் கலந்து கொண்டனர் .	அமைச்சு மற்றும் நீரியல் நிறுவனங்கள் தொடர்பான உத்தியோகத்தர்கள் பேர் கலந்துகொண்டனர் .
2	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் 1 ஆம் பிரிவின் பார்க்கவும் .
3	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி மேன்முறையீட்டு பட்டப் பாடநெறி (ஆங்கில தர கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)

Table 7.7: Monolingual training data example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள்.	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු .	කම්කරුවන්ට වන්දි ගෙවීම පිළිබඳ කටයුතු සඳහා වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එලෙසම දෙපාර්තමේන්තු ප්රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .	එසේම දෙපාර්තමේන්තු ප්රධානියෙකු ද , ඔහුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එවැනි නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ සඳහන් පරිද්දෙනි .

When analyzing the output, it can be noticed that the fluency of the language has improved significantly. For an instance, for the sentence "தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள்", the output generated by the benchmark model, (සිවිල් හානියට පත් කිරීම් පිළිබඳ වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු) has been improved significantly by this model (කම්කරුවන්ට වන්දි ගෙවීම පිළිබඳ කටයුතු සඳහා වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු). The language flow in the output generated in this method is much fluent than the benchmark model. Hence, we can empirically prove that use of target side monolingual data improves model's fluency.

7.4 Data Augmentation

Table 7.8 provides examples resulting from each augmentation procedure.

Table 7.8: Example synthetic data with highlighted

[original / **substituted**] and [original /**translated**] words

Method	Example
Initial Data Augmentation	Si: එතුමා මෙම සභාවට [දන්වන්නෙහිද / **බඳවාගැනීමට**]? (/*etumaa mema sabhaavaTa [danwannehida / **banndhavaagAniemaTa**] ?*/) Ta: அவர் இச்சபைக்குத் [தெரிவிப்பாரா/ **ஆட்சேர்ப்பிற்கு**]? (/*avar issapaikkudh [dherivippaaraa/ **aatseerppiRku**]*/) (En: For this session, he [will inform/ for hiring]?)
Part of speech tagging	Si: පුහුණු [සැලස්මට / **බඳවාගැනීමට**] අදාල තොරතුරු (/*puhuNu [sAlAsmaTa / **banndhavaagAniemaTa**] adaala toraturu*/) Ta: பயிற்சித் திட்டத்திற்கு [பொருத்தமான / **ஆட்சேர்ப்பிற்கு**] தகவல்கள் (/*payiRsidh dhittadhdhiRku [porudhdhamaana / aatseerppiRku] dhakavalkaL*/) (En: Information [related to training planning/ related to training hiring])
Morphological Analysis	Si: පහත සඳහන් ලිපිනයට කරුණාකර [ලදුපතක් / **කබායක්**] එවීමට කටයුතු කරන්න. (/*pahata sanndhahan lipinayaTa karuNaakara [ladupatak / **kabaayak**]ewiemaTa kaTayutu karanna.*/) Ta: கீழ் காணும் முகவரிக்கு தயவு

	செய்து [பற்றுச் சீட்டொன்றை / **மழைக்காப்பு**]அனுப்ப **நடவடிக்கை எடுக்கவும்**. (/*kiiz kaaNum mukavarikku dhayavu seydhu [paRRus siittonRai / mazaikkaappu]anuppa watavatikkai etukkavum.*/) (En: Kindly send a [receipt/coat] for the following address)

Considering Table 7.8, in the initial data augmentation method, substituting බඳවාගැනීමට (/*banndhavaagAniemaTa*/ - to hire) with දන්වන්නෙහිද (/*danwannehida*/ - will inform?) makes the resulting synthetic sentence meaningless. Sentences that are generated by analyzing POS tags seem to have an edge over initial data augmentation method. Since සැලස්මට (/*salasmataTa*/ - for plan) and බඳවාගැනීමට (/*banndhavaagAniemaTa*/ - to hire) (second row) have identical POS tags, high fluency is achieved in the resulting synthetic sentence pair. Synthetic sentence pair generated using morphological analyzing, preserves the meaning to a better extent. Word ලදුපතක් (/*ladupatak */ - receipt) and කබායක් (/*kabaayak */ - coat) have indefinite singular –Nominative, Accusative, Auxiliary, Locative morphological features in common.

Table 7.9 depicts the BLEU scores obtained for each method.

Table 7.9: Data Augmentation BLEU scores

Method	Si-Ta	Ta-Si
Benchmark Training	6.78	6.84
Initial Data Augmentation	7.68	8.86
POS Tagging	8.72	10.98
Morphological Analysis	8.94	11.84

To examine the impact of augmenting training data by creating contexts for rare words on the target side, we tested how each model performs on rare words. Most of the rare words are not ‘rare’ anymore in the augmented data since they were augmented sufficiently many times.

Synthetic data generated using the initial data augmentation method have improved the performance of Si-Ta and Ta-Si translation by 0.9 and 2.02 amounts, respectively. To verify that this gain is due to the rare word substitutions and not just due to the repetition of part of the training data, we performed an experiment where each sentence pair selected for augmentation is added to the training data unchanged (i.e. without creating synthetic data). This simple form of sampled data replication delivered 0.53 and 1.42 BLEU score gains for Si-Ta and Ta-Si, respectively. Hence initial data augmentation models have performed better compared to simple data replication method.

Use of POS tags has achieved 1.04 and 2.12 BLEU score gains over the initial data augmentation for Si-Ta and Ta-Si, respectively. Human evaluators who oversaw the quality of generated sentences revealed that the use of POS tags has increased the fluency of language and rare word translation performance by a significant amount. Thus, we can empirically prove that the use of POS tags improves the quality of synthetic training data, which in turn reduces the rare word problem in NMT.

Morphological features have played a vital role in reducing the rare word problem. When generating synthetic sentence pairs, we considered only Sinhala language morphological features. Being morphologically rich, there exist many number of variations for a given root word in Sinhala. Hence, checking the case-count combinations of a word when substituting, helps to preserve language semantics of the generated sentence. This is evident by analyzing the BLUE score gains of 1.26 and 2.98 for Si-Ta and Ta-Si translations compared to the initial data augmentation method.

BLUE score gains are consistent across both translation directions, regardless of whether rare word substitutions are first applied to Sinhala or Tamil. Hence it can be verified that using POS tagging and morphological features results in generating quality synthetic parallel data that preserve language semantics, which eventually leads to better translation performance.

Though overall rare word translation quality was improved by our methods, there were several cases where augmentation resulted in incorrect outputs that were correctly translated by our benchmark system. Table 7.10 corresponds to such an incorrect translation.

Table 7.10: Incorrect Outputs

Source Sentence	8. **உள்ளூர்/ வெளிநாட்டு திரைப்ப ட தயாரிப்பாளர்களுடன்** /*uLLur/ veLiwaattu dhiraippa ta dhayaarippaaLarkaLutan*/ - (With local and foreign film producers)
Reference Translation	8 .දේශීය / විදේශීය චිත්‍රපට නිෂ්පාදකයින් සමඟ /*8 .deaSieya / videaSieya citrpaTa nishpaadakayin samannga */ - (With local/foreign film

	producers)
Benchmark Translation	8 .දේශීය විදේශීය විකාශන කටයුතු සඳහා/* 8 .deaSieya videaSieya vikaaSana kaTayutu sanndhahaa*/ - (For local/ foreign broadcasting)
Data Augmentation	(8) විදේශ චිත්‍රපට ගනුදෙනු කිරීම /* (8) videaSa citrapaTa ganudenukiriema . */ - (For trading foreign films)

Our benchmark model has been able to correctly translate දේශීය (/*deaSieya*/ - local) and විදේශීය (/*videaSieya*/ - foreign) terms, whereas our new model has not been able to translate any of them. If the language model selects substitutions that have low probabilities, it results in generating outputs with low fluency. Another possible reason is errors in word alignments. If the word alignments are erroneous and phrase table contains faulty probabilities, this may lead to synthetic sentence pairs that do not correspond to each other.

7.5 Test Set Splitting

Table 7.11 and 7.12 show example translations obtained from this method.

Table 7.11: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	මාණ්ඩලික නිලධරයෙකු , ක්ෂේත්‍ර නිලධරයෙකු හෝ අතිකාල දීමනා ලැබීමට හිමිකමක් නැති වෙනත් නිලධරයෙකු රජයේ නිවාඩු දිනයක වැඩ කළ විට ඔහුට ලබාගැනීමට අවසර ඇත්තේ ඒ වෙනුවට දිනක හිලව් නිවාඩුවක් පමණි .	பதவிநிலை உத்தியோகத்தர் ஒருவர் , வெளிக்கள உத்தியோகத்தர் ஒருவர் அல்லது மேலதிக நேரப்படியினைப் பெறுவதற்கு உரித்தற்ற வேறோர் உத்தியோகத்தர் அரசாங்க விடுமுறை நாள் ஒன்றிலே பணியாற்றுமிடத்து , அவருக்குப் பெற்றுக்கொள்ள அனுமதி இருப்பது அதற்குப் பதிலாக	பதவிநிலை உத்தியோகத்தர் ஒருவர் , வெளிக்கள உத்தியோகத்தர் ஒருவர் அல்லது மேலதிகநேரப் வகிக்கின்ற உத்தியோகத்தர் ஒருவர் , அரசாங்க விடுமுறை பெறுகின்ற உத்தியோகத்தர்ஒருவ ருக்கு அரசாங்க

		ஒருநாள் பதில் விடுமுறையினை மாத்திரமே ஆகும் .	விடுமுறை நாள் எனப் பொருள்படும் .
2	අධ්‍යාපන අමාත්‍යංශයේ අතිරේක ලේකම්වරුන්	கல்வி அமைச்சின் மேலதிக செயலார்கள்	கல்வி அமைச்சின் மேலதிக செயலாளர்கள் செயலாளர்கள்
3	එම් . පී . බණ්ඩාර , අත්සන් කලේ වරිත හේරත්	எம் . பி . எம் . பீ . பண்டார , கையொப்பமிட்டது சரித ஹேரத்	திரு . எம் . பீ . பண்டார , பத்தரமுள்ளை

Table 7.12: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தேசிய மொழிகள் மற்றும் சமூக ஒருங்கிணைப்பு அமைச்சினால் ஏற்பாடு செய்யப்பட்ட சமூக ஒழுங்கிணைப்பு வாரம் - 2014 முன்னிட்டு இந்த பேச்சுப் போட்டி மேல்மாகாண சபை பாடசாலை மாணவர்களை இலக்காகக் கொண்டு நடாத்தப்பட்டது .	ජාතික භාෂා හා සමාජ ඒකාබද්ධතා අමාත්‍යංශය විසින් සංවිධානය කරන ලද සමාජ ඒකාබද්ධතා සතිය 2014 නිමිත්තෙන් මෙම කථික තරගය බස්නාහිර පළාත් පාසල් සිසුන් ඉලක්ක කරගනිමින් පැවැත්විණි .	ජාතික භාෂා හා සමාජ ඒකාබද්ධතා අමාත්‍යංශය විසින් සංවිධානය කරන ලද සමාජ ඒකාබද්ධතාසම්බන්ධීකාරක - 2014 මාර්තු 05 වන පරිදි පාසල් සිසුන් දැනුවත් කිරීම .
2	II . வெளியக ஒளிபரப்புகளுக்கான ஒளியீட்டல் உபகரணங்களை கொள்வனவு செய்தல் - ரூ . மி . 20 .	ii . බාහිර විකාශන සඳහා ආලෝකකරණ උපකරණ මිළදී ගැනීම - රු . මි . 20 .	ii . බාහිර තොටුපල ව්‍යාපෘතිය සඳහා උපකරණ මිලදී ගැනීම - රු . මි . 20.0 .
3	குறைத்தல் - இது பொருத்தமாக அமைவது , சம்பள ஏற்றத்தை	අඩු කිරීම - මෙය යෝග්‍ය වන්නේ වැටුප් වර්ධකය නතර කිරීමේ තීරණය	අඩු - මෙය යෝග්‍ය වන්නේ වැටුප් වර්ධකය වැටුප් වර්ධකය

	நிறுத்தும் தீர்மானம் ஒரு மாத காலத்திற்குள் நடைமுறைக்கு வராத சந்தர்ப்பங்களிலாகும் .	මාසයක් ඇතුළත දී ක්‍රියාත්මක නොවන අවස්ථාවල දී ය .	තාවකාලිකව නතර කිරීම සඳහා ගණන් ගත යුතුය .

The BLEU score gain that we achieved with respect to our benchmark model is negligible. Investigation of test results revealed following factors as the main reasons that hinder the translation performance.

1. Number of test sentences that were split using our model (70 for Sinhala and 110 for Tamil) was not sufficient to make an acceptable increase in the resulting performance.
2. Several test set splitting caused in producing test sentences that lower the overall fluency of test set.
3. The monolingual corpus that was used to train the language model was not sufficient.
4. There is an associated error when merging the translated sub sentences.

7.6 Transliteration

Table 7.13 depicts an example transliterated sentence.

Table 7.13: Example transliterated sentence

Original Sentence	Transliterated Sentence
හොඳම අධ්‍යාපන සහ සංස්කෘතික වැඩසටහන .	honndhama adhYaapana saha sa\nskrutika wADasaTahana .

The transliterated model showed a BLEU score gain of 1.52 for Tamil to Sinhala. Analysis of the results demonstrated the following points as the reasons for the BLEU score gain.

7.6.1 Loanwords

We identified that the transliterated model could translate loanwords of each language borrowed from the other language, because of the similarity in their transliteration form. e.g: word ‘உபகரணங்கள்’ (‘උපකරණ’ in Sinhala) was translated incorrectly by the baseline model as ‘වගකීම්’ whereas the transliterated model translated it correctly. Here ‘උපකරණ’, ‘உபகரணங்கள்’ are both transliterated into ‘UpakaraNa’. This is a loanword from Sanskrit to Tamil and Sinhala.

There exist a significant number of Sinhala words borrowed from the Tamil vocabulary, but relatively a small number of Tamil words borrowed from the Sinhala vocabulary. However, Sri Lanka has been under the Portuguese, Dutch and English colonial rule from early 16th to late 19th century. Hence there exist a considerable number of loanwords of Portuguese, Dutch and English origin that were borrowed for both languages. (table - මේසය /mesaya/ - மேசை /mesei/), (window - ජනේලය /janelaya/ -

ஜன்னல் /jannal/), (barrel - පීප්පය /peeppaya/ - பீப்பாய் /peeppai/) This vocabulary overlap has contributed to improve the translation quality with transliteration.

7.6.2 Word ordering and long sentences

We identified that the transliterated model can do better translation for long sentences that were poorly translated in the baseline model. We also noted that word ordering is better preserved in the transliterated model. Transliteration model was able to translate whole phrases of words more accurately. This proves that transliteration is useful for languages with close grammatical structures, lexical and syntactic similarities.

7.7 Byte Pair Encoding

Table 7.14 depicts an example byte pair encoded sentence.

Table 7.14: Example Byte Pair Encoded Sentence

Original Sentence	Byte Pair Encoded Sentence
මුද්දරයේ වටිනාකම රුපියල් 50ක් හෝ ඊට වැඩිනම් හෝ එය කැපීමෙන් හෝ විදීමෙන් හෝ අවලංගු කළ යුතු ය .	මුද්දර■ යේ වටිනාකම රුපියල් 50■ ක් හෝ ඊට වැඩිනම් හෝ එය කැපී■ මෙන් හෝ වි■ දීමෙන් හෝ අවලංගු කළ යුතු ය .

Table 7.15 and 7.16 show example translations obtained from this method.

Table 7.15: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	ප්‍රතිශෝදනය කිරීමේ දී නිලධරයෙකු අවසාන වශයෙන් ලැබූ වැටුප නව වැටුප් පරිමාණයේ ආරම්භක වැටුපට අඩු නම් , ඔහු නව වැටුප් පරිමාණයේ ආරම්භක වැටුපෙහි තබනු ඇත .	திருத்தம் செய்யப்படும் போது உத்தியோகத்தர் ஒருவர் இறுதியாக பெற்றுக் கொண்ட சம்பளமானது , புதிய சம்பள அளவுத் திட்டத்தின் ஆரம்பச் சம்பளத்தை விடவும் குறைவாகக் காணப்படின் , அவர் புதிய சம்பள அளவுத் திட்டத்தின் ஆரம்பப் படிமுறையில் வைக்கப்படுவார் .	மலேரியா வாடகைக்கான உத்தியோகத்தர் ஒருவர் உத்தியோகத்தர் ஒருவர் புதிய சம்பளஅளவுத் திட்டத்தின் போது புதிய சம்பள அளவுத் திட்டத்தின் படி , அவர் புதிய சம்பள

			அளவுத்திட்டத்தின் படி மாத்திரமேயாகும் .
2	(අ) යම් අමාත්‍යංශයක සේවය කරන සියලුම රාජ්‍ය නිලධරයන් සඳහා - අදාළ අමාත්‍යංශයේ ලේකම්වරයා හෝ බලය පවරන ලද නිලධරයා .	(அ) யாதேனுமோர் அமைச்சில் பணிபுரிகின்ற - உரிய அமைச்சின் செயலாளர் அனைத்து அரசாங்க உத்தியோகத்தர்களுக்கும் அல்லது அதிகாரமளிக்கப்பட்ட உத்தியோகத்தர் .	(அ) ஏதேனும் அமைச்சின் செயலாளரினால் உள்ள அனைத்து அரச உத்தியோகத்தர்களைஉள்ள அரசாங்க அமைச்சின் செயலாளர் அல்லது அதிகாரமளிக்கப்பட்ட உத்தியோகத்தர் .
3	ප්‍රතිශෝදන කිරීමේ දී නිලධරයෙකු අවසාන වශයෙන් ලැබූ වැටුප නව වැටුප් පරිමාණයේ ආරම්භක වැටුපට අඩු නම් , ඔහු නව වැටුප් පරිමාණයේ ආරම්භක වැටුපෙහි තබනු ඇත .	திருத்தம் செய்யப்படும் போது உத்தியோகத்தர் ஒருவர் இறுதியாக பெற்றுக் கொண்ட சம்பளமானது , புதிய சம்பள அளவுத் திட்டத்தின் ஆரம்பச் சம்பளத்தை விடவும் குறைவாகக் காணப்படின் , அவர் புதிய சம்பள அளவுத் திட்டத்தின் ஆரம்பப் படிமுறையில் வைக்கப்படுவார் .	மலேரியா வாடகைக்கான உத்தியோகத்தர் ஒருவர் உத்தியோகத்தர் ஒருவர் புதிய சம்பளஅளவுத் திட்டத்தின் போது புதிய சம்பள அளவுத் திட்டத்தின் படி , அவர் புதிய சம்பள அளவுத்திட்டத்தி

			ன் படி மாத்திரமேயாகும் .

Table 7.16: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එසේම දෙපාර්තමේන්තු ප්‍රධානියා සම්බන්ධයෙන් නිවාඩු අනුමත කළ ලේකම් විසින් සඳහන් කළ යුතු ය .	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .
2	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එවැනි නිලධරයා සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .
3	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் . இதன் XXX ஆம் அத	එක්කෙනෙකුට වෙනුවෙන් වන්දි ගෙවීම වෙනුවෙන් වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .

7.7.1 Independent BPE Models

Independent BPE models could show a BLEU score gain of 3.44 for Tamil to Sinhala translation. Aim of BPE is to represent an open vocabulary through a compact fixed-size sub word vocabulary, to address the rare word problem. We identified that words that the initial benchmark model could not translate were accurately translated in the BPE model by combining sub-word units.

e.g: ක්‍රියකාරීත්වය - Subwords → ක්‍රි + යා + කාරිත්වය

ක්‍රියාකාරිත්වය, which is formed by stem ‘ක්‍රියා’ + prefix ‘’, is a rare word in the corpus. As Sinhala and Tamil are both morphologically rich languages, there are many morphological forms of a single word, some of which are rare/unseen to the corpus while others are not. BPE model has demonstrated the ability to translate rare/unseen words such as above by combining different sub-word units.

BPE has also shown the ability to translate some of the loanwords similar to the transliteration model. Though Sennrich et al. [32] has demonstrated the potential of BPE model to translate named entities by learning a transliteration mapping based on sub-word units, our model failed to translate named entities accurately. This can be due to the ambiguity between Sinhala and Tamil transliteration as explained previously. Sub-word units of Sinhala and Tamil are different, which also makes it difficult for the model to learn an accurate mapping.

7.7.2 Analysis - Joint BPE Model

Table 7.17 depicts an example byte pair encoded transliterated sentence.

Table 717: Example Byte Pair Encoded Sentence

Original Sentence	Byte Pair Encoded Transliterated Sentence
මුද්දරයේ වටිනාකම රුපියල් 50ක් හෝ ඊට වැඩිනම් හෝ එය කැපීමෙන් හෝ විදීමෙන් හෝ අවලංගු කළ යුතු ය .	mud▪ dar▪ ayei waTinaakama rupiyal 5▪ 0k hoe ieTa wADinam hoe eya kAp▪ iemen hoe wid▪ iemen hoe awala ▪W▪ ng▪ u kaLa yutu ya .

The joint BPE model demonstrated a 3.52 BLEU gain over the initial model and a 0.08 BLEU gain over the independent BPE model. This supports the notion that BPE on the union of the two vocabularies are more effective than separate BPE models.

During our analysis, it is evident that generally many sentences were better translated by the joint model. Words, other than named entities that were not translated by the initial and independent models were translated by the joint model.

As theoretically expected, many of the name entities were correctly translated by the joint BPE model.

e.g: බිනරි ටෙලි නාට්ය and ප්රසන්න ජයසිංහ මහතා were correctly translated by the joint BPE model while it was translated inaccurately as යස ඉසුරු and මනෝජ් ලක්සිරි මහතා by the independent BPE model.

7.8 Parts of Speech Tagging

Use of POS tags has increased NMT performance by 0.03 for Sinhala to Tamil and by 0.09 for Tamil to Sinhala. Being an end to end translation system, NMT does not depend on feature functions. Yet, input features contain important details about the underlying language structure. When used to train the model, it helps underlying deep learning network to learn the grammatical structure more easily. We can empirically prove this claim using the increased fluency in the resulting output, which is demonstrated by the increased BLEU score.

POS tagging is not a static process. Depending on the context, the same word can have multiple POS tags. Hence the performance of the system is highly dependent on the quality of the POS tagger that is used. Since the POS tagger we used, was trained using the same corpus that we used to train our

benchmark model, the performance of the POS tagger can be assumed to be at acceptable level. This is based on our assumption that POS tagger can produce good results for the data that were used to train the POS tagger model.

7.9 SMT NMT Pipelining

This method was not able to produce results that we expected. Main underlying assumption of using a pipeline architecture is that, SMT system performs well when dealing with rare words and unknown words. Since SMT does not limit the vocabulary size when generating the translation model, this behavior is expected.

We identified the following as the underlying reasons for observed poor performance.

1. Quality of translated synthetic Tamil sentences is not in an acceptable level. Hence, when these data are used as the source side of the NMT model, the error gets propagated in the network.
2. The sentence structure of the resulting Tamil synthetic data is not in acceptable levels. This lowers the quality of resulting parallel corpus that is used to train the NMT model.
3. Due to errors in the SMT model, the test set that is used in the NMT model, gets distorted.

7.10 Word Similarity

Table 7.18 shows the similar words for the Sinhala word "මොවුන්", that are generated using the trained model, together with the similarity probability.

Table 7.18: Example Similar words generated by word2vec model

Word	Probability
ඔවුන්	0.805201361646
පුද්ගලයින්	0.663750439497
කාන්තාවන්	0.628971339458

Table 7.19 and 7.20 show example substitutions for both Sinhala and Tamil.

Table 7.19: Example Sinhala rare word and unknown word substitutions

Rare Word / Unknown Word	Substituted Word	Probability
වාතාවරණය	තත්ත්වය	0.768362602199
කුටියේ	කාමරයේ	0.730116656395

සුහදශීලි	විශ්වාසනීය	0.703216500255
තෙවැනි	දෙවැනි	0.921565258898
දෙදෙනෙක්	තිදෙනෙක්	0.942298978789
ලංකාවෙහි	ලංකික	0.90872329952

Table 7.20: Example Tamil rare word and unknown word substitutions

Rare Word / Unknown Word	Substituted Word	Probability
மீளவாங்கும்	தொடர்ச்சியாக	0.847796641009
வெளிநாடொன்றில்	பிரயாணம்	0.81455105451
ஓய்வுபெறச்செய்ய	எடுக்க	0.859584142125
யுகத்தில்	கட்டி	0.962392545926
தளத்திலும்	அணைத்து	0.829801285785
பத்தல	ஜயசேகர	0.838965435492

0.04 BLEU score gain was observed for Sinhala to Tamil and 0.05 for Tamil to Sinhala models. Compared to the improvements that were observed in original research [33], our models have performed poor.

Number of words that were replaced in the Sinhala test set was 399 words, whereas corresponding value for Tamil corpus was found to be 144. Compared to number of words in the test set, for each source and target language, number of words that was substituted is negligible. This leads to poor BLEU score increase.

Another governing factor that hinders the resulting improvement is the quality of similarity model that we used. Quality of similarity model depends on the quality and the size of the monolingual corpora that was used for training. Since the size of the monolingual data we used to train the similarity model is not sufficient, resulting BLEU score gain was minimized.

We observed a BLEU score decrease when POS tags and morphological analysis were used when measuring the word similarity. Observations of results revealed that quality of word substitutions have increased when POS tags and morphological analysis were used. Yet this has resulted in producing less number of substitutions due to increased restrictions in word substituting.

When translating the rare words and unknown words, we made the initial assumption that translations of similar words result in the same translated word. Yet due to morphological richness of the two languages, a slight variation in the source side results in producing different target words. This hinders the translation quality of the model, trained using this method.

7.11 SMT vs NMT Comparison

To compare NMT performance with SMT, we trained a SMT system using the same parallel corpus we used to train each NMT model. We trained only a Sinhala to Tamil SMT system.

Resulting SMT model delivered 17.3 BLEU score. The results obtained were contradicting with the results obtained by Bentivogli et al. [46], which states that NMT performs better than SMT for the same corpus size. For the small dataset we have, SMT performed better than NMT, according to our observations. Number of parameters that need to be learnt in the training process is higher in NMT compared to SMT, which leads to this observation.

7.12 Sentence Length vs BLEU Score

Figure 7.2 and 7.3 show the sentence length vs BLEU score graphs for each model, Sinhala to Tamil and Tamil to Sinhala, obtained for the benchmark model.

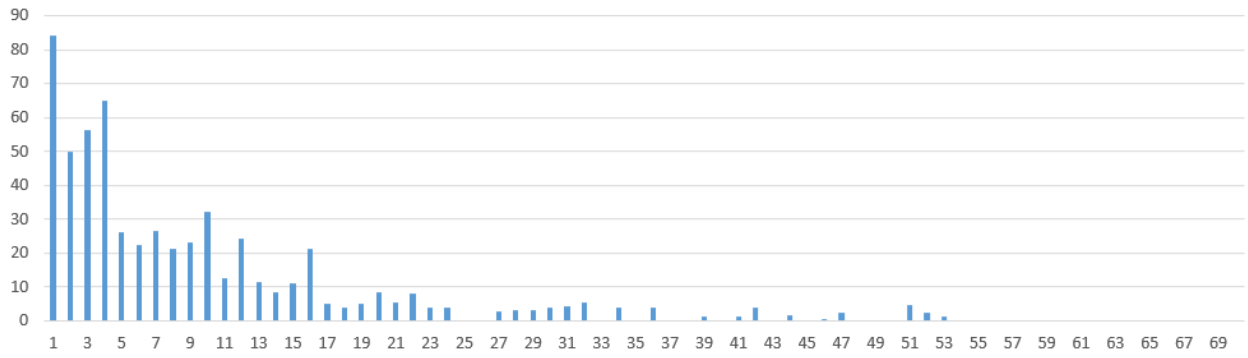

Figure 7.2 : Sinhala to Tamil Sentence length vs BLEU score graph

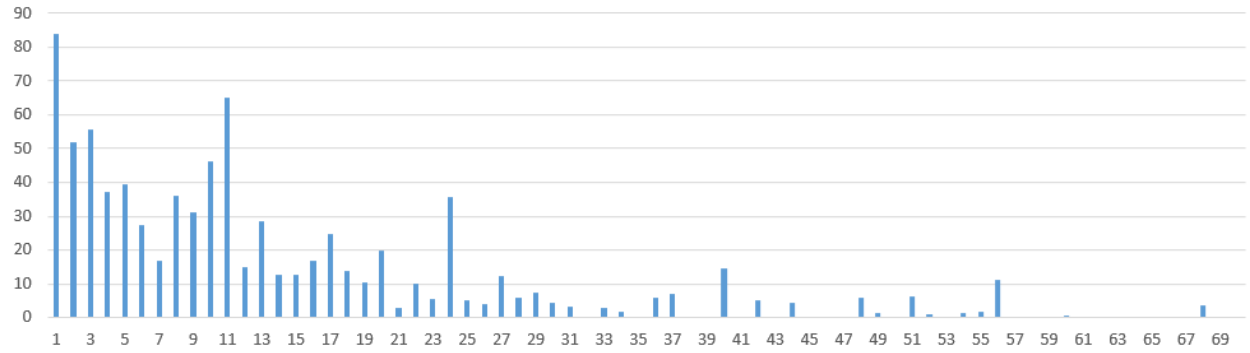

Figure 7.3 : Tamil to Sinhala Sentence length vs BLEU score graph

It can be seen that both models have performed better with shorter sentences than longer sentences. As the sentence length increases, the BLEU score has dramatically decreased for both models.

Attention mechanism was specially proposed to handle this problem of having low performance for longer sentences. It achieves this goal by eliminating the need to compress the whole input sequence in the context vector. Nevertheless, NMT performance has decreased when tested with sentences of length greater than 15 for Sinhala to Tamil machine translation task.

7.13 Sinhala to Tamil vs Tamil to Sinhala Results Comparison

Sinhala to Tamil BLEU score is worse than Tamil to Sinhala BLEU score for each method. For the same parallel corpus, number of words and unique words in Tamil are greater than respective values for Sinhala. Hence when translating from Sinhala to Tamil, out of vocabulary problem is more significant, compared to Tamil to Sinhala. One reason for this mismatch is the morphological difference between two languages.

One such major difference between these two languages that makes the BLEU score for Tamil to Sinhala high is their relative treatment of gender. In Sinhala language both human beings and other animals are treated equally with respect to their gender. In Tamil language, human beings and other animals are treated separately with respect to their gender. Hence when translating gender related terms from Tamil to Sinhala it is a many to one mapping whereas translation from Sinhala to Tamil is one to many mapping. Hence the results for Sinhala to Tamil are worse than Tamil to Sinhala when translating gender related terms.

8 CHALLENGES

8.1 Domain limitations

Due to the limitation of the available data sets, it was not possible to create a model that will cater to the translation needs of all domains – a generalized translation model. Hence our domain was limited to the translation of government documents. It will yield but poor results for out of domain test sets. Another shortcoming in limiting the model to a single domain is the inability to bring forth new data from other domains. It could be stated that including data sets from different domains where the use of written language is different, cannot be used to improve the train set, rather it will result in a possible decrease in translation quality.

8.2 Parallel corpora size

Size of the parallel corpus possess the major bottleneck for the research. Ongoing research which have shown good results use parallel corpus with more than 1 million sentences. The Si-Ta corpus contain only 23611 sentence pairs. NMT systems need large training sets to model correctly. Though SMT models can provide acceptable results for the given corpus size, NMT models require much larger train set to reach that benchmark.

8.3 Constraints on monolingual corpora size

For SMT, a Sinhala monolingual corpus was necessary to build the language model. An in-domain Sinhala corpus was not available (domain – Official document). Hence, we used a Sinhala corpus from a different domain. As it is partially out of domain, it could not contribute to increase the results.

8.4 Memory requirements

The recommended GPU for training is a GPU with a memory of 4GB. Initially, the available GPU for the project was GPU - GeForce GTX 480 with 1.5GB memory. This memory was insufficient to run the OpenNMT system. The GPU training lead to an error in the OpenNMT code. (has a ~1.8GB memory threshold to load all the models and data). Due to this constraint, we were forced to train the model using the CPU. The process took a very long time.

Later, we could get a new GPU - Tesla C2070 5.5 GB, which was sufficient for the research.

8.5 Use of existing tools for Sinhala and Tamil languages

There are many open source tools and repositories that support research in NMT. Most of the benchmark models cited in research papers are open source and available. But we faced a number of problems when trying to adapt them for Sinhala – Tamil translations.

Building a translation model is not an independent task, it requires many pre and post processing steps, such as stemming, pos tagging etc. For European languages such as English and German there exist standard tools which support those steps (e.g: -Stanford parser). Since there are a wide morphological, grammatical differences between European and Asian languages, it was not possible to use some of these available tools and repositories for our research. Furthermore the lack of accuracy of morphological analyser and the POS tagger that we used, caused the overall BLEU score to decrease.

8.6 Standard data sets

There are publicly available standard parallel corpora for many language pairs, auch as WMT for translating between pairs of European language (Bulgarian, English, Czech, German, Spanish, Basque,

Dutch, and Portuguese). There are also large public corpora available for Asian languages such as Japanese (Japanese – English: Tanaka Corpus, ASPEC, TAUS Memory), Chinese etc.

These data sets are the standard data that are used and cited in many research papers. There are also standard test data sets for evaluation in all these corpora. These test sets can be used as standard benchmarks for evaluation and comparison of results obtained by different research. Lack of a standard data set was another limitation in our research.

9 CONCLUSION

The purpose of this research was to implement an NMT system for Sinhala and Tamil language pair for the first time and, improve performance of NMT when corpus size is small. We can conclude that although Tamil to Sinhala and Sinhala to Tamil translations are unable to produce intelligible output with a parallel corpus of just 23611 sentence pairs, we can improve the translation performance by incorporating the following methods.

1. Adding word phrases
2. Incorporating monolingual Data
3. Data augmentation
4. Test set splitting
5. Transliteration
6. Byte pair encoding
7. Part of speech tagging
8. SMT NMT pipelining
9. Word Similarity

Adding word phrases, transliteration are the novel contributions that we have introduced to improve NMT performance for under resourced languages. We improved existing methods of using monolingual target side data, data augmentation, test set splitting, byte pair encoding, pipelining and word similarity to make them applicable for Sinhala and Tamil.

While, adding word phrases, monolingual data and data augmentation methods produced major improvements, other methods were not able to produce BLEU score gains greater than 1.0, which is the accepted threshold in NMT literature.

The purpose of data augmentation was to find out semantic preserving techniques for synthetic data generation to solve the rare word problem in NMT for the under-resourced language pair Sinhala and Tamil. POS tagging and morphological analysis show impressive results in reducing the rare word problem.

We experimentally showed how a joint BPE model (created with transliteration + BPE) improves the translation quality of languages with different alphabets. The results of our experiments showed that, in NMT, transliteration of the entire parallel corpus to a common alphabet is a promising technique that can be used for languages with a vocabulary overlap and grammatical similarities. We developed a transliteration system that can be used in many NLP tasks. Furthermore, we showed how BPE models greatly improve the translation quality by addressing the rare word problem. And more specifically, we showed the contribution of BPE models for low-resource morphologically rich language pairs, to translate different rare/unseen morphological forms of words appearing in the corpus.

We can expect performance to approach usable levels by collecting a large parallel corpus. Using this experience, we are currently collecting a more balanced parallel corpus.

10 FUTURE RESEARCH

Being morphologically rich, there exist a number of morphological features in Sinhala and Tamil that can be exploited to enhance the quality of augmented data. We expect to experiment with these features in the future.

Further improving the word segmentation of morphologically rich languages by incorporating linguistic features in BPE is another possible future research area.

Other than part of speech tagging, there exists many input features that can be used to train the model. One such input feature is morphological case-count combination. In future, we will focus on using them as input features.

A preliminary study shows that it is possible to improve performance for the same dataset we used for this research by treating words at the character level rather than word level [34]. In future, we are planning to investigate on the applicability of this character level NMT approach for Sinhala and Tamil.

We will continue to improve this NMT system to a level that it is capable of producing acceptable translations between Sinhala and Tamil for the use of a wider community.

11 REFERENCES

1. M. Alawneh and T. Sembok, "Rule-Based and Example-Based Machine Translation from English to Arabic", in *Sixth International Conference of Bio-Inspired Computing: Theories and Applications (BIC-TA)*, IEEE, 2011, pp. 343-347.

2. P. Brown, J. Cocke, S. Pietra, V. Pietra, F. Jelinek and J. Lafferty, "A statistical approach to machine translation", *Computational linguistics*, vol. 16, no. 2, pp. 79 - 85, 1990.

3. D. Bahdanau, K. Cho and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate", in *Third International Conference on Learning Representations*, 2015.

4. M. Luong and C. Manning, "Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models", in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 1054-1063.

5. Y. Wu, M. Schuster, Z. Chen, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation", *arXiv preprint arXiv:1609.08144*, 2016.

6. R. Weerasinghe, "A statistical machine translation approach to sinhala-tamil language translation.", *Towards an ICT enabled Society*, 2003 pp. 136-141.

7. K. Wołk and K. Marasek, "Neural-based machine translation for medical text domain. Based on European Medicines Agency leaflet texts", in *Procedia Computer Science*, 2015, pp. 2-9.

8. E. Greenstein and D. Penner, "Japanese-to-English Machine Translation Using Recurrent Neural Networks," Stanford Deep Learning for NLP Course, pp. 1–7, 2015.

9. A. Lopez, "Statistical machine translation", *ACM Computing Surveys*, vol. 40, no. 3, pp. 1-49, 2008.

10. J. Zhang, S. Liu, M. Li, M. Zhou and C. Zong, "Beyond Word-based Language Model in Statistical Machine Translation", *arXiv:1502.01446*, 2015.

11. P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi and B. Cowan, "Moses: Open source toolkit for statistical machine translation", in *Proceedings of the 45th annual meeting of the Association for Computational Linguistics on interactive poster and demonstration sessions*, 2007, pp. 177-180.
12. K. Cho, "Introduction to Neural Machine Translation with GPUs (part 1)", 2015. [Online]. Available: https://devblogs.nvidia.com/parallelforall/introduction-neural-machine-translation-with-gpus/. [Accessed: 11- Nov- 2017].

13. I. Sutskever, O. Vinyals and Q. Le, "Sequence to sequence learning with neural networks", in *Advances in neural information processing systems*, 2014, pp. 3104-3112.

14. K. Cho, "Introduction to Neural Machine Translation with GPUs (part 2)", 2015. [Online]. Available:

15. G. Klein, Y. Kim, Y. Deng, J. Senellart and A. M. Rush, "OpenNMT: Open-source toolkit for neural machine translation", arxiv preprint arXiv:1701.02810 [cs.CL], 2017.
16. "BLEU", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/BLEU. [Accessed: 12- Nov- 2017].

17. "NIST (metric)", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/NIST_(metric). [Accessed: 12- Nov- 2017].

18. "Evaluation of machine translation", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Evaluation_of_machine_translation#Human_evaluation. [Accessed: 12- Nov- 2017].

19. "Part-of-speech tagging", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Part-of-speech_tagging. [Accessed: 12- Nov- 2017].

20. Hettige and A. Karunananda, "A Morphological analyzer to enable English to Sinhala Machine Translation", in *Information and Automation (ICIA), International Conference on*, IEEE, 2006, pp. 21-26.

21. B. Hettige and A. Karunananda, "Computational model of grammar for English to Sinhala machine translation", in *Advances in ICT for Emerging Regions (ICTer), International Conference*, IEEE, 2011, pp. 26-31.

22. M. Jeyakaran, "A novel kernel regression based machine translation system for sinhala-tamil translation.", in *Proceedings of the 4th Annual UCSC Research Symposium*, 2013.

23. A. Silva and R. Weerasinghe, "Example based machine translation for English-Sinhala translations", in *Proceedings of the 09th International IT Conference*, 2008, pp. 27-28.

24. B. Hettige and A. Karunananda, "Theoretical based approach to English to Sinhala machine translation", in *Industrial and Information Systems (ICIIS), International Conference*, IEEE, 2009, pp. 380-385.

25. B. Hettige and A. Karunananda, "Transliteration system for English to Sinhala machine translation", in *Industrial and Information Systems,International Conference*, IEEE, 2017, pp. 209-214.

26. S. Sripirakas, A. Weerasinghe and D. Herath, "Statistical machine translation of systems for Sinhala-Tamil.", in *Advances in ICT for Emerging Regions (ICTer), 2010 International Conference*, IEEE, 2010, pp. 62-68.

27. R. Weerasinghe, "A statistical machine translation approach to sinhala-tamil language translation.", *Towards an ICT enabled Society*, 2003 pp. 136-141.

28. F. Farhath, S. Ranathunga, S. Jayasena, G. Dias and U. Thayasivam, "Improving Domain-Specific Statistical Machine Translation for Sinhala-Tamil using Bilingual Lists." (Unpublished)

29. S. Jean, K. Cho, R. Memisevic and B. Yoshua, "On Using Very Large Target Vocabulary for Neural Machine Translation", in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, 2015, pp. 1-10.

30. H. Mi, Z. Wang and A. Ittycheriah, "Vocabulary manipulation for neural machine translation", in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 124–129.

31. T. Luong, I. Sutskever, Q. Le, O. Vinyals and W. Zaremba, "Addressing the Rare Word Problem in Neural Machine Translation", in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, 2015, pp. 11-19.

32. R. Sennrich, B. Haddow and A. Birch, "Neural Machine Translation of Rare Words with Subword Units", in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 1715-1725.

33. X. Li, J. Zhang and C. Zong, "Towards Zero Unknown Word in Neural Machine Translation", in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2016, pp. 2852-2858.

34. W. Ling, I. Trancoso, C. Dyer and A. Black, "Character-based Neural Machine Translation", *arXiv:1511.04586*, 2015.

35. J. Lee, K. Cho and T. Hofmann, "Fully Character-Level Neural Machine Translation without Explicit Segmentation", *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 365-378, 2017.

36. M. Fadaee, A. Bisazza and C. Monz, "Data Augmentation for Low-Resource Neural Machine Translation", in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2017, pp. 567-573.

37. R. Sennrich, B. Haddow and A. Birch, "Improving Neural Machine Translation Models with Monolingual Data", in *ACL*, 2016,pp. 86-96.

38. C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H. Lin, F. Bougares, H. Schwenk and Y. Bengio, "On Using Monolingual Corpora in Neural Machine Translation", *arXiv:1503.03535*, 2015.

39. R. Sennrich and B. Haddow, "Linguistic Input Features Improve Neural Machine Translation", in *Proceedings of the First Conference on Machine Translation*, 2016, pp. 83-91.

40. F. Och and H. Ney, "The Alignment Template Approach to Statistical Machine Translation", *Computational Linguistics*, vol. 30, no. 4, pp. 417-449, 2004.

41. R. Priyanga, S. Ranatunga and G. Dias, "An Inflectional Morphological Generator for Sinhala Nouns." (Unpublished)

42. "Tilde MT", *Letsmt.eu*, 2017. [Online]. Available: https://www.letsmt.eu/Bleu.aspx. [Accessed: 12- Nov- 2017].

43. S. Fernando, S. Ranathunga, S. Jayasena and G. Dias, "Comprehensive Part-Of- Speech Tag Set and SVM based POS Tagger for Sinhala", in *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing*, 2016, pp. 173-182.

44. A. info@url.cz, "Computational Linguistic Research Group", *Au-kbc.org*, 2017. [Online]. Available: http://www.au-kbc.org/nlp/corpusrelease.html. [Accessed: 12- Nov- 2017].

45. "Helabasa noun analyser", 2017. [Online]. Available: http://translation.projects.mrt.ac.lk:8081/helabasa/noun _analyzer. [Accessed: 12- Nov- 2017]

46. L. Bentivogli, A. Bisazza, M. Cettolo and M. Federico, "Neural versus Phrase-Based Machine Translation Quality: a Case Study", in Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, Texas, USA, 2016,pp 257-267

APPENDIX A: Benchmark Results

Table 31: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	ශ්‍රී ලංකා රූපවාහිනී සංස්ථාව	இலங்கை ரூபவாஹினிக் கூட்டுத்தாபனம்	இலங்கை ரூபவாஹினிக் கூட்டுத்தாபனம்
2	අධ්‍යාපන අමාත්‍යංශයේ අතිරේක ලේකම්වරුන්	கல்வி அமைச்சின் மேலதிக செயலார்கள்	கல்வி அமைச்சின் செயலாளர் செயலாளர்கள்
3	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி அமைச்சுடன் பட்டப் பாடநெறி (ஆங்கில தர கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)

Table 32: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .	සිවිල් හානියට පත් කිරීම් පිළිබඳ වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා	එසේම දෙපාර්තමේන්තු ප්‍රධානියා දෙපාර්තමේන්තු

	உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	අනුමත කළ යුතු ය .	ප්‍රධානියා වෙත යැවිය යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තිය සංශෝධනය කර ඇත .

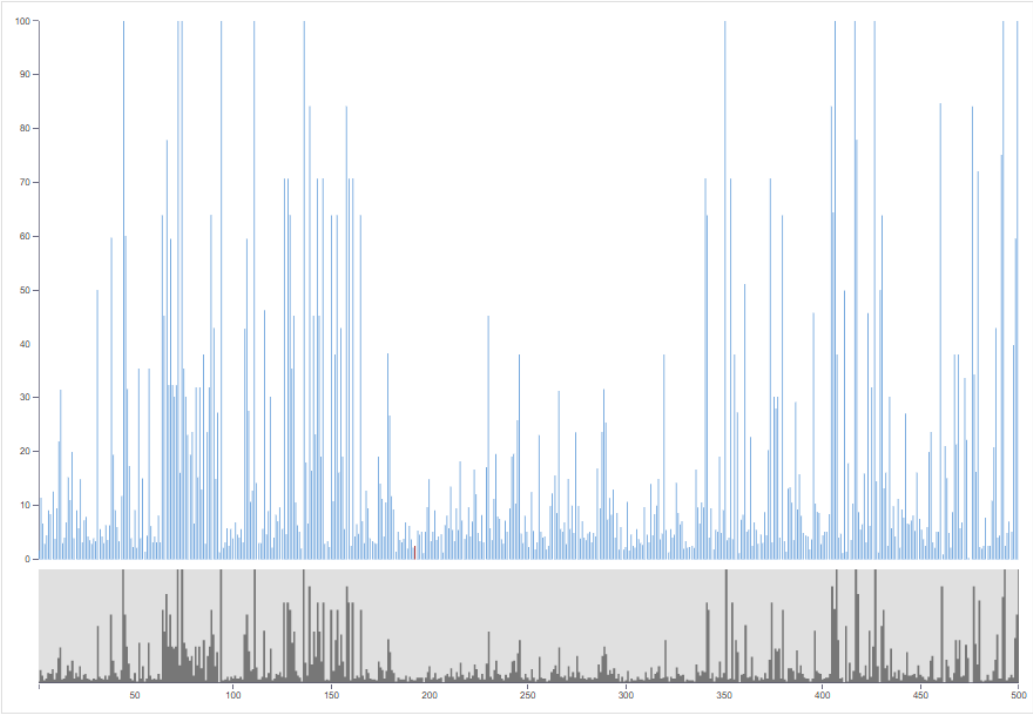

Figure 30: Sinhala to Tamil BLEU score graph

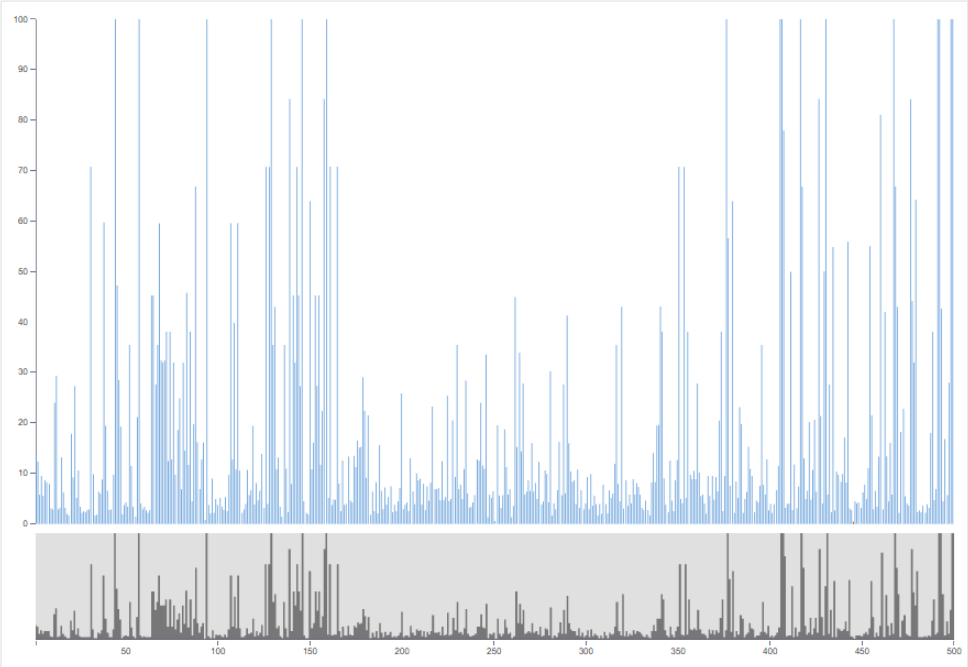

Figure 31: Tamil to Sinhala BLEU score graph

APPENDIX B: Adding Word Phrases Results

Table 33: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	අධ්‍යාපන අමාත්‍යංශයේ අතිරේක ලේකම්වරුන්	கல்வி அமைச்சின் மேலதிக செயலார்கள்	கல்வி அமைச்சின் மேலதிக செயலாளர்கள்
2	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)
3	රාජ්‍ය භාෂා කොමසාරිස් .	அரசகரும மொழிகள் ஆணையாளர் .	அரசகரும மொழிகள் ஆணையாளர் .

Table 34: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .	සමෘද්ධි වන්දි ගෙවීම සඳහා කටයුතු සඳහා වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා	එසේම දෙපාර්තමේන්තු ප්‍රධානියා විසින් නිවාඩු

	உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	අනුමත කළ යුතු ය .	අනුමත කළ යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වැනි වගන්තියේ විධිවිධාන අදාළ වේ .

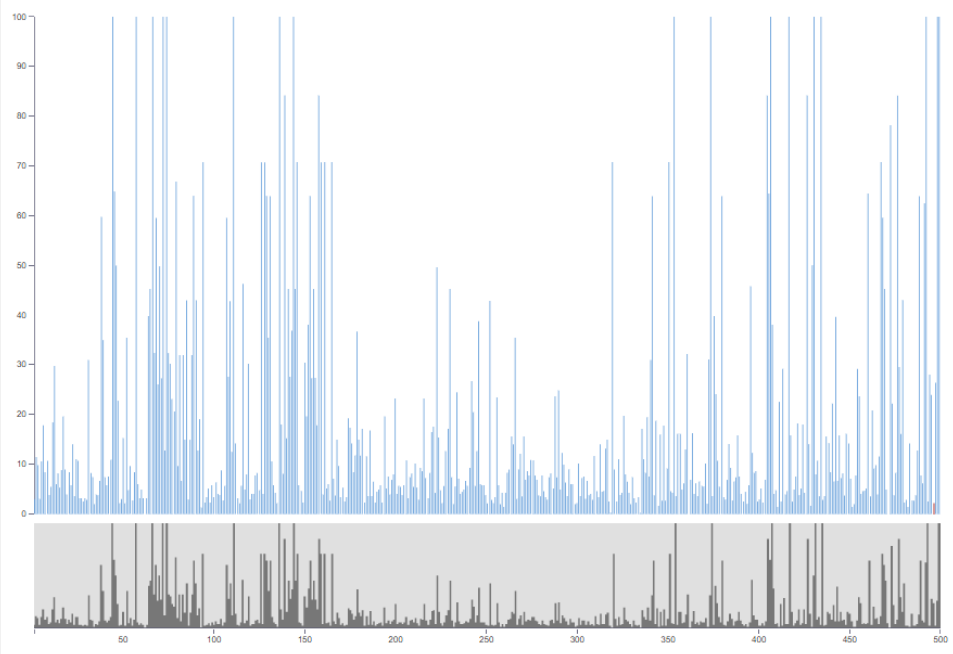

Figure 32: Sinhala to Tamil BLEU score graph

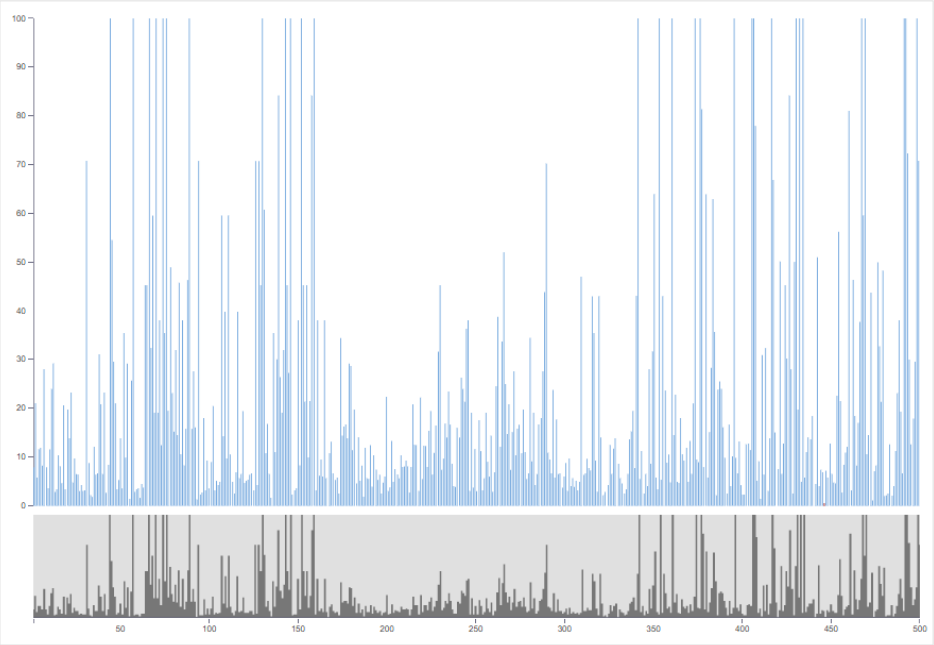

Figure 33: Tamil to Sinhala BLEU score graph

APPENDIX C: Monolingual Training Data Results

Table 35: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	අමාත්‍යංශය සහ අනුබද්ධ ආයතන වල නිලධාරීන් 30 ක් පමණ සහභාගි විය .	அமைச்சு மற்றும் அதன் உள்ளக நிறுவனங்களின் உத்தியோகத்தர்கள் 30 பேர் கலந்து கொண்டனர் .	அமைச்சு மற்றும் நீரியல் நிறுவனங்கள் தொடர்பான உத்தியோகத்தர்கள் பேர் கலந்துகொண்டனர் .
2	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் 1 ஆம் பிரிவின் பார்க்கவும் .
3	අධ්‍යාපන තාක්ෂණවේදී උපාධි පාඨමාලාව (ඉංග්‍රීසි භාෂාව ඉගැන්වීම) (වෘත්තීය තාක්ෂණ විශ්වවිද්‍යාලය)	கல்வி தொழில்நுட்பவியல் பட்டப் பாடநெறி (ஆங்கில மொழி கற்பித்தல்) (தொழில்நுட்பவியல் பல்கலைக்கழகம்)	கல்வி மேன்முறையீட்டு பட்டப் பாடநெறி (ஆங்கில தர கற்பித்தல்) (தொழில்சார்தொழில்நுட்பவியல் பல்கலைக்கழகம்)

Table 36: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු .	කම්කරුවන්ට වන්දි ගෙවීම පිළිබඳ කටයුතු සඳහා වෛද්ය නිලධරයන්ට ගෙවන ගාස්තු .
2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එලෙසම දෙපාර්තමේන්තු ප්රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .	එසේම දෙපාර්තමේන්තු ප්රධානියෙකු ද , ඔහුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එවැනි නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ සඳහන් පරිද්දෙනි .

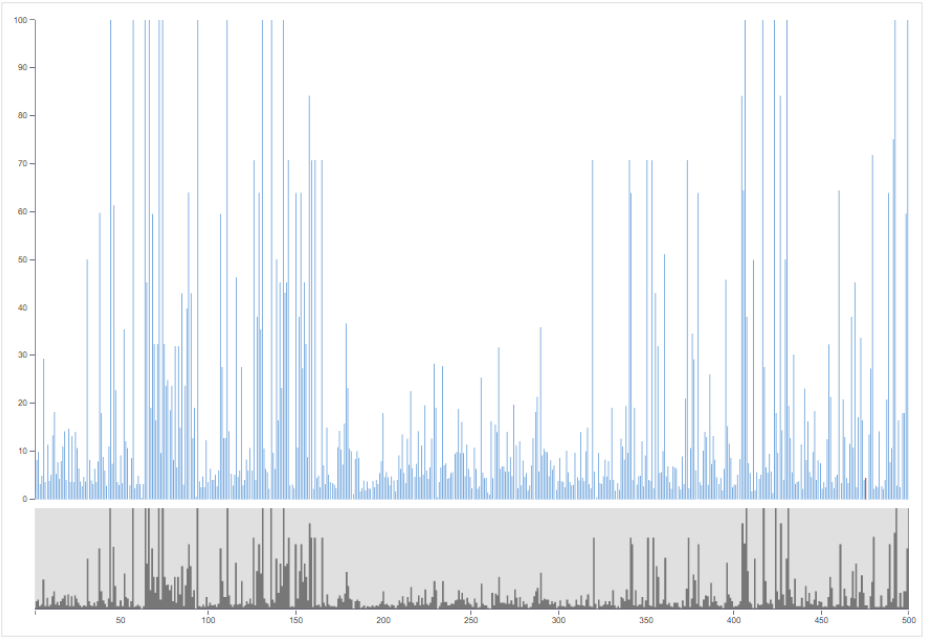

Figure 34: Sinhala to Tamil BLEU score graph

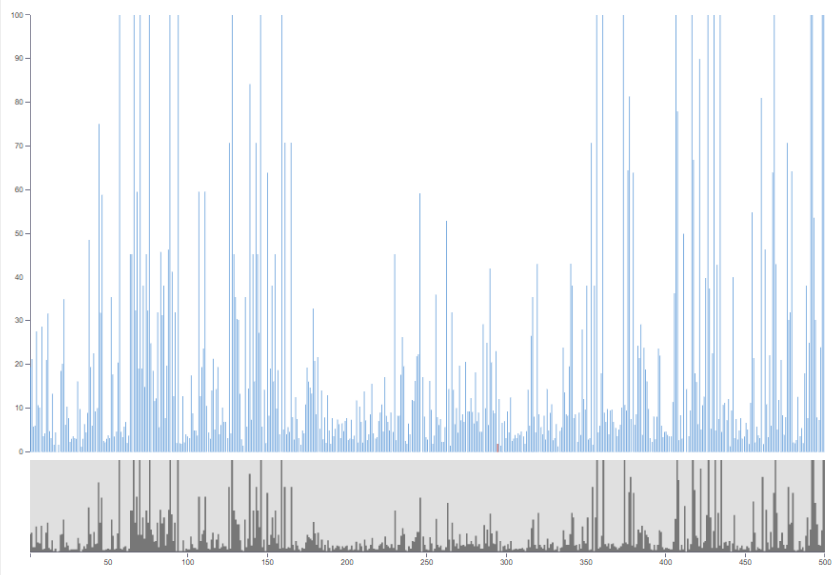

Figure 35: Tamil to Sinhala BLEU score graph

APPENDIX D: Data Augmentation Results

Table 37: Example Sinhala to Tamil Translations

	Source Sentence	Reference Translation	Machine Translation
1	අමාත්‍යංශය සහ අනුබද්ධ ආයතන වල නිලධාරීන් 30 ක් පමණ සහභාගි විය .	அமைச்சு மற்றும் அதன் உள்ளக நிறுவனங்களின் உத்தியோகத்தர்கள் 30 பேர் கலந்து கொண்டனர் .	அமைச்சு மற்றும் அதன் உள்ள உத்தியோகத்தர்கள் பேர் கலந்து கொண்டனர் .
2	දෙපාර්තමේන්තු ප්‍රධානියෙකු පිළිබඳ රහස්‍ය වාර්තාව පිළියෙළ කළ යුත්තේ අදාළ අමාත්‍යාංශයේ ලේකම් විසිනි .	திணைக்களத் தலைவர் ஒருவர் தொடர்பான அந்தரங்க அறிக்கை செயலாளரினாலேயே தயாரிக்கப்படல் வேண்டும் .	திணைக்களத் தலைவர் தொடர்பான அந்தரங்க அறிக்கையை தயாரித்தல் தொடர்பானதீர்மானம் .
3	ඇප වාර්තා වෙනුවෙන් අයකරන ගාස්තු .	பிணை அறிக்கைகளுக்காக அறவிடப்படும் கட்டணங்கள் .	பிணை அறிக்கைகளுக்காக அறவிடப்படும் கட்டணங்கள் .

Table 38: Example Tamil to Sinhala Translations

	Source Sentence	Reference Translation	Machine Translation
1	தொழிலாளர்களுக்கு நட்டஈடு வழங்குதல் தொடர்பான பணிகளுக்காக மருத்துவ அதிகாரிகளுக்குச் செலுத்தப்படும் கட்டணங்கள் .	කම්කරුවන්ට වන්දි ගෙවීම සම්බන්ධ කටයුතු වෙනුවෙන් වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .	නගරය වෙනුවෙන් වන්දි ගෙවීම පිළිබඳ කටයුතු සඳහා වෛද්‍ය නිලධරයන්ට ගෙවන ගාස්තු .

2	அவ்வாறே திணைக்களத் தலைவர் ஒருவரின் லீவினை உரிய செயலாளர் அனுமதித்தல் வேண்டும் .	එලෙසම දෙපාර්තමේන්තු ප්‍රධානියෙකුගේ නිවාඩු අදාළ ලේකම්වරයා අනුමත කළ යුතු ය .	එසේම දෙපාර්තමේන්තු ප්‍රධානියා නිවාඩු අනුමත කළ යුතු ය .
3	அத்தகைய உத்தியோகத்தர் ஒருவர் தொடர்பில் இதன் 1 ஆம் பிரிவின் ஏற்பாடுகள் ஏற்புடையனவாகும் .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ විධිවිධාන අදාළ වේ .	එබඳු නිලධරයෙකු සම්බන්ධයෙන් මෙහි 1 වගන්තියේ සඳහන් පරිදි වේ .

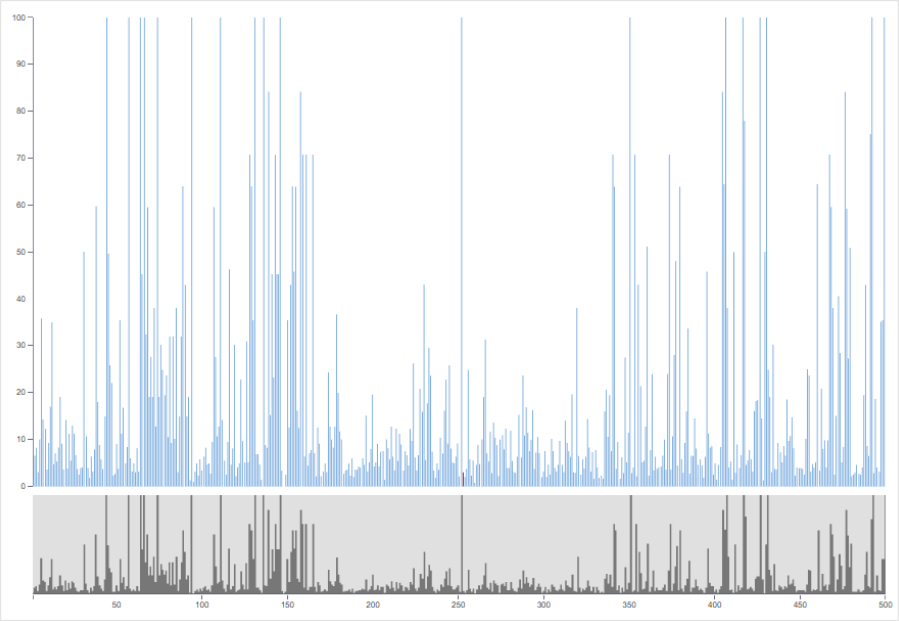

Figure 36: Sinhala to Tamil BLEU score graph

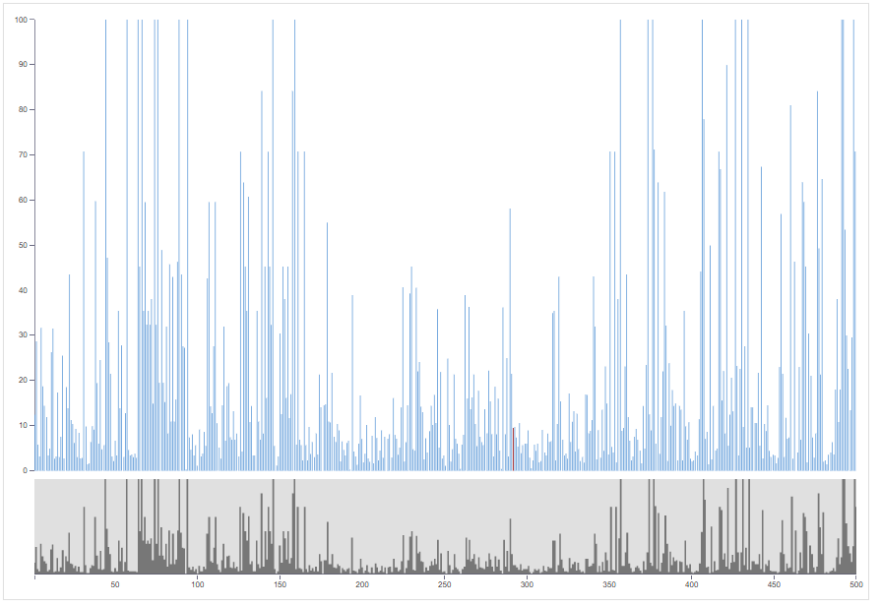

Figure 37: Tamil to Sinhala BLEU score graph

APPENDIX E: Neural Machine Translation for Sinhala and Tamil Languages research paper

Neural Machine Translation for Sinhala and Tamil Languages

Pasindu Tennage, Prabath Sandaruwan, Malith Thilakarathne, Achini Herath, Surangika Ranathunga, Sanath Jayasena, Gihan Dias

Department of Computer Science and Engineering, University of Moratuwa
Katubedda 10400, Sri Lanka
{pasindu.13, prabath.sandaruwan.13, malith.13, narmada.ah.13, surangika, sanath, gihan}@cse.mrt.ac.lk

***Abstract*—Neural Machine Translation (NMT) is becoming the current state of the art machine translation technique. Although NMT is successful for resourceful languages, its applicability in low-resource settings is still debatable. In this paper, we address the task of developing a NMT system for the most widely used language pair in Sri Lanka- Sinhala and Tamil, focusing on the domain of official government documents. We explore the ways of improving NMT using word phrases in a situation where the size of the parallel corpus is considerably small, and empirically show that the resulting models improve our benchmark domain specific Sinhala to Tamil and Tamil to Sinhala translation models by 0.68 and 5.4 BLEU, respectively. The paper also presents an analysis on how NMT performance varies with the amount of word phrases, in order to investigate the effects of word phrases in domain specific NMT.**

Keywords- Neural Machine Translation (NMT); Word Phrases

1 Introduction

Neural Machine Translation (NMT) is a new architecture that aims at building a single neural network that can be jointly tuned to maximize translation performance. NMT delivers state of the art results, especially for language pairs involving rich morphology prediction and significant word reordering. NMT generates outputs that have lower post-edit effort with respect to Statistical Machine Translation (SMT) outputs. NMT seems to have an edge especially on lexically rich texts [1]. However, performance of NMT degrades rapidly as the parallel corpus size gets small.

Sinhala and Tamil languages are under resourced due to lack of sufficiently large parallel corpora. Existing translation models for this language pair have not yet reached the stage of proliferation demonstrated by translation systems for other languages [2]. Thus, the use of NMT for Sinhala and Tamil is challenging.

In this research, we developed a domain specific NMT system for language pair - Sinhala and Tamil, the official languages in Sri Lanka. Official government document translation was focused in this research. In this research, we used the NMT architecture proposed by Bahdanau et al. [3] and Cho et al. [4] for all the experiments.

Given that word phrases play an important role in domain specific machine translation, we explore effective methods of using word phrases to improve NMT performance for these two under resourced languages. We discuss how NMT performance varies with the amount of word phrases added and the effect of word phrases in domain specific NMT. We also use an existing method of using monolingual target side data to improve domain specific NMT performance [5].

Finally, the paper highlights the effect of sentence length for translation performance for Sinhala and Tamil NMT.

2 Background and Related Work

11.1 Neural Machine Translation

NMT is an end-to-end translation process [3], which does not rely on pre-designed feature functions. The goal of NMT is to design a model of which every component is tuned based on training corpora to maximize its translation performance. Encoder Decoder architecture with attention mechanism is the current state of the art NMT architecture.

Recurrent activation function is applied recursively over the input sentence, until the end when the h_T which is the final internal state of the recurrent neural network (RNN) contains the summary of the whole input sequence. After the last word's continuous vector s_T (T is input sequence length) is read, the RNN's internal state h_T represents a summary of the whole source sequence. Decoder computes RNN's internal state z_i based on the summary vector h_T, the previous predicted word u_{i-1} and the previous internal state z_{i-1}. Using decoder's internal hidden state z_i, it's possible to score each target word based on how likely it is to follow all the preceding translated words. Once the score of every

word is computed, using softmax normalization, scores are turned into proper probabilities.

There have been multiple efforts to improve performance of NMT. In recent research, data augmentation to increase NMT performance for under resourced languages has been explored [6]. In this work, synthetic parallel sentences have been generated to increase the number of rare word occurrences. Using target side monolingual data to improve NMT performance for general under resourced languages has also been proposed [5]. According to this research, using synthetic source side sentences generated from back translation has increased the quality of translation by a significant amount.

A method to translate phrases in NMT by integrating a phrase memory into the encoder-decoder architecture of NMT has been presented by Wang et al. [7]. In this model, at each decoding step, the phrase memory is first rewritten by the SMT model, which dynamically generates relevant target phrases with contextual information provided by the NMT model.

11.2 Sinhala - Tamil machine translation

Sinhala language descends from Indic language family and Tamil from Dravidian family [8]. Being morphologically rich, Sinhala has up to 110 noun word forms and up to 282 verb word forms [9] and Tamil has around 40 noun word forms and up to 240 verb word forms [10]. Both these languages have the same word order of Subject-Object-Verb. However, both languages have the flexibility to alter the word order.

There is no published literature on applying NMT for Sinhala and Tamil machine translation. However, some research has been carried out on Sinhala-Tamil SMT [11]. Sinhala-Tamil language pair gives better performance compared to the Sinhala-English pair in SMT due to similarities between Sinhala and Tamil [11].

Recently a research has been carried out on development of a Sinhala-to-Tamil SMT system for official government documents "unpublished" [12]. This system has been developed with emphasis given to domain adaptation. Performance of the system has been evaluated with the static integration of three types of lists, namely, a list of government organizations and official designations, a glossary related to government administrations and operations, and a general bilingual dictionary to the translation model of the SMT system.

3 Methodology

In this research, we identified a novel approach of using word phrases to improve domain specific NMT performance. We also empirically tested the applicability of using target side monolingual training data to improve the performance of Sinhala to Tamil NMT, as done by Sennrich et al. [5].

11.3 Including Word phrases

We consider a word phrase as a combination of 1 or more words that has a specific meaning when taken together. Maximum word phrase size was set to 3 words. Several types of word phrases that are in domain with this translation task were extracted and added to the training corpus. These include a set of named entities, a set of common domain specific terms and phrases, a set of government designations and frequently used phrases that are used in government documents. These word phrases were integrated statically into the NMT system.

11.4 Monolingual Training Data

Monolingual data are especially helpful if parallel data are sparse, or if there is a poor fit for the translation task, for instance because of a domain mismatch. Techniques that can be used to improve the quality of NMT using monolingual data have been identified by Sennrich et al. [5]. According to the authors, adding synthetic target side monolingual data where the source side data are generated using automatic back translation produces better results compared to using dummy source sentences. Hence, we used the synthetic source sentence method to increase the performance of NMT. Back translation of target side monolingual data was done using our system itself.

4 Experimental Setup

11.5 Data

The domain of this translation task is official government documents of Sri Lanka. We used the parallel corpus developed by Farhath et al. "unpublished" [12]. Parallel corpus features government documents such as annual reports, establishment codes, order papers, and official letters. The extracted parallel data have been manually cleaned with the help of a custom developed tool. Human translators oversaw the process of extracting data from the above-mentioned documents and ensured the validity of the translation materials used in the data set. Statistics of the Sinhala-Tamil parallel dataset are shown in Table I.

Parallel corpus was divided into 3 parts: training set, validation set, and testing set. Each dataset consisted of parallel source and target data containing one sentence per line with tokens separated by a space. Validation files were used to evaluate the convergence of the training. To make an unbiased test data set, it was necessary to take the relevant ratios of sentence pairs from different sources.

11.6 Pre-Processing

Both Sinhala and Tamil languages contain one or more symbols per character, unlike English. Due to this characteristic of Sinhala and Tamil, existing tokenization tools were not able to tokenize the text, since they identified a single character as two characters. Hence a tokenizer that was specifically developed for Sinhala and Tamil was used in this research.

11.7 System Setup

The open source NMT system OpenNMT [13] was used for the experiments. OpenNMT supports standard encoder - decoder architecture with attention mechanism.

To evaluate the quality of the translation, Bilingual Evaluation Understudy (BLEU) metric [14] was used. We used the percentage BLEU score values in this paper (0 to 100 range).

11.8 Benchmark Training

Using the above parallel corpus, two benchmark systems were trained: Sinhala to Tamil, and Tamil to Sinhala. Training involved two different steps: pre-processing and model training. After completing the pre-processing step, two dictionaries (source dictionary and target dictionary) were generated to index mappings. Using these two dictionaries and the serialized file, a model was trained with 2-layer Long Short-Term Memory with 500 hidden units on both encoder and decoder. Since most of the operations inside the network were numeric and easily parallelizable, NVIDIA TESLA C2070 with GPU memory 5.5 GB was used to speed up the process.

11.9 Including Word Phrases

Four types of word phrases that are in domain with this translation task were extracted and added to the training corpus.

TABLE I. CHARACTERISTICS OF THE PARALLEL DATASET

Language	Total Words	Unique Words	Sentences
Sinhala	346030	19531	23611
Tamil	293821	37243	

- Set of named entities-11,561 pairs
- Set of common domain specific terms and phrases- 19,861 pairs
- Set of government designations- 5,291 pairs
- Frequently used phrases that are used in government documents (Letter heads, salutations etc.) - 610 pairs

To find the effect of number of word phrases for the BLEU score, a comprehensive analysis was carried out. We trained separate models for Sinhala to Tamil, and Tamil to Sinhala by adding 5000 more-word phrases to the initial training dataset each time. Experiments were carried out for 5k, 10k, 15k, 20k, 25k, 30k, 35k, 40k, 45k and 47k number of word phrases.

11.10 Monolingual Training Data

Target side in-domain monolingual data were extracted using official letters. Maximum sentence length was set to 30 to avoid the performance issues in NMT systems when sentence length is large. 10, 000 target side parallel sentences for each language were extracted and back translated. Model that was trained using the extended corpus that included word phrases was used to back translate. Since the generated synthetic source side data had translation errors, we restricted number of synthetic sentence pairs to be less than the number of sentence pairs in the original parallel corpus, to make sure that the overall quality of resulting parallel corpus remains acceptable. Two models were trained for Sinhala to Tamil and Tamil to Sinhala separately.

Another analysis was carried out to identify how BLEU score is affected by sentence length for Sinhala Tamil NMT. In this study, we calculated average BLEU score for each group of sentences with a particular length.

5 Results and analysis

Table II depicts the BLEU scores obtained for each method.

Compared to results achieved in general machine translation tasks [2, 8], the results we achieved were

significantly higher with respect to the dataset size we used. Major reason for this mismatch is the domain-specific nature of the dataset we used. Since the vocabulary and language constructs are smaller in our dataset, compared to datasets in general machine translation tasks, the model is fine tuned for the domain. Hence the BLEU scores are high.

The results obtained were contradicting with the results obtained by Bentivogli et al. [1], which states that NMT performs better than SMT for the same corpus size. For the small dataset we have, SMT performed better than NMT, according to our observations. Number of parameters that need to be learnt in the training process is higher in NMT compared to SMT, which leads to this observation.

Adding word phrases has increased the BLEU score by 0.68 for Sinhala to Tamil translation, and by 5.4 for Tamil to Sinhala translation. This BLEU score gain is due to two main factors. Firstly, adding word phrases increases the corpus size. It should be noted that the word phrases that were added are not complete sentences, but contain only 2-3 words per phrase.

Second reason for the BLEU score gain is the nature of domain-specific language translation behavior. In this research, we used official government documents as our domain. The word phrases included a significant amount of named entities that are widely used in official government documents. Even though there is no explicit language model in NMT, the decoder considers the last translated word when assigning the probability to the next translated word. There is a high chance that a named entity appears only once in the original training corpus. Hence, a low probability would be applied for the correct next word, due to low presence of two adjacent words in the corpus. Adding word phrases helps to increase this probability, thus reducing rare word problem.

Fig. 1 shows the graph that depicts BLEU score against the number of word phrases.

When the number of word phrases is increased by 5000, the increase in BLEU score is 0.0723 BLEU points in average for Sinhala to Tamil translation and 0.5744 BLEU points for Tamil to Sinhala translation.

Use of target side monolingual data improved the translation quality by 0.13 for Sinhala to Tamil and 3.43 for Tamil to Sinhala. A central theoretical expectation is that monolingual target-side data improve the model's fluency, and its ability to produce natural target-language sentences. This BLEU score gain is due to two factors. Target side monolingual data play a vital role in language modeling in SMT. Being an end to end process, NMT does not have a separate language model. Yet in the decoder, NMT system considers the previously translated

word when predicting the new translation. Hence when in-domain target side monolingual data are added to the training corpus by automatically back translating them to source side, NMT system can take advantage of language specific features in the target side. Second major reason for BLEU score gain is the increased corpus size. Even though the quality of back translated source side is low,

TABLE II. BLEU SCORES

Method	**Sinhala-Tamil**	**Tamil-Sinhala**
Benchmark system	6.78	6.84
+Adding word phrases	7.46	12.24
+Monolingual training data	6.91	10.27
+Adding word phrases +Monolingual training data	7.50	12.75
SMT [12]	17.06	Not Trained

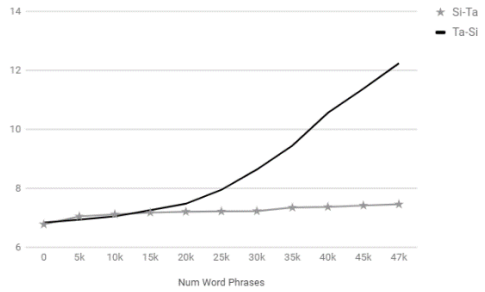

Figure 1. Number of word phrases vs BLEU score

compared to our original parallel corpus, the model output was increased due to the increased number of sentences in the parallel corpus.

According to Table II, Sinhala to Tamil BLEU score is worse than Tamil to Sinhala BLEU score for every method. In Table I, for the same parallel corpus, number of words and unique words in Tamil are greater than respective values for Sinhala. Hence when translating from Sinhala to Tamil, out of vocabulary problem is more significant, compared to Tamil to Sinhala.

Another reason for this mismatch could be the morphological differences between the two languages, such as relative treatment of gender. In Sinhala language, both human beings and other animals are treated equally with respect to their gender. In Tamil language, human beings and other animals are treated separately with respect to their gender. Hence when translating gender related terms from Tamil to Sinhala, it is a many to one mapping whereas translation from Sinhala to Tamil is one to many mapping.

Fig. 2 depicts the relationship between sentence length and BLEU score for both Sinhala to Tamil, and Tamil to Sinhala translations, respectively.

Both models have performed better with shorter sentences than longer sentences. As the sentence length increases, the BLEU score has dramatically decreased for both models.

6 Conclusion

The purpose of this research was to improve performance of NMT when corpus size is small. We can conclude that while Tamil to Sinhala and Sinhala to Tamil translations are unable to produce intelligible output with a parallel corpus of just 23611 sentence pairs, we can improve the translation performance by adding word phrases and using monolingual training data. We can expect performance to approach usable levels by collecting a large parallel corpus. Using this experience, we are currently collecting a more balanced parallel corpus.

Morphological richness in the two languages is one of the major reasons to get lower results. Furthermore, a preliminary study shows that it is possible to improve

performance for the same dataset we used for this research by treating words at the character level rather than word level [15]. In future, we are planning to investigate on the applicability of this character level NMT approach for

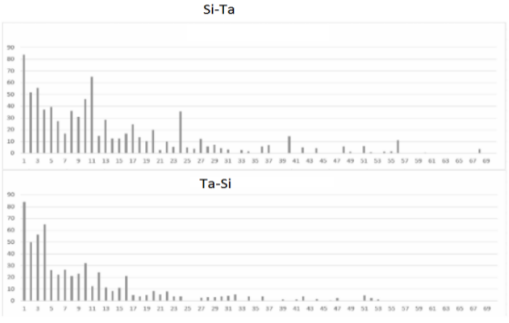

Figure 2. Sentence length vs BLEU score

Sinhala and Tamil. We will continue to improve this NMT system to a level that it is capable of producing acceptable translations between Sinhala and Tamil for use by the wider community.

This work paves way for new research topics related to word phrases. Applicability of word phrases for big data applications is a possible future work. Domain adaptation of machine translation systems using in domain word phrases is another possible future research topic.

7 Acknowledgment

The authors would like to thank the reviewers for their helpful comments and suggestions. The authors are grateful to members of the National Languages Processing Centre at University of Moratuwa for their significant contribution in developing the basic linguistic resources, and the Department of Official Languages of Sri Lanka for providing corpus data needed to carry out the research.

8 References

[1] L. Bentivogli, A. Bisazza, M. Cettolo and M. Federico, "Neural versus Phrase-Based Machine Translation Quality: a Case Study", in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Austin, Texas, USA, 2016, pp 257-267.

[2] R. Sennrich, B. Haddow and A. Birch, "Edinburgh Neural Machine Translation Systems for WMT 16", in *Proceedings of the First Conference on Machine Translation (WMT)*, Association for Computational Linguistics, 2016, pp. 371-376.

[3] D. Bahdanau, K. Cho and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate", *arxiv preprint arXiv:1409.0473 [cs.CL]*, 2014.

[4] K. Cho, B. Merrienboer, D. Bahdanau and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches", in *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, 2014, pp. 103-111.

[5] R. Sennrich, B. Haddow and A. Birch, "Improving Neural Machine Translation Models with Monolingual Data", in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*,ACL, 2016,pp. 86-96.

[6] M. Fadaee, A. Bisazza and C. Monz, "Data Augmentation for Low-Resource Neural Machine Translation.", in *Proceedings of the Annual Meeting of the Association for Computational Linguistics* ,ACL , 2017, pp. 567-573.

[7] X. Wang, Z. Tu, D. Xiong and M. Zhang, "Translating Phrases in Neural Machine Translation", in *Conference on Empirical Methods in Natural Language Processing*, 2017,pp. 1432-1442..

[8] P. Randil, R. Weerasinghe and M. Niranjan, "Sinhala-Tamil Machine Translation: Towards better Translation Quality", *Australasian Language Technology Association Workshop 2014*, vol. 129, pp. 129-133.

[9] V. Welgama, D. Herath, C. Liyanage, N. Udalamatta, R. Weerasinghe and T. Jayawardana, "Towards a Sinhala Wordnet", in *Conference on Human Language Technology for Development*, Alexandria, Egypt, 2011, pp. 39-43.

[10] S. Lushanthan, A. Weerasinghe and D. Herath, "Morphological analyzer and generator for tamil language", in *IEEE conference on Advances in ICT for Emerging Regions (ICTer), ,* 2014, pp. 190-196.

[11] R. Weerasinghe, "A statistical machine translation approach to sinhala-tamil language translation.", *Towards an ICT enabled Society*, 2003 pp. 136-141.

[12] F. Farhath, "Sinhala-to-Tamil Machine Translation of Short Official Documents".

[13] G. Klein, Y. Kim, Y. Deng, J. Senellart and A. M. Rush, "OpenNMT: Open-source toolkit for neural machine translation", *arxiv preprint arXiv:1701.02810 [cs.CL]*, 2017.

[14] K. Papineni, S. Roukos, T. Ward and W. Zhu, "BLEU: a method for automatic evaluation of machine translation.", in *Proceedings of the 40th annual meeting on association for computational linguistics*, Association for Computational Linguistics, 2002, pp. 311-318.

[15] Chung, K. Cho and Y. Bengio, "A Character-Level Decoder without Explicit Segmentation for Neural Machine Translation", in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*,ACL 2016, pp. 16

APPENDIX F: Handling Rare Word Problem using Synthetic Training Data for Sinhala and Tamil Neural Machine Translation research paper

Handling Rare Word Problem using Synthetic Training Data for Sinhala and Tamil Neural Machine Translation

Pasindu Tennage, Prabath Sandaruwan, Malith Thilakarathne, Achini Herath, Surangika Ranathunga
Department of Computer Science and Engineering
University of Moratuwa
Katubedda 10400, Sri Lanka
{pasindu.13, prabath.sandaruwan.13, malith.13, narmada.ah.13,surangika,}@cse.mrt.ac.lk

Abstract
Lack of parallel training data influences the rare word problem, which limits the performance of Neural Machine Translation (NMT) systems, particularly for under-resourced languages. Using synthetic parallel training data (data augmentation) is a promising approach to handle the rare word problem. Previously proposed methods for data augmentation do not consider language semantics when generating synthetic training data. This leads to generation of sentences that lower the overall quality of parallel training data. In this paper, we discuss the suitability of using Parts of Speech (POS) tagging and morphological analysis to prune the generated synthetic sentence pairs that do not adhere to language semantics. Our models show an overall 2.16 and 5.00 BLEU score gains over our benchmark Sinhala to Tamil and Tamil to Sinhala translation systems, respectively. Although we focus on Sinhala and Tamil NMT for the domain of official government documents, we believe that these synthetic data pruning techniques can be generalized to any language pair.

Keywords: Neural Machine Translation, POS Tagging, Morphological Analysis, Data Augmentation

9 Introduction

Neural Machine Translation (NMT) is the current state-of-the-art machine translation architecture that aims at building a single neural network that can be jointly tuned to maximize the translation performance. Despite being successful in producing acceptable outputs for language pairs having large parallel corpora (Sennrich et al, 2016), NMT performs poorly for language pairs that lack the luxury of having sufficiently large parallel data (Tennage et al, 2017). This is due to the requirement of numerous instances of sentence pairs with words occurring in different contexts in order to accurately train a NMT model. Being an under-resourced language pair that is unable to satisfy this requirement, Sinhala and Tamil NMT falls short of reaching state-of-the-art performances (Tennage et al, 2017).

Limited corpus size directly influences the rare word problem. Rare word problem refers to the inability of the neural network to properly model words that appear in the corpus only a very few times. Being morphologically rich languages, there exist many inflections for each word in Sinhala and Tamil languages. Hence having many rare words in the corpus is inevitable.

One way to handle the rare word problem is to increase corpus size using synthetic parallel training data. To generate synthetic data, Fadaee et al. (2017) presented a possible technique. This creates new contexts for rare words when generating synthetic training data, thus giving a possible solution for the rare word problem. However, the main limitation of this approach is that this synthetic sentence pair generation technique does not take language semantics into consideration, which eventually lowers expected BLEU score gain.

In this paper, we present two sentence pruning techniques based on Parts of Speech (POS) tagging and morphological analysis to remove synthetic sentence pairs that do not preserve language semantics. Compared to Fadaee et al.'s (2017) method, POS tagging method shows an improvement of 1.04 and 2.12 BLEU score gains for Sinhala to Tamil (Si-Ta) and Tamil to Sinhala (Ta-Si) models, respectively. Use of morphological analyzing improves the quality of translation by 1.26 and 2.98 BLEU scores. Overall, synthetic parallel training data methods yield an improvement of 2.16 and 5.00 BLEU score gains over our benchmark Si-Ta and Ta-Si models.

10 Background and Related Work

Fadaee et al. (2017) presented a data augmentation approach that targets low-frequency words by generating new sentence pairs containing rare words in new, synthetically created contexts. They have produced experimental results on low-resource settings and have achieved considerable improvement over the benchmark systems. They have focused mainly on fluency and grammatical structure of synthetic training data, and have disregarded its semantic correctness.

Strategies to train with monolingual data without changing the neural network architecture have been proposed by Sennrich et al. (2016). It is based on the intuition that encoder-decoder NMT architecture already has the capacity to learn the same information as a language model. By pairing

monolingual training data with an automatic back-translation, synthetic parallel training data are generated. Quality of synthetic training data generated using this method highly depends on the machine translator that is used for back translation.

11 Methodology

11.11 Initial data augmentation

For initial synthetic sentence generation, we use the technique used by Fadaee et al. (2017). Initially we obtain a list of rare words by considering the unique words and their counts. Words that appear only R (rare word threshold) times or less are considered as rare words. For each rare word r, we iterate through each sentence pair in our parallel corpus.
In the below expressions, s_i and t_i denote the i^{th} word in the source sentence and target sentence, respectively. Each word in the source sentence is iterated through and substituted by r. Trigram language probability around r is checked thereafter. If the i^{th} source word is substituted,

Original language probability $p_1 = LM(s_{i-1}, s_i, s_{i+1})$
Synthetic language probability $p_2 = LM(s_{i-1}, r, s_{i+1})$

if ($p_2 > M * p_1$): this is a valid source substitution (M is fluency threshold).

To generate the target side synthetic sentence, we need to substitute the translation of r to the word in the corresponding original target sentence that is aligned to the word that we removed from the source sentence. Statistical approach of automatic word alignment (Och, Ney, 2004) is used to accomplish this task. Using automatic word alignment, it is possible to get the index of the target word that is aligned with the source word that was removed.
To get the translation of a rare word r, phrase tables that are generated using word alignment are used. For a given word e, there exist several possible translations f according to the generated phrase tables. To find the exact translation, we use a two-way translation probability as follows.

$$translation(e) = \underset{f \in possible\ translations}{argmax} (p(f|e) * p(e|f))$$

where,

p(f|e) : Probability of f being the translation of e.
p(e|f) : Probability of e being the translation of f.

If there exists a target side word q corresponding to r, with two-way translation probability greater than T (translation threshold), we select it as a viable translation for r. q is substituted to the word that is aligned to the word that was removed in source side. If the trigram language probability around that word is greater than M times the original trigram language probability, then we select it as a correct target word substitution.
A synthetic sentence pair that satisfies all these conditions is added to the synthetic parallel corpus. To reduce distortion of the meaning, only a single rare word substitution per sentence was allowed. Use of language modeling ensures the fluency of synthetic sentences whereas use of the translation modeling ensures the correspondence between source sentence and target sentence. Table 1 depicts an example synthetic sentence pair.

Original Sentence Pair	**Synthetic Sentence Pair**
එසේ පවරා දෙනු ලැබුවේ කවරෙකුටද (/* esea pavaraa denu lAbuwea kavarekuTada*/) - (It was assigned to whom?)	එසේ පවරා දෙනු ලැබුවේ ඔබටද (/*esea pavaraa denu labuwea oba Tada */) (It was assigned to you?)
அவ்வாறு யாருக்கு ஒப்படைக்கப்பட்டுள்ளது? (/*avvaaRu yaarukku oppataikkappattuLLadhu*/) (It was assigned to whom?)	அவ்வாறு யாருக்கு உங்களுக்கும்? (/*avvaaRu yaarukku ungkaLukkum*/) (It was to whom and to you)[1]

Table 1: Initially Generated Synthetic Sentence Pair

Human evaluation of the synthetic parallel training data generated using this method revealed that the resulting sentences do not preserve language semantics. Hence, we investigated on methods to prune the synthetic sentence pairs that do not preserve language semantics.

11.12 Parts of Speech tagging

POS tags contain important semantic details about the word in the context that it appears. Based on this property, we further increased the quality of synthetic training data by checking the POS tag of each rare word that is substituted.
Initially, the original parallel corpus is POS tagged. Then using the methodology proposed in section 3.1, all possible synthetic sentence pairs are generated. Then the synthetic parallel sentences are also POS tagged. Algorithm 1 describes this method.

Here,
s_i = word that was removed from source sentence.
t_i = word that was removed from target sentence.
r = rare word that was introduced to source sentence.
t = translation of r that was introduced to target side.

[1]Word ordering in Sinhala is different from English. The exact English translations are “To whom was it assigned?” / “Was it assigned to you?” (“To whom” is replaced by “to you”)

11.13 Morphological Analysis

To further preserve language semantics, we use morphological features. In this research, we pay attention to morphological features of Sinhala nouns only, since most of the rare words are noun word forms. We use two morphological features of Sinhala nouns,

1. Count (වචනය /*wachanaya*/)
2. Case (විභක්තිය /*wibhaktiya*/)

Count can take three values, definite singular (DS), indefinite singular (IS) and definite plural (DP). Case is a suffix that is added to a stem to derive nouns in different meanings. Sinhala language consists of 9 cases, ප්‍රථමා (/*prathamaa*/) - Nominative, කර්ම (/*karma*/) - Accusative, කර්තෘ (/*kartru*/) – Auxiliary, කරණ (/*karaNa*/) - Instrumental, සම්ප්‍රදාන (/*sampradaana*/) - Dative, අවදි (/*awadi*/) - Abalative, සම්බන්ධ (/*sambandha*/) - Possesive, ආධාර (/*aadhaara*/) - Locative, ආලපන (/*aalapana*/) – Vocative (Priyanga, Ranatunga & Dias, n.d.).

Synthetic parallel corpus that was generated in section 3.2 is further improved using morphological features. For a given word, there exists a variable number of case - count combinations. In this approach, we check whether the case - count combinations of the word that was removed have an intersection with the case - count combinations of the word that is introduced synthetically. We consider it as a semantic preserving sentence pair only if there exists an intersection of at least one element.

12 Experimental Setup

11.14 Data Collection and Preprocessing

Official government document translation is the domain used in this translation task. We used the parallel corpus developed by Farhath et al. (2017). Parallel corpus features government documents, annual reports, gazette papers, establishment codes, order papers, official letters and parliament documents. Characteristics of the Sinhala-Tamil parallel dataset are shown in Table 2.

Language	Total Words	Unique Words	Sentences
Sinhala	267,613	21,548	19,153
Tamil	226,160	38,651	

Table 2: Characteristics of the parallel dataset

Parallel corpus was divided into 3 parts: training set (14653 sentence pairs), validation set (4000 sentence pairs) and testing set (500 sentence pairs).

11.15 Experimental Setup

The open source NMT system: OpenNMT (Klein et al. 2017) was used for the experiments. GIZA++ (Och, Ney, 2004) was used for automatic word alignment. It uses the standard alignment heuristic grow-diag-final for word alignment. Tri-gram language models were trained for both source side and target side using the Stanford Research Institute Language Modeling toolkit (Stolcke et al, 2002) with Kneser- Ney smoothing. For translation evaluation, Bilingual Evaluation Understudy (BLEU) metric (Papineni et al, 2002) was used.

11.16 Benchmark training

Using the above parallel corpus, Si-Ta and Ta-Si translation models were trained. Training involved two steps: pre-processing and model training. After completing the pre-processing step, two dictionaries (source dictionary and target dictionary) were generated to index mappings. Using two dictionaries and the serialized file, a model was trained with a 2-layer LSTM with 500 hidden units on both encoder and decoder.

11.17 Initial data augmentation

Out of 15383 number of unique words, 6421 words appeared only once in the Sinhala language side whereas corresponding values for Tamil side was found to be 31186 and 17238, respectively. Hence, we chose rare word threshold R to be 1. Considering the tradeoff between the number of sentences generated and semantic preservation of synthetic data, we chose fluency threshold M to be 2 and translation threshold T to be 0.9.

11.18 POS tagging

Both original and synthetic parallel corpora that were generated in the previous section were POS tagged. We used the POS tagger developed by Fernando et al. (2016) for Sinhala, and the POS tagger developed by the Computational Linguistic Research Group (2017) for Tamil.

11.19 Morphological Analysis

We considered morphological features only when generating the Sinhala side of the parallel corpus. We used Helabasa - Noun Analyzer (2017) to retrieve Sinhala morphological features. When training the translation model for each technique, we appended the synthetic corpus generated from that technique to our original corpus in one to one ratio and trained a separate model.

13 Results and analysis

Table 3 provides examples resulting from each augmentation procedure.

Method	Example
3.1	Si: එතුමා මෙම සභාවට [දන්වන්නෙහිද / **බඳවාගැනීමට**]? (/*etumaa mema sabhaavaTa [danwannehida / **banndhavaagAniemaTa**] ?*/) Ta: **அவர் இச்சபைக்குத்** [**தெரிவிப்பாரா**/ ***ஆட்சேர்ப்பிற்கு***] ? (/*avar issapaikkudh [dherivippaaraa/ *aatseerppiRku*]*/) (En: For this session, he [will inform/ for hiring]?)[2]
3.2	Si: පුහුණු [සැලස්මට / **බඳවාගැනීමට**] අදාල තොරතුරු (/*puhuNu [sAlAsmaTa / **banndhavaagAniemaTa**] adaala toraturu*/) Ta: **பயிற்சித் திட்டத்திற்கு** [**பொருத்தமான** / ***ஆட்சேர்ப்பிற்கு***] **தகவல்கள்** (/*payiRsidh dhittadhdhiRku [porudhdhamaana / *aatseerppiRku*] dhakavalkaL*/) (En: Information [related to training planning/ related to training hiring])[3]
3.3	Si: පහත සඳහන් ලිපිනයට කරුණාකර [ලදුපතක් / **කබායක්**] එවීමට කටයුතු කරන්න. (/*pahata sanndhahan lipinayaTa karuNaakara [ladupatak / **kabaayak**]ewiemaTa kaTayutu karanna.*/) Ta: **கீழ் காணும் முகவரிக்கு தயவு செய்து** [**பற்றுச் சீட்டொன்றை** / ***மழைக்காப்பு***]**அனுப்ப நடவடிக்கை எடுக்கவும்.** (/*kiiz kaaNum mukavarikku dhayavu seydhu [paRRus siittonRai / *mazaikkaappu*]anuppa watavatikkai etukkavum.*/) (En: Kindly send a [receipt/coat] for the following address)

Table 3: Examples synthetic data with highlighted [original / **substituted**] and [original /*translated*] words

Considering Table 3, in the initial data augmentation method (first row), substituting බඳවාගැනීමට (/*banndhavaagAniemaTa*/ - to hire) with දන්වන්නෙහිද (/*danwannehida*/ - will inform?) makes the resulting synthetic sentence meaningless. Sentences that are generated by analyzing POS tags seem to have an edge over initial data augmentation method. Since සැලස්මට (/*salasmataTa*/ - for plan) and බඳවාගැනීමට (/*banndhavaagAniemaTa*/ - to hire) (second row) have identical POS tags, high fluency is achieved in the resulting synthetic sentence pair. Synthetic sentence pair generated using the method mentioned in 3.3, preserves the meaning to a better extent. Word ලදුපතක් (/*ladupatak */ - receipt) and කබායක් (/*kabaayak */ - coat) have indefinite singular – Nominative, Accusative, Auxiliary, Locative morphological features in common.

Table 4 depicts the BLEU scores obtained for each method.

Method	Si-Ta	Ta-Si
Benchmark training	6.78	6.84
+ Initial data augmentation	7.68	8.86
+POS tagging	8.72	10.98
+Morphological Analysis	8.94	11.84

Table 4: BLEU scores

To examine the impact of augmenting training data by creating contexts for rare words on the target side, we tested how each model performs on rare words. Most of the rare words are not ‘rare’ anymore in the augmented data since they were augmented sufficiently many times.

Table 5 depicts an example of successful translation of a rare word.

Reference Source	**විශේෂ පිරිවු** (/*viseeta pirivu*/)
Reference Target	විශේෂ ඒකකය (/*viSeasha eakakaya*/) (Special unit)
Benchmark Model	විශේෂ අංශය (/*viSeasha a\nSaya*/) (Special sector) - erroneous
Our method (3.2 and 3.3)	විශේෂ ඒකකය (/*viSeasha eakakaya*/) (Special unit) - correct

Table 5: Rare Word example

Synthetic data generated using the initial data augmentation method have improved the performance of Si-Ta and Ta-Si translation by 0.9 and 2.02 amounts, respectively. To verify that this

[2]Exact English translation: “Will he inform this session?” / “For this session will he be hiring?” (“will inform” is replaced by “for hiring”)

[3]Exact English translation: “Information related for training planning” / “Information related for training hiring” (“for training" is replaced by "for hiring")

gain is due to the rare word substitutions and not just due to the repetition of part of the training data, we performed an experiment where each sentence pair selected for augmentation is added to the training data unchanged (i.e. without creating synthetic data). This simple form of sampled data replication delivered 0.53 and 1.42 BLEU score gains for Si-Ta and Ta-Si, respectively. Hence initial data augmentation models have performed better compared to simple data replication method.

Use of POS tags has achieved 1.04 and 2.12 BLEU score gains over the initial data augmentation for Si-Ta and Ta-Si, respectively. Human evaluators who oversaw the quality of generated sentences revealed that the use of POS tags has increased the fluency of language and rare word translation performance by a significant amount. Thus, we can empirically prove that the use of POS tags improves the quality of synthetic training data, which in turn reduces the rare word problem in NMT.

Morphological features have played a vital role in reducing the rare word problem. When generating synthetic sentence pairs, we considered only Sinhala language morphological features. Sinhala being morphologically rich, there exist many number of variations for a given root word. Hence checking the case-count combinations of a word when substituting, helps to preserve language semantics of the generated sentence. This is evident by analyzing the BLUE score gains of 1.26 and 2.98 for Si-Ta and Ta-Si translations compared to the initial data augmentation method.

BLUE score gains are consistent across both translation directions, regardless of whether rare word substitutions are first applied to Sinhala or Tamil. Hence it can be verified that using POS tagging and morphological features results in generating quality synthetic parallel data that preserve language semantics, which eventually leads to better translation performance.

Though overall rare word translation quality was improved by our methods, there were several cases where augmentation resulted in incorrect outputs that were correctly translated by our benchmark system. Table 6 corresponds to such an incorrect translation.

Source Sentence	8. உள்ளூர்/ வெளிநாட்டு திரைப்ப ட தயாரிப்பாளர்களுடன் /*uLLur/ veLiwaattu dhiraippa ta dhayaarippaaLarkaLutan*/ - (With local and foreign film producers)
Reference Translation	8 .දේශීය / විදේශීය චිත්‍රපට නිෂ්පාදකයින් සමඟ /*8 .deaSieya / videaSieya citrpaTa nishpaadakayin samannga */ - (With local/foreign film producers)
Benchmark translation	8 .දේශීය විදේශීය විකාශන කටයුතු සඳහා/* 8 .deaSieya videaSieya vikaaSana kaTayutu sanndhahaa*/ - (For local/ foreign broadcasting)
Our method (3.2 and 3.3)	(8) විදේශ චිත්‍රපට ගනුදෙනු කිරීම /* (8) videaSa citrapaTa ganudenukiriema . */ - (For trading foreign films)

Table 6: Incorrect outputs

Our benchmark model has been able to correctly translate දේශීය (/*deaSieya*/ - local) and විදේශීය (/*videaSieya*/ - foreign) terms, whereas our new model has not been able to translate any of them. If the language model selects substitutions that have low probabilities, it results in generating outputs with low fluency. Another possible reason is errors in word alignments. If the word alignments are erroneous and phrase table contains faulty probabilities, this may lead to synthetic sentence pairs that do not correspond to each other.

14 Conclusion

The purpose of this research was to find out semantic preserving techniques for synthetic data generation to solve the rare word problem in NMT for the under-resourced language pair Sinhala and Tamil. POS tagging and morphological analysis show impressive results in reducing the rare word problem. Being morphologically rich, there exist a number of morphological features in Sinhala and Tamil that can be exploited to enhance the quality of augmented data. We expect to experiment with these features in the future.

15 Acknowledgements

The authors would like to thank the reviewers for their helpful comments and suggestions. The authors are grateful to members of the National Languages Processing Centre at University of Moratuwa for their significant contribution in developing the basic linguistic resources, and the Department of official languages of Sri Lanka for providing corpus data needed to carry out the research.

16 Bibliographical References

Bahdanau, D., Cho, K., &Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align

and Translate. Arxiv Preprint Arxiv:1409.0473 [Cs.CL].
Cho, K., van Merrienboer, B., Bahdanau, D., &Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8) ,pp. 103-111.
Fadaee, M., Bisazza, A., &Monz, C. (2017). Data Augmentation for Low-Resource Neural Machine Translation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics ,pp. 567-573.
Klein, G., Kim, Y., Deng, Y., Senellart, J., & M. Rush, A. (2017). OpenNMT: Open-source toolkit for neural machine translation",.Arxiv Preprint Arxiv:1701.02810 [Cs.CL],.
Levenshtein distance. (2017). En.wikipedia.org. Retrieved 26 September 2017, from https://en.wikipedia.org/wiki/Levenshtein_distance
Och, F. J., & Ney, H. (2004). The alignment template approach to statistical machine translation. Computational linguistics, 30(4),pp. 417-449.
Papineni, K., Roukos, S., Ward, T., & Zhu, W. (2002). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics ,pp. 311-318.
Priyanga, R., Ranatunga, S., & Dias, G. An Inflectional Morphological Generator for Sinhala Nouns. (Unpublished)
Pushpananda, R., Weerasinghe, R., & Niranjan, M. (2014). Sinhala-Tamil Machine Translation: Towards better Translation Quality. In proceedings of Australasian Language Technology Association Workshop, 129, pp. 129-133.
Sennrich, R., Haddow, B., & Birch, A. (2016). Edinburgh Neural Machine Translation Systems for WMT 16. In Proceedings of the First Conference on Machine Translation (WMT) ,pp. 371-376. Association for Computational Linguistics.
Sennrich, R., Haddow, B., & Birch, A. (2016). Improving Neural Machine Translation Models with Monolingual Data. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 86-96.
Stolcke, A. (2002). SRILM-an extensible language modeling toolkit. In Interspeech. pp. 901-904.
Tennage, P., Sandaruwan, P., Thilakarathne, M., Herath, A., Ranathunga, S., Dias, G., & Jayasena, (2017). Neural Machine Translation for Sinhala and Tamil Languages, *(Submitted)*.
Welgama, V., Herath, D., Liyanage, C., Udalamatta, N., Weerasinghe, R., &Jayawardana, T. (2011). Towards a Sinhala Wordnet. In Conference on Human Language Technology for Development ,pp. 39-43.

17 Language Resource References

Computational Linguistic Research Group. (2017). Au-kbc.org. Retrieved 2 October 2017, from http://www.au-kbc.org/nlp/corpusrelease.html
Farhath, F., Ranathunga, S., Jayasena, S., Dias, G., & Thayasivam, U. (2017). Improving Domain-Specific Statistical Machine Translation for Sinhala-Tamil using Bilingual Lists, *(Submitted)*.
Fernando, S., Ranathunga, S., Jayasena, S., & Dias, G. (2016). Comprehensive Part-Of-Speech Tag Set and SVM based POS Tagger for Sinhala. In Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (pp. 173 - 182).
Helabasa - Noun Analyzer. (2017). Translation.projects.mrt.ac.lk. Retrieved 1 October 2017, from http://translation.projects.mrt.ac.lk:8081/helabasa/noun_analyzer