Neural Machine Translation for Sinhala-English Code-Mixed Text

Archchana Kugathasan
Department Of Computational Mathematics
University Of Moratuwa
Katubedda
archchanakugathasan@gmail.com

Sagara Sumathipala
Department Of Computational Mathematics
University Of Moratuwa
Katubedda
sagaras@uom.lk

Abstract

Code-mixing has become a moving method of communication among multilingual speakers. Most of the social media content of the multilingual societies are written in code-mixed text. However, most of the current translation systems neglect to convert code-mixed texts to a standard language. Most of the user written code-mixed content in social media remains unprocessed due to the unavailability of linguistic resource such as parallel corpus. This paper proposes a Neural Machine Translation(NMT) model to translate the Sinhala-English code-mixed text to the Sinhala language. Due to the limited resources available for Sinhala-English code-mixed(SECM) text, a parallel corpus is created with SECM sentences and Sinhala sentences. Srilankan social media sites contain SECM texts more frequently than the standard languages. The model proposed for code-mixed text translation in this study is a combination of Encoder-Decoder framework with LSTM units and Teachers Forcing Algorithm. The translated sentences from the model are evaluated using BLEU(Bilingual Evaluation Understudy) metric. Our model achieved a remarkable BLEU score for the translation.

1 Introduction

Before 1990, translation was considered a difficult task due to many reasons such as ambiguity, translation mismatch, co-reference, translation divergence and development of language over time(Sreelekha et al., 2016) but Machine Translation(MT) since 1990 has been a vast and successful research area in natural language processing. Machine translation has been given importance in the research field because it is used to translate texts for military authorities to track enemies, foreign business collaborations, marketing, etc.(Kalchbrenner and Blunsom, 2013).

Expressing the thoughts of the personal interests, daily life etc., of a person in social media networks has become a trending activity among people. Texts extracted from social media lead to measure the social dynamics of several societies(Arguello et al., 2008). Content-based search engines, personalized advertisements, recommendation systems, etc. use the user-generated content from social media to increase their value and to provide more accurate results to the users(Sippel and Brodt, 2008). Processing a content in standard language(without code-mixing) is considered as an easy task, where the text with code-mixing has been considered as a road block to extract the needed information. Code-mixing is considered as an invention of bilingualism and multilingualism. The capability of speaking in two languages, is called bilingualism and more than two languages is called multilingualism. Most of the Srilankans are bilingual. The user-generated content such as posts, comments,reviews etc., in Srilankan social media are mostly in SECM text. The main focus of this study is to translate the SECM text to Sinhala language.

To understand the necessity of processing SECM text, a survey study was conducted among 82 native Sinhala speaking citizens as a part of our research study to collect information about the usage of SECM text in Sri-Lanka. Figure 1 shows the results of a few essential questions from the survey. The majority of the people have stated that they use SECM text in social media rather than their native language.

Proceedings of Recent Advances in Natural Language Processing, pages 718–726
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_082

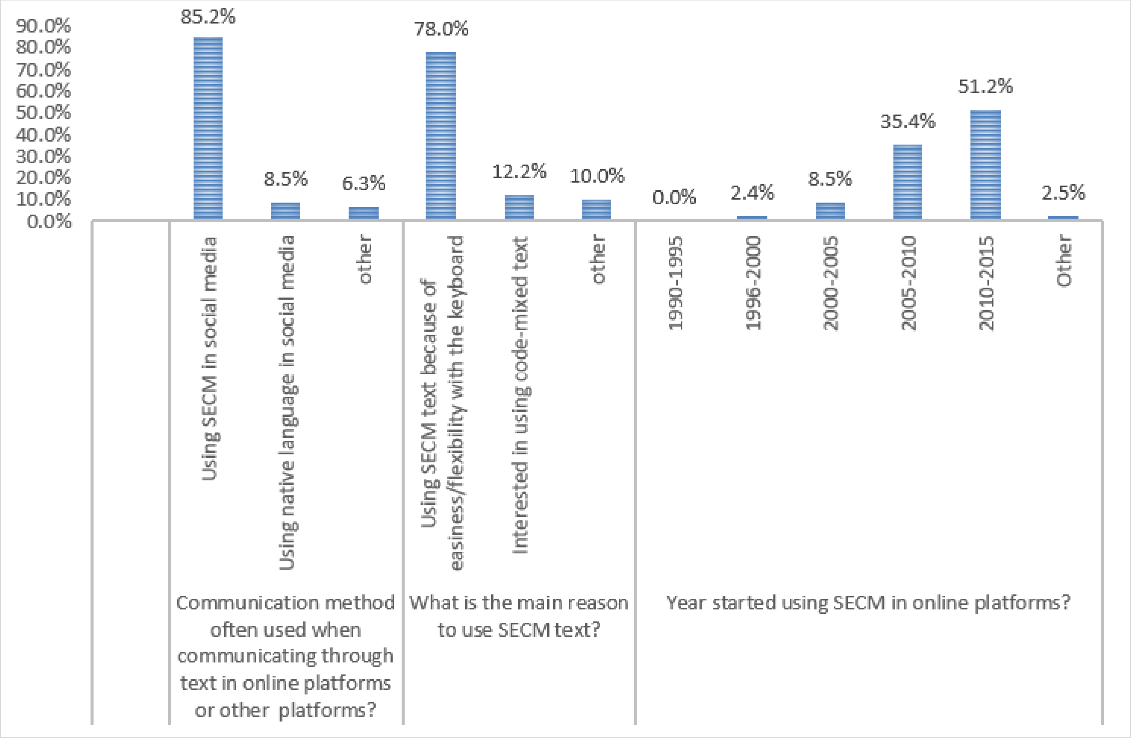


Figure 1: Survey results on Sinhala-English code-mixed text usage among Srilankans

In SECM texts there are several problems identified related to morphology, syntax and the semantic structure of the text. The challenges found in SECM texts are,

- Inconsistency in the transliteration
- Spelling mistakes
- Code-switching
- Words combined with suffixes of another language
- Improper usage of discourse marker
- Unnecessary numerical characters combined with words

For example, in the SECM sentence shown in Figure 2, the words '*lassana*', '*ekak*', '*ekka*' and '*gatha*' are transliterated (Words from one language written with the alphabet of another language) Sinhala words. The word '*atmosphere*' and '*so*' are English words. The sentence starts with Sinhala transliterated word, switches to English and again switches back to Sinhala transliterated format. The language of words is switched from one to another in a single sentence, called as code-switching. The word '*so*' is a discourse marker in English, which is used for joining two sentences which has Sinhala base. The '*friendla*' is a SECM word, where English singular noun '*friend*' is combined with Sinhala transliterated suffix '*la*' to make it look like the plural word '*Friends*'. The word '*4to*' represent the English word '*Photo*' with spelling mistakes and unnecessary numerical character. The numerical character combined with the word presents the phonetic sound of the word '*four*' and '*to*'. Together it is understood as the word '*photo*'. The research study of Kugathasan and Sumathipala (2020) clearly explains the challenges in Sinhala-English code-mixed texts.

This paper is divided into six sections. Section 2 elaborates on the works related to Machine Translation. Section 3 describes the SECM texts and corpus creation. The Section 4 explains the methodology, which explains about the model implementation and prediction. Section 5 describes the experimental setting of the model and the result gained. As the final section the paper discusses the conclusion of the research study.

2 Related Work

The high demand for Machine translation is increased due to many reasons such as business in overseas, tracking down the information of another language for military services, high usage of social media, etc.

Machine translation was initiated by Warren Weaver in 1955. The research combined Statistical Machine Translation(SMT) with Claude Shannon's information theory(Weaver, 1949/1955). SMT models output the degree of similarity between the source and target sentences (Carrera et al., 2009). The structure of a sentence, feature engineering and design are considered as valuable factors in SMT. Also SMT approach is noted as 'not suitable' for generalized sentence pair, sentences with hidden details (Kalchbrenner and Blunsom, 2013) and language pairs with different word orders (Masoud et al., 2019). Chiang (2005) and Koehn et al. (2003), showed that features focused in SMT are not helpful to track the long-distance dependency of a sentence like in Recurrent Neural Network. Collobert and Weston (2008) explained, how semantic, synthetic and morphological similarities are captured better when there is continues representation of words that carry task-dependent knowledge.

Kalchbrenner and Blunsom (2013) introduced the Recurrent Continues Translation model, which has two parts, Convolutional Sentence Model and Recurrent Language Model. Convolutional Sentence Model, uses a convolutional n-gram approach on source sentences in the encoder. The sentences are mapped into semantic vectors. The recurrent language model is applied in the decoder. A similar approach is proposed by Cho et al. (2014), where the Recurrent Neural Network(RNN) is used for translation. Sentences needing translation are encoded into a sequence with a fixed length vector and decoded with another sequence of symbols. In this approach, the encoder and the decoder are jointly trained to increase the conditional probability of phrase pairs in the sentence using RNN.

Some research studies on translation are based on a monolingual dataset. A semi-supervised approach is proposed by Cheng and Duan (2020) with labelled and unlabeled corpus. Labelled corpus is a parallel corpus with source and target sentences of the Chinese-English dataset. Unlabeled corpus contains the monolingual dataset. In the semi-supervised setting, parallel and monolingual corpus are joined to learn Bidirectional NMT(source to target and target to source models). Sennrich et al. (2015) proposed two approaches to translate monolingual datasets. In the first approach the monolingual corpus is matched with dummy inputs to construct the parameters of the encoder with attention model (Choi et al., 2018) and the second approach utilizes a pre-trained NMT model.

Using NMT for translating one standard language to another standard language has been a success over the decade(Sreelekha et al., 2016). However, it is not experimented well with the code-mixed text due the to lack of resources. In the domain of translating code-mixed text, very few researches have been carried on. A combined approach of Statistical Modelling (Neale et al., 1999) and Knowledge Translation(Sudsawad, 2007) approach is introduced by Carrera et al. (2009) for cross-language social media texts. Rijhwani et al. (2016) introduce an approach for translating code-mixed text, where words in sentences are categorized as dominant and non-dominant languages. Words from dominant language are labelled as matrix language and non-dominant language are labelled as embedded language. The first task before the translation was word-level language identification. Next, the data is applied to a current translator to translate the words to another language. Dhar et al. (2018) used a code-mixed corpus from ICON 2017 tool contest for translation. Machine translation augmentation approach was used in their research study which achieved a BLEU score of 16.90. Masoud et al. (2019) used a combined corpus from OPUS3 and EnTam4, evaluated the corpus using several approaches and calculated the BLEU score. Word Hybrid Baseline approach achieved a BLEU score of 21.05, Byte Pair Encoding (BPE) Hybrid baseline approach achieved a BLEU score of 21.93, Word Hybrid Baseline Google approach received a BLEU score of 21.35. Finally, the Word Hybrid Baseline Google approach achieved the

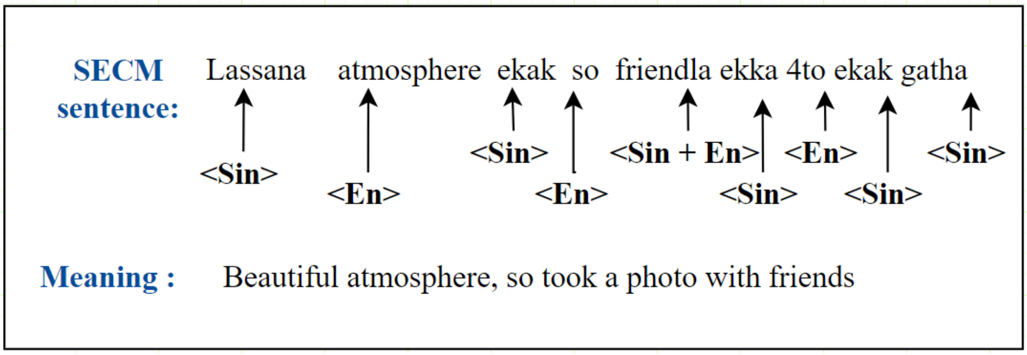

SECM sentence:
Lassana atmosphere ekak so friendla ekka 4to ekak gatha
<Sin>
<En>
<Sin>
<En>
<Sin + En>
<Sin>
<En>
<Sin>
<Sin>
Meaning :
Beautiful atmosphere, so took a photo with friends

Figure 2: Example of SECM code-mixed text. <Sin> - Sinhala transliterated word, <En> - English word , <Sin + En> - Combination of Sinhala transliterated word and English word

highest BLEU score of 22.46.

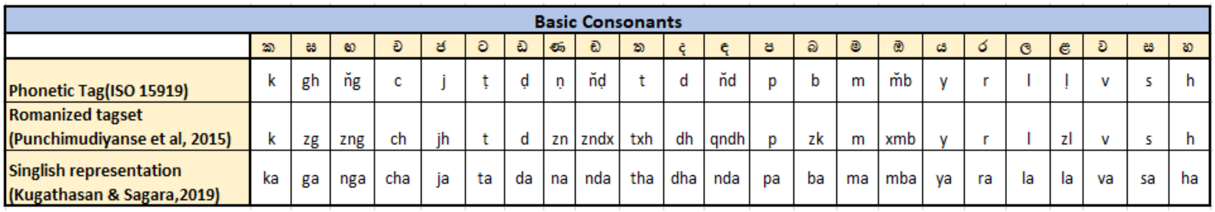
3 SECM text and corpus creation

Sinhala is the native language of the majority of the people in Sri Lanka. But in Srilankan social media, Sinhala-English code-mixed text is used frequently because of the multi-lingual users. In 2001 International Organization for Standardization published the ISO15919 standard. It is an international standard for romanization which includes many languages including Sinhala language . Weerasinghe et al. (2005) used the IPA(International Phonetic Alphabet) format to represent Sinhala letters in their research study. ISO15919 and IPA both present Sinhala letters with English alphabets, which is called transliterated format or romanized text format (Hettige and Karunananda, 2007). Wasala et al. (2006) propose a conventional tag set, which uses the 26 alphabets of English to present the phonetic sound of Sinhala letters using the festival framework.

Code mixing is described as a way of writing roman script (Davies and Bentahila, 2007). Even though the standard tag sets from Punchimudiyanse and Meegama (2015) is defined as roman representation, the romanization of actual code-mixing used by multilingual societies is different from the standard tag sets. Figure 3 and Figure 4 shows us how the standard romanization defined for Sinhala letters differs from the roman representation used in Singlish text.

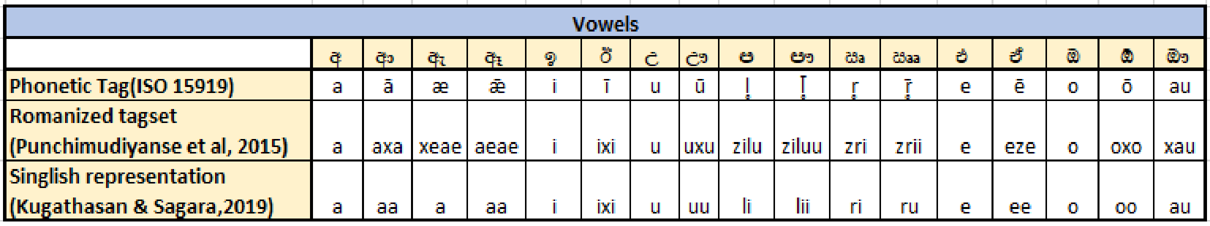
According to Figure 3 and Figure 4, Phonetic Tagset(PT) and Romanized Tagset(RT) are almost similar. But there is a huge difference between these two tagset and code-mixed text representations of Sinhala letters. Due to no consistency in the pattern of Singlish text, it is not easy to translate the Sinhala code-mixed text without a parallel corpus. To achieve a good outcome most machine translation systems needs a sufficient amount of parallel sentences in the corpus.

We collected the SECM sentences from public Facebook posts, comments and reviews. For the translation of SECM to Sinhala, implementing the parallel corpus is an important part for our research study. Each SECM sentence was human translated by a linguistic expert of Sinhala language to create the parallel corpus. In the parallel corpus SECM is considered the source sentence and Sinhala is considered the target sentence. The human translator was advised to follow the Singlish to Sinhala mapping provided in the research study of Kugathasan and Sumathipala (2020) as the guide to maintain the consistency in the translation. The corpus contains around 1500 parallel sentences of SECM and it's translated Sinhala sentences. After the human translation the dataset was checked to see whether the translated sentences are FC (Fully Correct) or CR (Correction Required). If a parallel sentence was annotated with a Correction Required tag by the annotator, the same annotator would provide the alternate translation as well. Each sentence in the corpus is annotated by two annotators. The annotators are people whose native language is Sinhala and we made sure that they are fluent in the Sinhala language. The annotators were provided with guidelines regarding the annotation process. The guidelines made sure that the annotators were checking whether there are any



Basic Consonants																						
	ක	ඝ	ඟ	ච	ජ	ට	ඩ	ණ	ඬ	ත	ද	ඳ	ප	බ	ම	ඹ	ය	ර	ල	ළ	ව	ස	හ
Phonetic Tag(ISO 15919)	k	gh	ňg	c	j	ṭ	ḍ	ṇ	ňḍ	t	d	ňd	p	b	m	m̆b	y	r	l	ḷ	v	s	h
Romanized tagset (Punchimudiyanse et al, 2015)	k	zg	zng	ch	jh	t	d	zn	zndx	txh	dh	qndh	p	zk	m	xmb	y	r	l	zl	v	s	h
Singlish representation (Kugathasan & Sagara,2019)	ka	ga	nga	cha	ja	ta	da	na	nda	tha	dha	nda	pa	ba	ma	mba	ya	ra	la	la	va	sa	ha

Figure 3: Difference between standard phonetic tags, romanized tags and Singlish representation for Sinhala basic consonants



Vowels																	
	අ	ආ	ඇ	ඈ	ඉ	ඊ	උ	ඌ	ඏ	ඐ	ඍ	ඎ	එ	ඒ	ඔ	ඕ	ඖ
Phonetic Tag(ISO 15919)	a	ā	æ	ǣ	i	ī	u	ū	ḷ	ḹ	ṛ	ṝ	e	ē	o	ō	au
Romanized tagset (Punchimudiyanse et al, 2015)	a	axa	xeae	aeae	i	ixi	u	uxu	zilu	ziluu	zri	zrii	e	eze	o	oxo	xau
Singlish representation (Kugathasan & Sagara,2019)	a	aa	a	aa	i	ixi	u	uu	li	lii	ri	ru	e	ee	o	oo	au

Figure 4: Difference between standard phonetic tags, romanized tags and Singlish representation for Sinhala vowels

spelling mistakes, grammatical issues in the Sinhala translation.

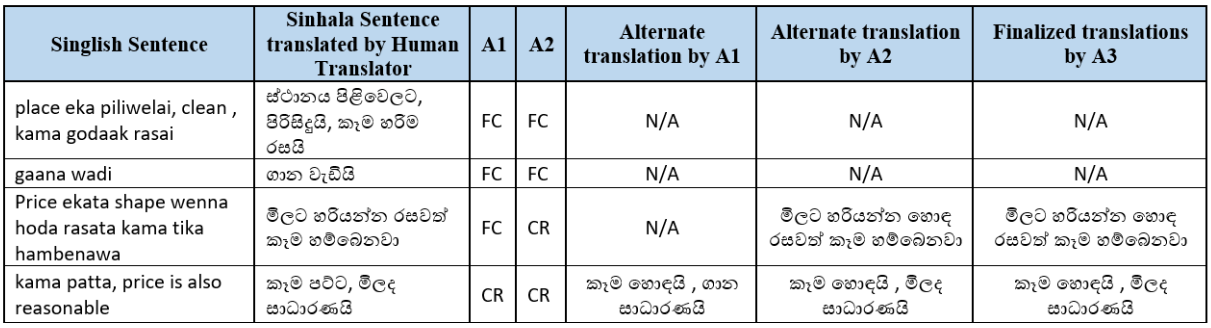
As shown in Figure 5, when there is a sentence tagged with two different tags, for example one annotator annotated with FC tag and the other annotated with CR tag with an alternate translation, we considered the alternate translation. Also when a sentence is annotated with CR tag from both the annotators, the alternate translations provided by both the reviewers are checked with a third annotator and the most suitable translation is selected. To evaluate the standard of the corpus, 100 randomly chosen sentences are provided to 3 experts of Sinhala for ranking. The translations are ranked as good or bad, according to the meaningful translation, grammatical pattern and spelling errors. Fleiss' Kappa approach is used to calculate the reliability of the agreement between the raters(Randolph, 2008). The overall Fleiss' Kappa score received for the translation is 0.88.

4 Methodology

After corpus creation, the dataset was applied with several pre-processing steps. Initially, all the sentences in the Singlish corpus are converted into lower case and all the Sinhala sentences in the target corpus are added with START and END tokens. The unique words from the corpus are extracted and unique numbers are allocated for each word according to the order of frequency of each word. These word-number arrays are called WordToIndex dictionaries.

Encoder-Decoder framework is used as the base to initiate our model. Encoder and decoder can be considered as two separate Recurrent Neural Networks. In the encoder the SECM sentence is fed as input. It produces the sentence with fixed-sized representation by encoding. Each word from the input sentence provided into the encoder would be mapped into an integer using the WordToIndex dictionary and converted into one-hot encoding. The embedding layer maps the one-hot encoded representation into a smaller dimension. The word embedding would be the input to the next layer with Long Short Term Memory(LSTM) as the basic unit. In LSTM we have a cell state that is passed with each timestep. The LSTM unit(Sundermeyer et al., 2012) determines to neglect some unnecessary information and add some new information from the input fed to the current timestep. The significant information collected from the encoder would be passed into a context vector with the output and hidden states. Only hidden states are passed as input to the decoder.

Decoder produces the target sentence using the significant information passed through the hidden state from the encoder and the input target word. Each word from the target sentence is mapped into an integer using Word-



Singlish Sentence	Sinhala Sentence translated by Human Translator	A1	A2	Alternate translation by A1	Alternate translation by A2	Finalized translations by A3
place eka piliwelai, clean , kama godaak rasai	ස්ථානය පිළිවෙලට, පිරිසිදුයි, කෑම හරිම රසයි	FC	FC	N/A	N/A	N/A
gaana wadi	ගාන වැඩියි	FC	FC	N/A	N/A	N/A
Price ekata shape wenna hoda rasata kama tika hambenawa	මිලට හරියන්න රසවත් කෑම හම්බෙනවා	FC	CR	N/A	මිලට හරියන්න හොඳ රසවත් කෑම හම්බෙනවා	මිලට හරියන්න හොඳ රසවත් කෑම හම්බෙනවා
kama patta, price is also reasonable	කෑම පට්ට, මිලද සාධාරණයි	CR	CR	කෑම හොඳයි , ගාන සාධාරණයි	කෑම හොඳයි , මිලද සාධාරණයි	කෑම හොඳයි , මිලද සාධාරණයි

Figure 5: Sample sentences from annotated corpus, A1 - Annotator1, A2 - Annotator2, A3 - Annotator3, FC - Fully Correct, CR - Correction Required, N/A - Not Applicable

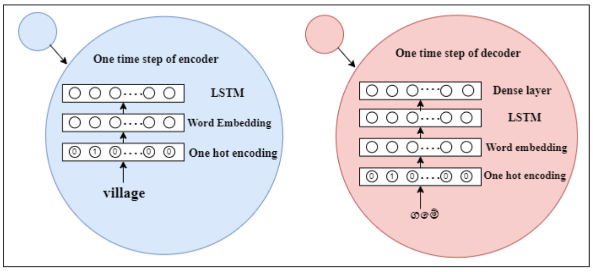


Figure 6: Architecture of each timestep in Encoder and Decoder

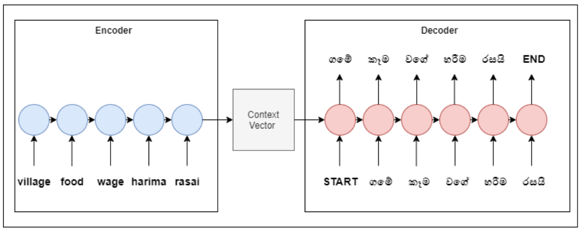


Figure 7: Encoder - Decoder framework

ToIndex dictionary and converted to one-hot encoding. The word embedding layer maps the embedding into a continuous representation which has a lot smaller dimension. Figure 6 shows the architecture inside timesteps in encoder and decoder.

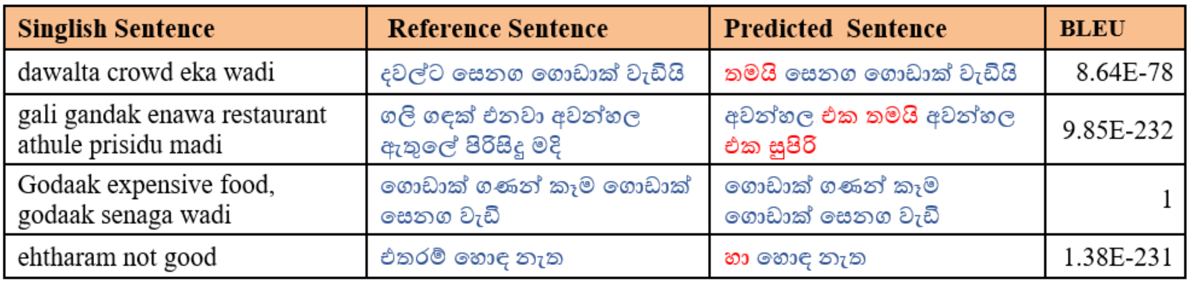
Teacher Forcing algorithm(Goodfellow et al., 2017) is added in the decoder. Teacher Forcing algorithm inputs the expected output of previous timestep *t-1* to the current timestep *t*. The advantage of using Teacher Forcing mechanism is the hidden state of the model would be updated with the correct expected outputs rather than the wrongly predicted output from the previous timestep. If the predicted output from previous timestep *t-1* is fed to the next timestep *t*, the number of errors would be increased and the model would face difficulty in learning. Combination of encoder and decoder is called as Sequence to Sequence model(Seq2Seq) as shown in Figure 7.

Final phase of the proposed architecture of the system is prediction. Singlish sentence from the corpus is given as input, and output would be the predicted Sinhala sentence. Prediction phase is built with the sequence to sequence architecture. Each timestep in decoder passes predicted output for the next timestep unlike the decoder in the training phase of the model.

5 Experimental setting and Result

From the corpus 70% of the data is allocated for training, and 30% of the data is allocated for testing. Inputs for the encoder and the decoder are in the shape of 2D arrays. The shape of the encoder array is (10,27), where the batch size is ten and maximum length of source sentence is twenty seven. Shape of the decoder array is (10,26), where the batch size is ten and maximum length of source sentence is twenty six. Rmsprop is used as the optimizer and Categorical Cross Entropy is used to calculate the loss. The RMSprop optimizer is used because it balances the step size and, decreases the no of steps for massive gradients to neglect the exploding and increases the number of steps for small gradient to avoid vanishing gradients issue. Weights calculated after the training phase of the model are saved for the prediction phase. The model reached the training accuracy of 71.42% and testing accuracy of 37.17%.

After the model's training, randomly se-



Singlish Sentence	Reference Sentence	Predicted Sentence	BLEU
dawalta crowd eka wadi	දවල්ට සෙනග ගොඩාක් වැඩියි	තමයි සෙනග ගොඩාක් වැඩියි	8.64E-78
gali gandak enawa restaurant athule prisidu madi	ගලි ගදක් එනවා අවන්හල ඇතුලේ පිරිසිදු මදි	අවන්හල එක තමයි අවන්හල එක සුපිරි	9.85E-232
Godaak expensive food, godaak senaga wadi	ගොඩාක් ගණන් කෑම ගොඩාක් සෙනග වැඩි	ගොඩාක් ගණන් කෑම ගොඩාක් සෙනග වැඩි	1
ehtharam not good	එතරම් හොඳ නැත	හා හොඳ නැත	1.38E-231

Figure 8: Example of predicted sentences and relevant BLEU score. Words highlighted in red are the words that are different from the reference sentence.

lected hundred SECM sentences from the corpus are inputted to predict the translated Sinhala sentence. The predicted Sinhala sentences are saved to calculate the BLEU score. Figure 8 shows examples of some predicted Sinhala translations. BLEU score is the evaluation metrics (Papineni et al., 2002) used to evaluate the translated sentences. BLEU metric provides a score for the translation based on the predicted sentence and relevant reference sentence. For each sentence in the corpus, modified precision of unigram, bigram, trigram and four-gram are calculated. The weight of 0.25 has been given to each modified precision.

$$BLEU = BP.exp(\sum_{n=1}^{N} W_n log p_n) \qquad (1)$$

Equation 1, is used to calculate the BLEU score. N is the number of n-grams and W_n is the weight for each modified precision, p_n is modified precision. BP is the brevity penalty to penalize short machine translations(Papineni et al., 2002).

$$BP = \begin{cases} 1 & \text{if c > r} \\ exp(1 - \frac{r}{c}) & \text{if c} \leq \text{r} \end{cases}$$

The value of BP is decided according to the values of c and r. c is the number of unigrams in all the predicted sentences, and r is the best match length for each predicted sentence in the corpus. Our model received a cumulative BLEU4 score of 31.54. Comparing to previous models proposed for code-mixed text translation (Dhar et al., 2018; Masoud et al., 2019) our proposed approach with Teacher Forcing mechanism gives a remarkable BLEU score for the translation.

6 Conclusion & Future work

This paper presents a deep analysis of Sinhala-English code mixed texts. The difference between standard tagsets available for the romanization of Sinhala letters and the romanization used in SECM text are compared. The differences are discussed in this study. Challenges in the pattern of SECM sentences such as code-switching, spelling errors, improper usage of discourse marker etc., are also discussed in this paper. A parallel corpus is created containing SECM sentences and the relevant Sinhala sentences translated by a human translator who is a linguistic expert. The corpus is validated using annotators who are native Sinhala language speakers. The parallel corpus introduced in this paper can be considered a useful resource for researches based on SECM text. We combined Teacher Forcing Algorithm with the Sequence to Sequence approach with LSTM units to translate SECM sentences to Sinhala sentences. Teacher Forcing Algorithm updates the hidden state of each timestep in the decoder with the expected output from the previous time step, which leads on providing more accurate results for the translation. The BLEU score received for our model revealed that comparing the state of the art of other translation models for code-mixed texts, our model achieved significantly higher BLEU score. The future work we would like extend this research to focus on sentiment analysis and entity extraction using the parallel corpus created in this research study.

References

Jaime Arguello, Jonathan L Elsas, Jamie Callan, and Jaime G Carbonell. 2008. Document representation and query expansion models for blog recommendation. *ICWSM*, 2008(0):1.

Jordi Carrera, Olga Beregovaya, and Alex Yanishevsky. 2009. Machine translation for cross-language social media. *PROMT Americas Inc.*

Yong Cheng and Mofan Duan. 2020. Chinese grammatical error detection based on BERT model. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 108–113, Suzhou, China. Association for Computational Linguistics.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*, pages 263–270.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Heeyoul Choi, Kyunghyun Cho, and Yoshua Bengio. 2018. Fine-grained attention mechanism for neural machine translation. *Neurocomputing*, 284:171–176.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Eirlys E Davies and Abdelali Bentahila. 2007. Contact linguistics: Bilingual encounters and grammatical outcomes.

Mrinal Dhar, Vaibhav Kumar, and Manish Shrivastava. 2018. Enabling code-mixed translation: Parallel corpus creation and MT augmentation approach. In *Proceedings of the First Workshop on Linguistic Resources for Natural Language Processing*, pages 131–140, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2017. *Deep learning*. MIT Press.

Budditha Hettige and Asoka S Karunananda. 2007. Transliteration system for english to sinhala machine translation. In *2007 International Conference on Industrial and Information Systems*, pages 209–214. IEEE.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1700–1709.

Philipp Koehn, Franz J Och, and Daniel Marcu. 2003. Statistical phrase-based translation. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST.

Archchana Kugathasan and Sagara Sumathipala. 2020. Standardizing sinhala code-mixed text using dictionary based approach. In *2020 International Conference on Image Processing and Robotics (ICIP)*, pages 1–6. IEEE.

Maraim Masoud, Daniel Torregrosa, Paul Buitelaar, and Mihael Arcan. 2019. Back-translation approach for code-switching machine translation: A case study. In *Proceedings of the 27th AIAI Irish Conference on Artificial Intelligence and Cognitive Science*, Galway, Ireland.

Michael C Neale, Steven M Boker, Gary Xie, and H Mx Maes. 1999. Statistical modeling. *Richmond, VA: Department of Psychiatry, Virginia Commonwealth University*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

M Punchimudiyanse and RGN Meegama. 2015. Unicode sinhala and phonetic english bidirectional conversion for sinhala speech recognizer. IEEE International Conference on Industrial and Information Systems 2015.

Justus J Randolph. 2008. Online kappa calculator. *Retrieved October*, 20:2011.

Shruti Rijhwani, Royal Sequiera, Monojit Choudhury Choudhury, and Kalika Bali. 2016. Translating codemixed tweets: A language detection based system. In *3rd Workshop on Indian Language Data Resource and Evaluation-WILDRE-3*, pages 81–82.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Bryan Sippel and Susan E Brodt. 2008. The psychology of blogging communities: Social identities and knowledge transfer across work-groups. In *ICWSM*.

S Sreelekha, Pushpak Bhattacharyya, Shishir K Jha, and D Malathi. 2016. A survey report on evolution of machine translation. *Int. J. Control Theory Appl*, 9(33):233–240.

Pimjai Sudsawad. 2007. *Knowledge translation: introduction to models, strategies and measures*. Southwest Educational Development Laboratory, National Center for the ....

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.

Asanka Wasala, Ruvan Weerasinghe, and Kumudu Gamage. 2006. Sinhala grapheme-to-phoneme conversion and rules for schwa epenthesis. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 890–897, Sydney, Australia. Association for Computational Linguistics.

Warren Weaver. 1949/1955. Translation. In William N. Locke and A. Donald Boothe, editors, *Machine Translation of Languages*, pages 15–23. MIT Press, Cambridge, MA. Reprinted from a memorandum written by Weaver in 1949.

Ruvan Weerasinghe, Asanka Wasala, and Kumudu Gamage. 2005. A rule based syllabification algorithm for sinhala. In *International Conference on Natural Language Processing*, pages 438–449. Springer.