[1] B N V Narasimha Raju

[2] K. V. V. Satyanarayana

[2] M. S. V. S. Bhadri Raju

BiLSTMs and BPE for English to Telugu CLIR

Abstract: - A crucial component of Cross Lingual Information Retrieval (CLIR) is Neural Machine Translation (NMT). NMT performs a good job of transforming queries in the English language into Indian languages. This study focuses on the translation of English queries into Telugu. For translations, the NMT will make use of a parallel corpus. Due to a lack of resources in the Telugu language, it is exceedingly challenging to provide NMT with sizable parallel corpora. Thus, the NMT will encounter an issue known as Out of Vocabulary (OOV). Long Short-Term Memory (LSTM) with Byte Pair Encoding (BPE), which breaks up rare words into subwords and attempts to translate them to solve the OOV issue. Issues such as Named Entity Recognition (NER) continue to plague it. In sequence-to-sequence models, bidirectional LSTMs can solve certain NER challenges. Systems that need to be trained in both directions to recognize named entities can benefit from the use of Bidirectional LSTMs (BiLSTMs). The translation efficiency of NMT with BiLSTMs is significantly higher than normal LSTMs, as indicated by the accuracy metrics and Bilingual Evaluation Understudy (BLEU) score.

Keywords: Machine Translation, Bidirectional LSTMs, Cross-lingual IR, BPE, LSTMs, Neural Machine Translation, Beam search.

I. INTRODUCTION

CLIR utilizes a database that stores data in languages distinct from the user query language. If a user needs data in more than one language, it assists them in retrieving pertinent information in those languages. In this, the English queries are converted to Telugu queries. Machine Translation (MT) approaches are utilized to create these kinds of translations. The CLIR is crucial in nations resembling India since a sizable portion of the population still does not speak English fluently and is becoming more interested in learning their native languages. Based on a report published by KPMG in 2017 [1], it is projected that there will be a yearly growth rate of 18% in the number of native-language users of the internet in India. Therefore, MT is required to translate the query so that users can access more online content that was originally written in their native languages.

MT is mostly carried out in corpus-based translation. Corpus-based methods have demonstrated more improvement than direct translations. The most applied category in corpus-based translation is NMT. For training this model needs a parallel corpus. The NMT employs both machine learning and neural network models. Performance-wise, the NMT has outperformed the other models. This NMT approach will make use of BiLSTMs, which necessitate a sizable parallel corpus [2]. The parallel corpus is what determines how accurate is RNN translation [3]. Therefore, for improved translations, the standards of the parallel corpus should be preserved. In NMT, preprocessing is the first step which involves enhancing the quality of the data, encoding is the second step which involves training the model for use and decoding is the third step which involves translating the data. There aren't many parallel corpora available for languages like English-Telugu. The reason for the presence of inconsistencies and noise in these types of data sets is Telugu's rich morphology. Replications can also be present in the dataset. Preprocessing aids in the elimination of these issues.

Although NMT performs better, there are still issues with OOV [4] and NER in the translations. OOV issues leading to poor translations will be driven by small datasets with a high frequency of words [5]. If there are more often occurring terms in the original sentence when translating, the translation will be of higher quality. Otherwise, the

[1] Research Scholar, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, KLEF, Vaddeswaram-522502, India, E-mail: buddaraju.narasimharaju@gmail.com.

[2] Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, KLEF, Vaddeswaram-522502, India, E-mail: kopparti@kluniversity.in.

[3] Professor, Department of Computer Science and Engineering, S R K R Engineering College, Bhimavaram, India, E-mail: msramaraju@gmail.com.

translation will be of poor quality. These problems are common for morphologically rich and resource-poor languages. Although dictionaries have been used in the past to address OOV difficulties, there are still words that are not included in them [6, 7]. The BPE mechanism was employed to solve the OOV difficulties. Unknown words in BPE are broken down into smaller terms using word segmentation techniques, and these smaller words are then attempted to be translated. The BPE approach will effectively address certain out-of-specification (OOV) concerns in this manner.

The subsequent stages of NMT involve the utilization of encoding and decoding methodologies. The NMT architecture is comprised of a neural network with two distinct layers, specifically an encoding layer and a decoding layer [8]. According to the NMT encoder, the input consists of the source text, whereas the decoder yields the target text as the output. Another issue with NMT is NER, which is significant since it helps locate the specified things like named entities. It is beneficial to ascertain if a word in a sentence is a named entity or not. BiLSTMs are useful for identifying the named entities since they train models both forward and backward. Unidirectional LSTMs train in a forward direction; nevertheless, depending on the context that precedes a word, it may be possible to determine whether the word is a named object. Therefore, BiLSTMs will be more beneficial in achieving improved recognition performance for the named entities.

II. RELATED WORK

In previous instances, statistical machine translation approaches have been utilized as a valuable mechanism for the translation of sentences. P. F. Brown et al., [9] have used statistical machine translation techniques that incorporate both the language and translation modelling as well as the decoding to carry out translations. The translation produced by NMT surpasses that of Statistical MT. In order to translate from English to Arabic, Mai Oudah et al. [8] merged the NMT and Statistical MT. Although it still has issues translating shorter sentences. This problem has been solved by statistical MT and NMT together, yet tokenization presents some difficulties. Issues with tokenization are resolved at the preprocessing stage of the dataset. Preprocessing enhances the translation's quality and is a crucial step in NMT. Tokenization and the elimination of noisy content are two aspects of preprocessing data, the use of these techniques varies depending on the circumstances. Duygu Ataman et al. have utilized [10] a fixed-sized vocabulary. Semantic and syntactic losses are associated with conventional approaches. The acquisition of unsupervised morphological skills will result in a reduction of such losses. Anoop Kunchukuttan et al. [11] suggested using tokenization together with text normalization for the Hindi-English parallel dataset.

Kyunghyun Cho et al. [12] introduced the RNN encoder-decoder, which features a gated recursive convolutional neural network. The input is variable length, and the encoder extracts fixed-length representations from it. Translations are produced by decoding these representations. When sentences are shorter and contain fewer unfamiliar terms, these algorithms have better outcomes. However, in the context of the OOV problem, the efficiency of NMT may fall as the quantity of unfamiliar phrases grows. Despite the utilization of a BPE technique by B N V Narasimha Raju et al. [13] to tackle certain OOV concerns, there are still hurdles in translation production. BPE has been used for parallel corpora by Mattia A. Di Gangi et al. [14], especially for languages like English and German. They rely on the sentence's leftward context to determine the following word. It is insufficient in certain cases, and the efficiency is declining as well.

In previous times, the process of decoding was plagued by particular challenges due to the inadequate focus on decoders. The focus of Jiatao Gu et al. [15] has been on the challenge of decoding. They made use of a trainable decoding algorithm. To find a translation, this training is done on a decoding algorithm. They took advantage of a greedy decoder, which produced better translations with less overhead. During greedy decoding, the symbol with the largest conditional probability is selected at every node by following a path that is dependent on the conditional probability. Its performance has outperformed that of previous decoding methods.

III. BIDIRECTIONAL LSTMS

The method of Bidirectional LSTMs and BPE involves multiple phases, namely preprocess, BPE, encoder, and decoder, as depicted in Fig. 1. The encoding phase will employ BiLSTM network architectures, whereas the decoding phase will employ unidirectional LSTM architectures. In general, NMT relies on parallel data. More data is essential to produce more accurate translations. The parallel corpus of English and Telugu contains instances of replications, inconsistencies, and noise that are prevalent in languages with limited linguistic resources. The parallel corpus issues cause the performance of NMT to decrease. Preprocessing is used to fix these problems.

First, all characters are converted to lowercase during the preprocessing stage. Next, unwanted characters are eliminated, and so on. In order to eliminate duplication of information from the parallel corpus, it is important to transform it to the Unicode format, hence facilitating the effective elimination of duplicate entries within the corpus. In order to train the BiLSTMs, it is necessary to have a parallel dataset that includes pairs of English and Telugu languages. Sentences with a high frequency of words may result in OOV issues if resources are limited [16]–[18]. As a method of data compression, BPE [7], [13] is useful for merging byte pairs that appear frequently. The BPE mechanism may be helpful in word segmentation. Delimiters are employed to denote the termination of character sequences, while character vocabularies are utilized to populate symbol vocabularies. In order to return the original token, delimiters will be very important. It is crucial to count the symbol pairs in the corpus. An n-gram character is used to replace the most common symbol pairs found in the corpus. A singular symbol is generated through the amalgamation of frequently occurring n-grams. The initial and final sizes of BPE's vocabulary are identical. Fig. 2 presents the BPE Algorithm.

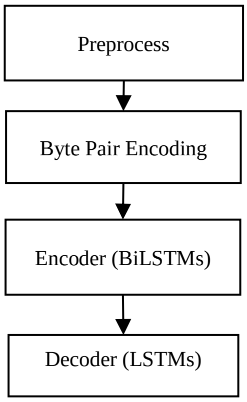

Figure 1. BPE and BiLSTMs in NMT

BPE Algorithm
Input: *C* – group of strings, *v* - magnitude of the vocabulary that is being targeted.
BPE (*C*, *v*)

- *Q* - distinct characters ϵ *C*
- The process should be repeated when the variable $|Q| < v$
  - Bigrams *a* and *b* are commonly found ϵ *C*
  - $z = a + b$
  - $Q = Q + [z]$
  - In *C*, every occurrence of *a* and *b* is substituted with *z*.
- Return *Q*

Figure 2. BPE Algorithm

For NMT, input data is provided to the encoder following the use of BPE. The lengths of the input as well as the output in a conventional NMT have the potential to vary. The preceding tokens in the sequence aid in predicting the subsequent token. Since the next token in this scenario depends on the leftward context, an NMT with a single unidirectional LSTM encoder is enough. The subsequent token in the other sequence learning task can be anticipated by considering both the preceding and succeeding context. Use the BiLSTMs [19] in the encoder in this case as the unidirectional LSTMs are insufficient for the function. Take the next two statements, for instance

- I intend to procure an orange from a shop due to its place as my preferred fruit.
- I intend to acquire an orange doll from the store as it is my preferred color for a toy.

Both sentences contain a token referred to as orange. The identification of the next token in a unidirectional LSTM model is based on the leftward context. Because of this, it might not be able to tell if the word orange will refer to a toy or a fruit. By utilizing both the contexts present in the leftward and rightward directions, a BiLSTM can accurately identify the following token.

Two unidirectional LSTM layers coupled in opposite directions serve as the encoder in BiLSTMs. The initial layer of LSTM receives input values a_1, a_2,..., a_n while the subsequent LSTM layer receives input values a_n, a_{n-1},... a_1. The output can be generated by combining the outputs of both LSTM models. Fig. 3 illustrates the architectural design of BiLTSMs.

LSTM assesses the interpretation $(\overrightarrow{d_y})$ of each word y in the left reference of a sentence. The equations (1)-(4) provide the assessment of $(\overrightarrow{d_y})$.

$$l_y = \sigma(W_{al}a_y + W_{dl}d_{y-1} + W_{sl}s_{y-1} + b_l) \quad (1)$$

$$s_y = (1 - l_y)\odot s_{y-1} + l_y \odot tanh(W_{as}a_y + W_{ds}d_{y-1} + b_s) \quad (2)$$

$$e_y = \sigma(W_{ae}a_y + W_{de}d_{y-1} + W_{se}s_{y-1} + b_e) \quad (3)$$

$$d_y = e_y \odot tanh(s_y) \quad (4)$$

The symbol σ denotes a function known as sigmoid, which carries out element-wise operations, while ⊙ symbolizes element-wise multiplication.

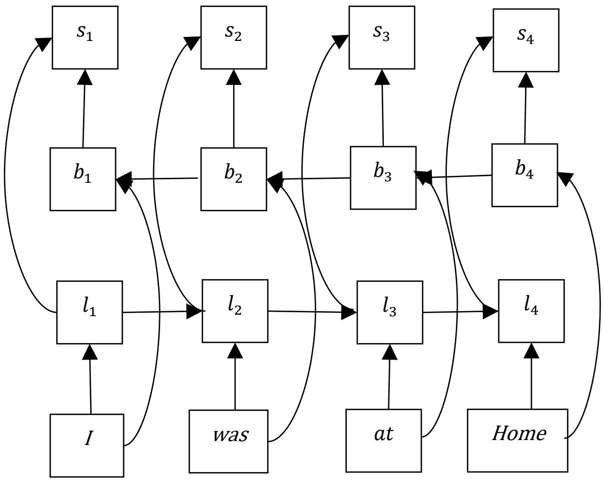

Figure 3. Architectural design of

Let us consider a sentence including n words, denoted as (a_1, a_2,..., a_n). In the given sequence, each input step (d_1, d_2,..., d_n) corresponds to a specific data point. The LSTM model assesses the interpretation $(\overrightarrow{d_y})$ of each word *y* in the left reference of a sentence. Typically, the second LSTM achieves the representation with the right context $(\overleftarrow{d_y})$ by reading the same sequence in a reverse manner. In essence, the forward LSTM is the first LSTM, and the backward LSTM is a second LSTM. The notation for the words in this model is d_y = $[(\overrightarrow{d_y}, \overleftarrow{d_y})]$. The aforementioned representations are advantageous in identifying the named entities within the provided context.

LSTMs are also employed as the decoder in NMT, and they produce translations for the input pattern [20]. Translations occur throughout the decoding process using beam search in combination with a unidirectional LSTM model. In beam search [21], the number of beams is controlled, the user-defined b-steps are maintained, and the system attempts to extend all feasible future steps. The beam search is in Fig. 4.

Beam search Algorithm
IS = Initial State
While *IS* is not equivalent to empty do

- Remove the most optimal node from the *IS*, denoted as *c*.
- If *c* represents the desired state, backtrack until the desired state is achieved and send the path.
- Produce and evaluate *c* successors, incorporate into *IS*, and list their parents.
- The value of | *IS* | is greater than *b*, where *b* represents the width of the beam, then select the best *b* nodes and exclude the rest from *IS*.

End

Figure 4. Algorithm for Beam Search

The model is evaluated using measures such as the BLEU score, cross-entropy perplexity and accuracy. Accuracy serves as an indicator of the level of correct classifications. On the other hand, the degree of perplexity indicates how well the probability model predicts a sample. In cross-entropy, the objective is to determine the loss function. The assessment of prediction accuracy is conducted by the utilization of the BLEU score.

IV. RESULTS AND DISCUSSIONS

There are no replications in the English-Telugu parallel dataset since the replications would involve translating a single source sentence more than once. Hence, the algorithm can generate precise translations for the original sentence by removing all instances of duplicate sentences from the corpus. In the process of acquiring new features from the replicated corpus, the system may experience confusion, perhaps resulting in overfitting of the model and suboptimal translation performance. Translations of higher quality will be produced from the parallel corpus if all these issues in the parallel datasets have been addressed.

Preprocessing eliminates duplicates, noisy data, and inconsistent data from the parallel corpus. The performance of NMT using BiLSTM and Unidirectional LSTM along with BPE is compared. Both models take their input from the English-Telugu parallel corpus. Unidirectional LSTM model is characterized by just one layer, in contrast to the dual-layer architecture observed in BiLSTMs. BiLSTMs will still feature an encoding process with two layers and a decoding mechanism with one layer. The LSTM layer in both models will be 500 in size, employ Adam as optimizer, and have a learning rate of 0.01. With the model decay and dropout rates set at 0.5 and 0.3, respectively, a total of 35000 training steps will be completed.

NMT's performance is assessed by contrasting it with BPE and other methods like BiLSTM and Unidirectional LSTM. Cross-entropy, perplexity, and accuracy are some of the metrics used to assess both the training and validation performance of BiLSTMs and unidirectional LSTMs. The values of these parameters are shown in Table 1. The performance of the NMT using BiLSTMs with BPE is improved.

Table 1. A Comparative Evaluation of Parameters for BiLSTM and Unidirectional LSTM

Parameters		**BiLSTM**	**Unidirectional LSTM**
Train	accuracy	95.37	90.46
	perplexity	1.16	1.43
	cross-entropy	0.15	0.36
Validation	accuracy	50.39	49.74
	perplexity	102.5	113.2
	cross-entropy	4.62	4.76

The graphs in Fig. 5 show the training graphs for BiLSTM and Unidirectional LSTM. The graph presented in Fig. 5(a) illustrates the comparison of training accuracy, the graph in Fig. 5(b) illustrates the perplexity, and the graph in Fig. 5(c) illustrates the cross-entropy comparisons between BiLSTM and Unidirectional LSTM. The unidirectional LSTM training accuracy is 90.46, whereas for BiLSTMS it is 95.37. The accuracy rates exhibit greater values is preferable. These findings indicate that BiLSTMS has superior performance. The training perplexity score of BiLSTM is 1.16, but the unidirectional LSTM has a value of 1.43. The model with a lower perplexity score is considered preferable. These findings indicate that BiLSTMS has superior performance. The training cross-entropy score of BiLSTM is 0.15, but the unidirectional LSTM has a value of 0.36. The model with a lower cross-entropy score is considered preferable. These findings indicate that BiLSTMS has superior performance.

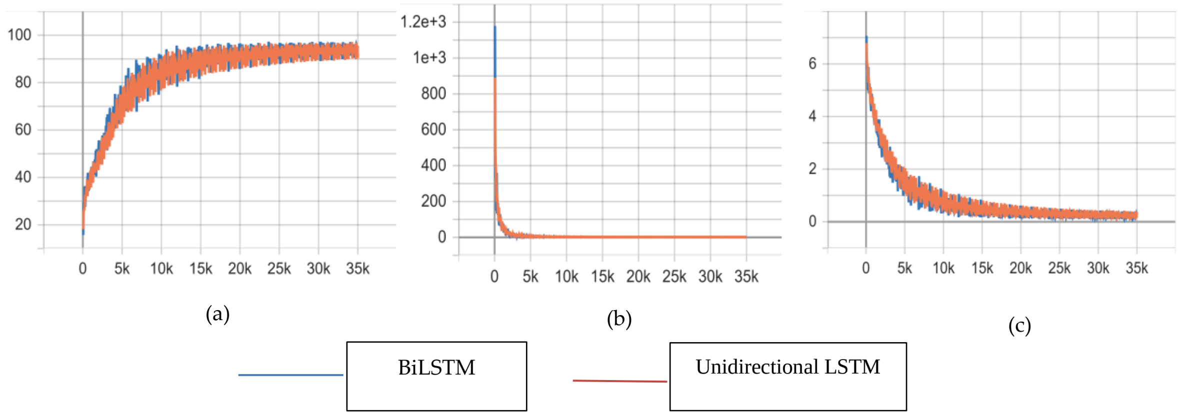

Figure 5. The training graphs for BiLSTMs and Unidirectional LSTM along with the BPE. They are: (a) Accuracy (b) Perplexity (c) Cross-entropy.

The graphs in Fig. 6 show the validation graphs for BiLSTMs and Unidirectional LSTM. The graph presented in Fig. 6(a) illustrates the validation accuracy, the graph in Fig. 6(b) illustrates the perplexity, and the graph in Fig. 6(c) illustrates the cross-entropy comparisons between BiLSTM and Unidirectional LSTM along BPE. In the unidirectional LSTM, the training accuracy is 49.74, whereas for BiLSTMS it is 50.39. The accuracy rates exhibit greater values is preferable. These indicate that BiLSTMS has superior performance. The validation perplexity score of BiLSTM is 102.5, but the unidirectional LSTM has a value of 113.2. The model with a lower perplexity score is considered preferable. These findings indicate that BiLSTMS with BPE has superior performance. The validation cross-entropy score of BiLSTM is 4.62, but the unidirectional LSTM has a value of 4.76. The model with a lower cross-entropy score is considered preferable. These findings indicate that BiLSTMS with BPE has superior performance.

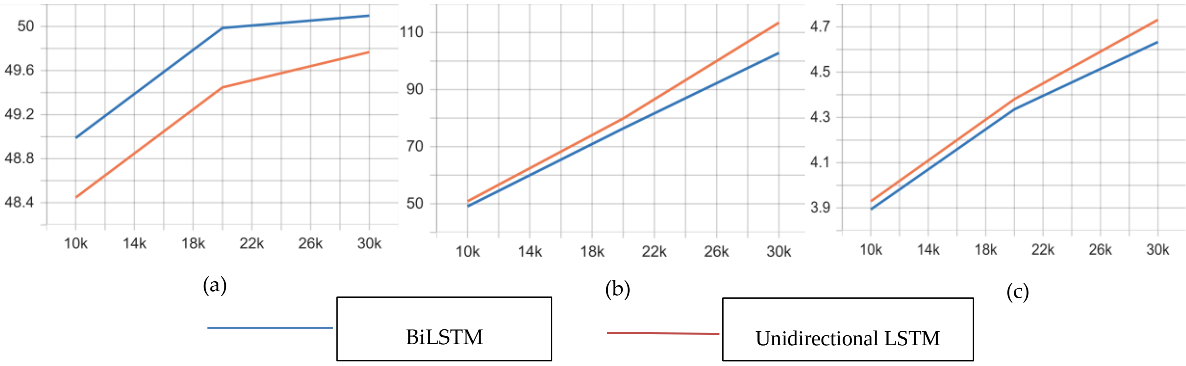

Figure 6. The validation graphs for BiLSTMs and Unidirectional LSTM along with the BPE. They are: (a) Accuracy (b) Perplexity (c) Cross-entropy.

The efficiency of both models is evaluated using a statistic known as the BLEU score. Table 2 displays the BLEU values. In the comparison of BiLSTMs and Unidirectional LSTMs, it is observed that the BLEU score indicates that the utilization of BiLSTMs with BPE in NMT results in translations that exhibit higher levels of accuracy. Preprocessing and BPE models are beneficial for solving OOV difficulties and enhancing the quality of parallel corpus, both of which assist NMT produce translations that are more accurate.

Table 2. A Comparative Evaluation using BLEU

Model	BLEU Score
BiLSTMs	19.58
Unidirectional LSTMs	18.76

The BLEU ratings of BiLSTM and Unidirectional LSTM along BPE are compared in Fig. 7. BiLSTM model has a value of 19.58, whereas the unidirectional LSTM model has a value of 18.76. Both models have BLEU scores. The model with a greater BLEU score is superior. These findings indicate that BiLSTM and BPE have superior performance.

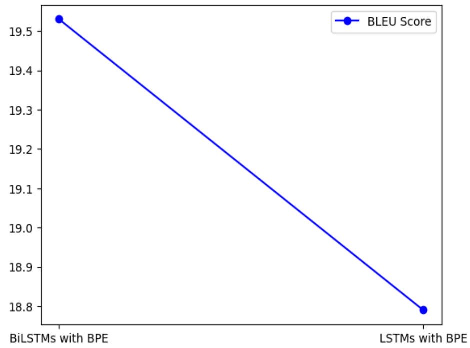

Figure 7. BLEU Score for BiLSTM and

V. CONCLUSION

In NMT, better translations can be produced if replications, noisy data, and inconsistencies are removed. Using BPE, OOV difficulties such as unfamiliar terms are resolved. The preprocessing stage of this work uses the English-Telugu parallel corpus, and the output of this method is used as input for the encoding stage. The model is tested using language pairs between Telugu and English. The metrics used to determine translation quality are cross-entropy, accuracy, perplexity and BLUE scores. When comparing the accuracy and translations of BiLSTMs models to Unidirectional LSTMs along with BPE, it was demonstrated through performance analysis that the former performs better. Thus, some OOV difficulties can be resolved by incorporating BPE into the model. Improved translations in languages with limited resources are the result of solving OOV issues. Some of the NER recognition issues have been resolved by the BiLSTMs in the NMT as opposed to employing Unidirectional LSTMs. Unidirectional LSTMs only rely on leftward context, whereas BiLSTMs use both leftward and rightward context in the sentence to identify NER. So, BiLSTMs outperform unidirectional LSTMs in terms of translation accuracy. Beam search will produce a higher-quality translation during the decoding phase. Because of this, the NMT that uses BiLSTMs and BPE produces better translations for the parallel corpus of English and Telugu. Considering this, NMT made up of BiLSTMs and BPE is recommended for the parallel corpus of English and Telugu. Therefore, BiLSTMs with BPE are useful for improving CLIR's translation accuracy.

REFERENCES

[1] KPMG, “Indian Languages - Defining India's Internet,” kpmg.com,https://kpmg.com/ky/en/home/insights_new/2017/04/indian-language-internet-users.html (Apr. 25, 2017).

[2] Karunesh Kumar Arora, Shyam S. Agrawal, “Pre-Processing of English-Hindi Corpus for Statistical Machine Translation,” Computación y Sistemas, pp. 725-737, 2017.

[3] Richard Kimera, Daniela N. Rim, Heeyoul Choi, “Building a Parallel Corpus and Training Translation Models Between Luganda and English,” Journal of KIISE, Vol. 49, No. 11, pp. 1009-1016, 2022.

[4] J. Xiong, "Treatment of OOV Words in Neural Machine Translation," 2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT), Jilin, China, pp. 1083-1085, 2023.

[5] Thi-Vinh Ngo, Thanh-Le Ha, Phuong-Thai Nguyen, and Le-Minh Nguyen. “Overcoming the Rare Word Problem for low-resource language pairs in Neural Machine Translation,” In Proceedings of the 6th Workshop on Asian Translation, pp. 207–214, 2019.

[6] Mengjiao Zhang and Jia Xu, “Byte-based Multilingual NMT for Endangered Languages,” In Proceedings of the 29th International Conference on Computational Linguistics, pp. 4407–4417, 2022.

[7] Rico Sennrich, Barry Haddow and Alexandra Birch, "Neural Machine Translation of Rare Words with Subword Units,"In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, August 7-12, pp. 1715-1725, 2016, DOI: 10.18653/v1/P16-1162.

[8] Mai Oudah, Amjad Almahairi and Nizar Habash, "The Impact of Preprocessing on Arabic-English Statistical and Neural Machine Translation,"ArXiv.org, Aug. 19-23, pp. 214-221, 2019, Web Link: https://www.aclweb.org/anthology/W19-6621.

[9] P. F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, et al., "A Statistical Approach to Machine Translation," Computational Linguistics pp. 79-85, Vol. 16, No. 2, 1990.

[10] Duygu Ataman, M. N., Marco Turchi, Marcello Federico, "Linguistically Motivated Vocabulary Reduction for Neural Machine Translation from Turkish to English," The 20th Annual Conference of the European Association for Machine Translation (EAMT), pp. 331-342, 2017.

[11] Anoop Kunchukuttan, P. M., Pushpak Bhattacharyya, "The IIT Bombay English-Hindi Parallel Corpus," European Language. Resources Association (ELRA), pp. 3473-3476, 2018, Web Link: https://www.aclweb.org/anthology/L18-1548.

[12] Kyunghyun Cho, B. V. M., Dzmitry Bahdanau, Yoshua Bengio, "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches,"Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, pp. 103-111, 2014.

[13] B. N. V. Narasimha Raju, M. S. V. S. Bhadri Raju, K. V. V. Satyanarayana, "Effective preprocessing based neural machine translation for English to Telugu cross-language information retrieval," IAES International Journal of Artificial Intelligence (IJ-AI), pp. 306-315, Vol. 10, No. 2, June 2021, DOI: 10.11591/ijai.v10.i2.pp306-315

[14] Mattia A. Di Gangi and Nicola Bertoldi and Marcello Federico, "FBK's Participation to the English-to-German News Translation Task of WMT 2017," Proceedings of the Conference on Machine Translation (WMT), Vol. 2, pp. 271-275, 2017, DOI: 10.18653/v1/W17-4723.

[15] Jiatao Gu, Kyunghyun Cho and Victor O.K. Li, "Trainable Greedy Decoding for Neural Machine Translation," Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1968–1978, September 7–11, 2017, DOI: https://doi.org/10.48550/arXiv.1702.02429.

[16] Ali Araabi, Christof Monz, and Vlad Niculae, "How Effective is Byte Pair Encoding for Out-Of-Vocabulary Words in Neural Machine Translation?," In Proceedings of the 15th biennial conference of the Association for Machine Translation in the Americas (Volume 1: Research Track), pp. 117–130, 2022.

[17] Aloka Fernando and Surangika Ranathunga, "Data Augmentation to Address Out of VocabularyProblem in Low Resource Sinhala English Neural Machine Translation," In Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation, pp. 61–70, 2021.

[18] Longtu Zhang and Mamoru Komachi, "Neural Machine Translation of Logographic Language Using Sub-character Level Information," In Proceedings of the Third Conference on Machine Translation: Research Papers, pp. 17–25, 2018.

[19] Martin Sundermeyer, Tamer Alkhouli, Joern Wuebker, and Hermann Ney, "Translation Modeling with Bidirectional Recurrent Neural Networks," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 14–25, October 25-29, 2014, DOI: 10.3115/v1/D14-1003

[20] Ilya Sutskever, Oriol Vinyals, Quoc V. Le, "Sequence to Sequence Learning with Neural Networks," NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems, Vol. 2, pp. 3104-3112, 2014, Web Link: https://dl.acm.org/doi/10.5555/2969033.2969173.

[21] Markus Freitag and Yaser Al-Onaizan, "Beam Search Strategies for Neural Machine Translation," Proceedings of the First Workshop on Neural Machine Translation, pp. 56–60, August 2017.