Attention based Neural Machine Translation for English-Tamil Corpus

Vijaay KU.[1], Mahesh N.[2], Karthikeyan B.[3], Rama S.[4]

[1]B.Tech, Computer Science Engineering, SRM Institute of Science and Technology, Chennai, India
[2]B.Tech, Computer Science Engineering, SRM Institute of Science and Technology, Chennai, India
[3]B.Tech, Computer Science Engineering, SRM Institute of Science and Technology, Chennai, India
[4]Professor, Dept. Of Computer Science Engineering, SRM Institute of Science and Technology, Chennai, India

---***---

Abstract - *Neural machine translation (NMT) is an approach to machine translation that uses an artificial neural network to estimate the probability for a chain of words.In this paper, we apply NMT for the English-Tamil language pair. We employed a NMT technique using Byte-Pair-Encoding (BPE) along with word embedding which overcomes the out of vocabulary problem (OOV) and in certain cases for idioms and phrases in Tamil which does not have a proper corpus with parallel English translations. We use the BLEU score for evaluating the system performance. Experimental results confirm that our translator has a BLEU score of 7.19.*

***Key Words*: Neural Machine Translation, LSTM, Out Of Vocabulary, Tamil, Translator, Attention.**

1. INTRODUCTION

English is one among the foremost widely spoken languages within the world. Hence vast majority of information present on the internet is in English. According to Wikipedia only 10% (1.25 crore) of Indian population(126 crore) speaks English , thus there is a need for these contents to be translated into regional languages in order to reach a larger audience. However translating such content manually is an arduous, expensive and laborious task. To eliminate these obstacles we need an automated, robust and simple system for translation. We aim to accomplish this using Neural Machine Translation (NMT).

In countries like China and India there exist region specific languages. For example, China is said to 297 living languages of which standard Chinese (Mandarin) is most popular. India has 22 official languages and 99 other languages. Out of the 1.3 billion of the Indian population, only 10% of them speak English [1]. And of this 10% just 2% are fluent and able to read, write and speak. The other 8% are only able to understand basic phrases and can speak broken up sentences in English with a plethora of accents that never ceases to amaze. The magnitude of information that is available on the internet is astounding, however the vast majority of this information is present in English. A significant population like those of India and China deserve access to the internet in a way that they can comprehend and thus it is necessary to translate these content into regional languages in order to enrich the people with knowledge. Such an undertaking will also help people in their professional and private conversations. For this purpose translation allows us to bridge the communication gap that exists among people. The amount of information that needs to translated is humungous and hence manual translation is not a feasible option and thus comes the need for machine translation. This paper involves the machine translation of English to a regional language, Tamil.

1.1 Differences in Languages

There are quite a few obstacles when it comes to machine translation associated with Tamil. Tamil is different from English in terms of word order and morphological complexity. English has a Subject-Verb-Object (SVO) structure whereas Tamil has a Subject-Object-Verb (SOV) structure [2]. Moreover, English is a fusional language whereas Tamil is agglutinative language. Another daunting issue is the lack of a corpus with English and Tamil sentences in parallel. This is due to the morphological differences in the languages. Developing translation models is difficult because of the syntactical differences between the two languages. Existing translation models fail to take into account the rich vocabulary of Tamil and indifferently end up transliterating a lot of words and phrases for which there are authentic translations available on the Tamil language.

1.2 Techniques

A lot of work is being carried out on the machine translation of various languages of the world including Indian languages but unlike foreign languages the machine translation of Indian languages are restricted to conventional techniques. Newer techniques such as word-embedding and Byte-Pair-Encoding (BPE) have shown considerable advances in neural machine translation and these concepts haven't yet been applied to Indian languages. Thus, in this paper, we perform neural machine translation technique with word embedding and BPE. English-Tamil language pair is one of the most difficult pair to translate due to morphological richness of Tamil language. We obtain the data from EnTamv2.0 and Opus and also create our customized vocabulary, and evaluate our result using widely used evaluation matric BLEU. Experimental results confirm that we got much better results than conventional machine translation techniques on Tamil language. We believe that our work can also be applied to other Indian language pairs too.

Main contributions of our work are as follows:

- We apply BPE with word embedding on Indian language pair (English- Tamil) with NMT technique.
- We achieve comparable accuracy with a simpler model in less training time rather then training on deep and complex neural network which requires much time to train.
- We have shown how and why data preprocessing is a crucial step in neural machine translation.

2. RELATED WORKS

The main challenges of machine translation are morphological and syntactical divergence. There are a number of techniques related to machine translation and the conventional one being rule based machine translation. Rule based machine translation requires rules representing regular sentence structure in both languages and an appropriate dictionary that maps a English word to a Tamil word. Rule based machine translation can be (i) direct mapping (ii) transfer based (iii) interlingua based.

The main advantage of rule based system is that it doesn't require any bilingual corpus. However forming rules and training a model for two languages is a laborious task. Approaches that involve the usage of bilingual data include (i) statistical machine translation (ii) example based machine translation (iii) neural machine translation. Statistical machine translations do not fit well for language pairs with different word order such as the English-Tamil pair. They are also prone to statistical anomalies. Example based machine translation makes use of tailored bilingual sentences that allows the model to learn phrases which it can identify and translate in the future. Creating a corpus that conforms to the requirements of this approach can be costly considering the existing lack of a gamut of bilingual data for the English-Tamil pair.

Neural machine translation is a modern approach to computer translation. Unlike the conventional phrase-based translation method, which consists of several small sub-components that are tuned separately, NMT attempts to construct and train a single, broad neural network that reads a sentence and delivers a correct translation using an artificial neural network to predict the probability of the phrase. NMT departs from phrase-based statistical approaches that use separately engineered sub-components. Its main departure is the use of vector representations for words (word-embedding) and internal states. The structure of the models is simpler than the phrase-based model. There is no separate language model, a translation model, and a reordering model, but just a single sequence model that predicts one term at a time.

Nevertheless, this sequence prediction is dependent on the entire source sentence and the entire target sequence that has already been generated. The word sequence modeling was usually done in a recurrent neural network (RNN). At first, word sequence modeling was typically performed using a recurrent neural network (RNN). A bidirectional RNN is the encoder and is in turn used by the neural network to encode the source sentence for a second recurrent neural network which is known as a decoder, and is used to predict words in the focus language. Furthermore Byte-Pair-Encoding is used before word embedding in order to produce better translations while keeping the model from getting complex. Out of Vocabulary (OOV) problem occurs when model encounters words which it had not previously seen in the training corpora.

3. SYSTEM DESCRIPTION

Seq2seq turns one sequence into another sequence. It does so by use of a recurrent neural network (RNN) or more often Long Short Term Memory (LSTM) or Gated Recurrent Units (GRU) to avoid the problem of vanishing gradient. The context for each element is the performance of the previous stage. The primary components are one network encoder and one network decoder. The encoder transforms each object into an effective secret vector containing the object and its meaning. The decoder reverses the cycle by converting the vector into an output object, using the previous output as an input reference.

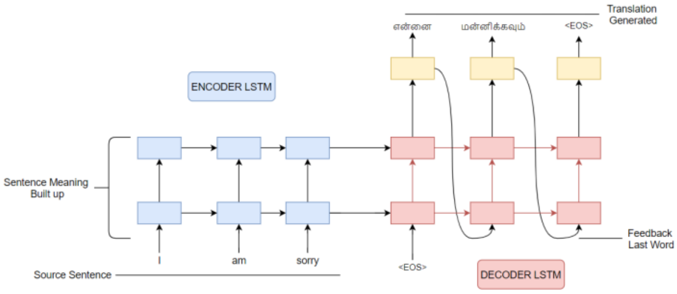

Fig -1: Seq2Seq Architecture for English-Tamil

The trouble with traditional seq2seq is that the only information that the decoder receives from the encoder is the last encoder hidden state. So, for short inputs this might work fine but for longer inputs it is unreasonable to expect the decoder to use just this one vector representation to output the translation. This might lead to forgetting information. Thus, we need to provide the decoder a vector representation from every encoder time step so that it can make well informed translations. For this purpose we utilize the attention model. Attention is the interface between the encoder and decoder that provides the decoder with information from every encoder hidden state. With attention, the model is able to selectively focus on the useful parts of the input sentence and hence, learn the alignment between them. This helps the model to effectively translate long input sentences. In Fig 2, h1,h2 etc are attention vectors generated by the encoder from the inputs x1,x2,x3. For each output time step context vector α is calculated using the concatenation of attention vectors. Using context vectors and hidden state st and previously predicted words(yt-1) the decoder generates the output yt.

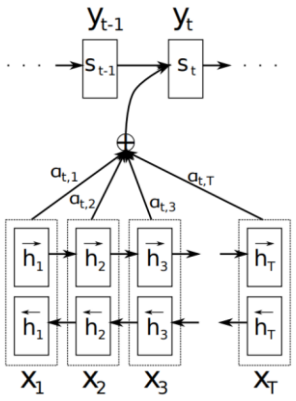
y_{t-1}
y_t
s_{t-1}
s_t
$a_{t,1}$
$a_{t,2}$
$a_{t,3}$
$a_{t,T}$
$\overrightarrow{h}_1$ $\overrightarrow{h}_2$ $\overrightarrow{h}_3$ $\overrightarrow{h}_T$
$\overleftarrow{h}_1$ $\overleftarrow{h}_2$ $\overleftarrow{h}_3$ $\overleftarrow{h}_T$
x_1 x_2 x_3 x_T

Fig -2: Attention Model

Word Embeddings encodes the relationship between words by vector representations of terms. The word vectors are similar to the sense of the word. In view of the OOV word and the expression in which it is written, language modeling is used to sequence words in a sentence and predict the meaning of a term in comparison with similar sentences. This is an elegant way to learn word definitions on the move. There are different pre-trained word embeddings such as word2vec, glove, fast text. These can also be used to create custom word embeddings for our data set. In our model we convert the English and Tamil words into a vector of length 500. We train the model with the same number of encoders and decoders in a layer.

BPE Example:

D1: She is an active dog. He is also active.

D2: Bruno is an active animal.

The dictionary generated can be a list of specific tokens (words) in the corpus. =[‘He’,’She’,’active’,’dog’,’Bruno’,’animal’]

Here, D=2, N=6

The count matrix N of shape 2 X 6 will be given as –

	He	She	active	dog	Bruno	animal
D1	1	1	2	1	0	0
D2	0	0	1	0	1	1

Byte pair encoding or diagram encoding is a simple type of data compression in which the most frequent pair of consecutive bytes of data is replaced by a byte that is not present within that data. A replacement table is needed to reconstruct the original data. The algorithm was first identified publicly by Philip Gage in the February 1994 article "New Algorithm for Data Compression." However in our model we use BPE for sub-words generation and model can translate new words based on the sub-words. We learn encodings from our source and target training data and then apply it for train,test and validation data.

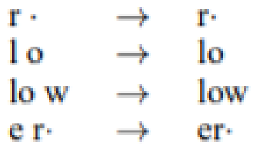

Fig -3: BPE merge operations learned from dictionary {‘low’, ‘lowest’, ‘newer’, ‘wider’}.

Above figure represents the learning of BPE from the vocabulary of 4 words {low, lowest, newer, wider }. At the test time, we first break the words into character sequences, then apply the learned operations to transform the characters into larger, established symbols. It refers to every word and allows for open vocabulary networks of defined vocabulary symbols. In our example, the OOV ‘lower’ would be segmented into ‘low er·’ where · is a special end of word character.

4. METHODOLOGY

Model architecture consists of bidirectional LSTM encoder and decoder each having two layers. Size of the LSTM is 500 which is same as the word embedding. We establish a vocabulary size of 50,000 words for both source and target languages. Optimization algorithm is Adam and the learning rate is 0.001. We employ aforementioned attention model with a dropout of 0.3. We trained on a GPU(Nvidia GeForce GTX 1060) which has increased the computation speed. Best version of our model was achieved after training for 7hrs.

BLEU (bilingual evaluation understudy) is a text quality assurance algorithm that has been machine-translated from one natural language to another. Quality is known to be the similarity between the output of a machine and that of a human being: "the closer a computer translation is to a skilled human translation, the better it is" – this is the idea driving BLEU.

4.1 Dataset

Data was taken from various sources[3]. Data consists of sentences used in various domains such as news, movie subtitles, text-books. Many of the sentences in the corpus have wrong translations , some of the entries have missing values and some sentences have multiple translations and certain words have their meanings explained instead of parallel translations and repetitions of the same sentence. Thus it is essential for us to clean the data set before using it to train our model. After pre-processing we have 231,899 lines of training data.

4.2 Performance

We have observed that the model performs well when sentences with a decent amount of contextual information are input. Translations are handy enough to use in casual conversation as well as official document. From test results we can infer that our model overcomes OOV problem in some cases.

5. FUTURE SCOPE

With the recent success of Text-to-Text Transfer Transformer we can experiment T5 on translation for Indian Languages. We can also leverage the pre-trained models of T5 and improve on it to make the translation much more effective.

We can enhance the NMT by creating larger corpus as there are very few open source English-Tamil parallel corpuses available online which can be used to effectively train the model. In addition, we can explore the possibility of using the above techniques for various translations of English into Indian.

We can extend this to a real time speech to speech translator, which can be utilized in many areas. This can be used to bridge the gap of human translators at heritage sites and in the parliament. This can also be incorporated as a text-to-speech translator for document reading and the blind thus bridging the gap in their disability.

We can unleash the potential of this NMT system by resolving other problems that exist in machine translations such as context awareness and identifying idioms and phrases.

This NMT system performs exceptionally with technical and complex documents and hence it can be developed into a proper document translator by implementing other required features like spell and grammar check etc. The model can also be deployed on the cloud for a faster response which can be accessed from anywhere.

6. CONCLUSION

In this paper we have applied NMT to the English-Tamil language pair, we have shown that NMT when applied with word embedding and byte-pair encoding performs well on said language pair and the results turn out satisfactory. Our model can be used for translation purposes in domains such as education. Further more, we can apply the same technique to various other language pairs for ease of translations.

REFERENCES

[1] https://web.archive.org/web/20191113211224/http://www.censusindia.gov.in/2011census/C-17.html

[2] Klein, Guillaume & Kim, Yoon & Deng, Yuntian & Crego, Josep & Senellart, Jean & Rush, Alexander. 2017. OpenNMT: Open-source Toolkit for Neural Machine Translation.

[3] Himanshu Choudhary, Aditya Kumar Pathak, Rajiv Ratan Saha, Ponnurangam Kumaraguru. 2018. Neural Machine Translation for English-Tamil. In Proceedings of the Third Conference on Machine Translation: Shared Task Papers.

[4] Ramasamy, Loganathan & Bojar, Ondřej & Zdenvek,. 2012. Morphological Processing for English-Tamil Statistical Machine Translation. 113-122.

[5] Mistry, J.G., Verma, A., & Bhattacharyya, P. 2017. Literature Survey: Study of Neural Machine Translation.

[6] B., Premjith & Kumar, M. & Kp, Soman. (2019). Neural Machine Translation System for English to Indian Language Translation Using MTIL Parallel Corpus: Special Issue on Natural Language Processing. Journal of Intelligent Systems. 28. 10.1515/jisys-2019-2510.

[7] Raj Nath Patel, Prakash B. Pimpale, M Sasikumar. 2018. Machine Translation in Indian Languages: Challenges and Resolution. arXiv preprint arXiv:1708.07950

[8] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. arXiv preprint arXiv:1409.0473

[9] Minh-Thang Luong, Hieu Pham, Christopher D. Mannin. 2015. Effective Approaches to Attention-based Neural Machine Translation. arXiv preprint arXiv:1508.04025

[10] Rico Sennrich, Barry Haddow, Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. arXiv preprint arXiv:1508.07909

[11] Krzysztof Wołk, Krzysztof Marasek. 2015. Neural-based machine translation for medical text domain. Based on European Medicines Agency leaflet texts. arXiv preprint arXiv:1509.08644