IRLAB-DAIICT@DravidianLangTech-EACL2021: Neural Machine Translation

Raj Prajapati
DA-IICT / Gandhinagar, India
prajapatiraj.97@gmail.com

Vedant Parikh
DA-IICT / Gandhinagar, India
vedant.parikh.6299@gmail.com

Prasenjit Majumder
DA-IICT / Gandhinagar, India
prasenjit.majumder@gmail.com

Abstract

This paper describes our team's submission of the EACL DravidianLangTech-2021's shared task on Machine Translation of Dravidian languages.We submitted our translations for English to Malayalam , Tamil , and Telugu. The submissions mainly focus on having adequate amount of data backed up by good pre-processing of it to produce quality translations,which includes some custom made rules to remove unnecessary sentences. We conducted several experiments on these models by tweaking the architecture, Byte Pair Encoding (BPE) and other hyperparameters.

1 Introduction

We participated in the shared task on Machine Translation in Dravidian languages DravidianLangTech, EACL 2021 .The advancement of technology has increased our internet usage and majority of the languages have acclimatised to the growing digital world. However, there are many regional languages which are under-resourced languages and still lack development.One such language family is the Dravidian languages , these languages are majorly spoken in south India ,Nepal, Pakistan, Sri Lanka and South Asia, we have submitted our translations for three language pairs namely:

1. English-Malayalam
2. English-Tamil
3. English-Telugu

Our implementations uses Transformer architecture and for that we have used OpenNMT-py (Klein et al., 2017) framework and BLEU (Papineni et al., 2002) score as the evaluation metric for our translation system.

Our main focus was on proper pre-processing of the data and often we have seen that improper preprocessing has led to horrendous translations. We have done extensive data pre-processing starting basic cleaning of punctuation symbols to language specific script normalization , apart from this we have added some custom rules as well. Which is followed by tokenization , truecasing and byte pair encoding (BPE).

For Indic languages espacially Dravidian langauges we often face the problem of Out of Vocabulary word (OOV) which is taken care by word segmentation using BPE ,so we deal with subwords instead of words.

This paper is arranged as follows : First we describe the task undertaken which is followed by in-depth explanation of the model architecture, then next we have described the experimental setup which includes provided data set information , preprocessing steps and clean data statistics. After that , we describe the experiments conducted on different language pairs and analysis of the results produced. At last we draw some conclusions and propose some future work.

2 Task Description

The task focuses on improvement to access and production of information for speakers of Dravidian languages. Due to low resources available , the research community has not developed much of an interest in this domain , the main focus of this task is to promote research in this area and build machine translation systems for native monolingual speakers of these group of languages.

In the era of digitization there is a large population who are not fully connected to the digital world because of their inability to access the digital world in their native language, which is what this task tries to accomplish.

The experiment setup contains the detailed information about our experiments,data and vision.

Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pages 262–265

3 Architecture

3.1 Encoder Decoder Frame work

Given two parallel sentences (a , b), the NMT model tries to learn the parameters θ by maximizing the probability P(b | a ; θ) . The Encoder generates a mapping from the input sentence to a hidden set of representations h and the decoder generates a target token b_t using the previously generated target tokens b_k where k<t and source representations h.Both encoder and decoder can be individually RNN/LSTM/GRU models as adopted by (Bahdanau et al., 2014) along with that self attention mechanism explained by (Vaswani et al., 2017) which is a vital combination for NMT systems.

3.2 Transformers

Introduction of Transformer models has increased the interests of researchers in NMT , transformers have preserved the idea of an encoder - decoder framework , with an addition of attention mechanism as explained in (Vaswani et al., 2017) it increases its worth . Transformer is one of its kind model which only uses self attention mechanisms to generate intermediate representation of input data . Transformers were initially tested on English-French dataset and were pretty successful achieving state of the art results.Unlike English-French language pair, Indian languages are a bit difficult to model because of certain reasons like richness in morphology, free word ordering. So more often we get poor translations.

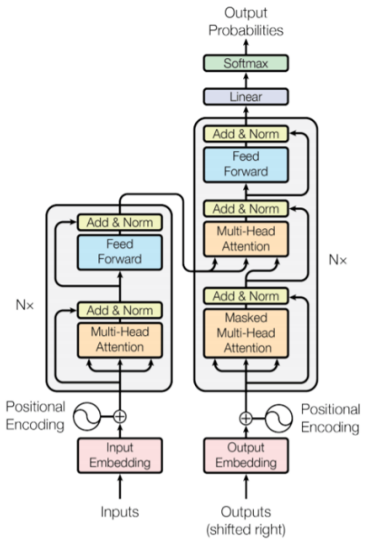

Figure 1: Transformer Architecture from (Vaswani et al., 2017)

Language Pair	No. of sentences
English-Malayalam	382K
Total parallel data	382K
English-Tamil	28K
Additional parallel data	170K
Total parallel data	198K
English-Telugu	23K
Additional parallel data	95K
Total parallel data	118K

Table 1: Raw Data Statistics

Figure 1 is the architecture used by almost every recent NMT paper, the biggest challenges in any NMT system are : Missing words , data sparsity . To overcome these challenges subword models were introduced to understand the subwords and how can we utilise them to increase our translation quality. Byte pair encoding is one way to compute subwords , initially introduced as a compression format but has been very efficient in word segmentation.

4 Experimental Setup

We have tried to perform experiments on three different language pairs as mentioned in the introduction section. Below mentioned is the detailed explanation of our approaches.

4.1 Datasets

Data is the key for any Neural Machine Translation system, this is something which is a driving factor.The language pairs are very resource scarce and the official training data (Chakravarthi et al., 2021) is not sufficient, so we took some additional parallel data as well.

Table 1 contains the data statistics for each language pair we have taken into consideration.

We have taken parallel data for language pairs from different sources. So for English-Tamil pair we have used[1] , similarly for English-Telugu we have used OPUS[2].

4.2 Preprocessing

Preprocessing is one of the main steps in any Machine translation system. In our experiment we have perform several steps which are listed below:

[1] http://ufal.mff.cuni.cz/~ramasamy/parallel/html/

[2] http://opus.nlpl.eu/

Language Pair	No. of sentences
English-Malayalam	380K
English-Tamil	169K
English-Telugu	110K

Table 2: Cleaned training data statistics

Language Pair	No. of sentences
English-Malayalam	2K
English-Tamil	1.5K
English-Telugu	1.3K

Table 3: Cleaned validation data statistics

- The sentences were normalized for punctuation by using Indic NLP Library[3].
- Some of the sentences consists numerical either only in source sentences or target sentences.We removed these sentences from dataset.
- We also removed sentences which contains of repetition of words.
- The words which were not of same language they where transliterated using Indic NLP's transliteration tool.
- Some specific special character(s) were also manually removed from the sentences.
- Many sentences were not aligned properly , so they were removed directly.
- In many sentences either source or target or both sentences were blank, which were also removed.

After all this prepossessing the final data statistics are explained pairwise in Tables 2 and 3 for training and validation.

4.3 BPE segmentation

An NMT system relies on mapping each word into a vector space, and we have a word vector corresponding to each word in a fixed vocabulary. The pertaining issues of data scarcity and inability of the system to learn high quality representations for rarely occuring words, (Sennrich et al., 2016) proposed to learn subwords and perform translation at a subword level.Subword segmentation is achieved using Byte Pair Encoding (BPE), by using BPE the vocabulary size is reduced drastically therefore we see a reduction in out-of-vocabulary words error, but it adds an overhead post processing step to convert the subwords back to the original word. We have used google's SentencePiece[4] to perform word segmentations using BPE in which we kept a uniform vocabulary size between 2K and 3K.

Configuration	Value
architecture	transformer
word embedding	512
Encoder depth	3
Decoder depth	3
transformer heads	4
size of FFN	2048
tranformer dropout attention	0.1
tranformer dropout FFN	0.1

Table 4: The main model configuration

4.4 Metrics

This work used BLEU (Papineni et al., 2002) score as evaluation metrics. A BLEU score compares a machine-translated sentence with the actual reference sentence by matching thier n-grams. The higher the number of n-grams matches, the closer are the two sentences.However, there are several implementations of BLEU available online, we have used multi-bleu[5] script from Mosesdecoder[6].

4.5 Modelling

For all the experiments we used OpenNMT-py Klein et al. (2017) toolkit. Table 4 describes the model configuration used in this experiment.

Table 5 describes the training parameters used by us to model data. We validate the model for every 5000 steps on BLEU and perplexity on validation set. We used 2000 as vocab size for English-Malayalam, English-Tamil, Tamil-Telugu and 2500 for English-Telugu language pairs.

[3] https://anoopkunchukuttan.github.io/indic_nlp_library

[4] https://github.com/google/sentencepiece

[5] https://github.com/marian-nmt/moses-scripts/blob/master/scripts/generic/multi-bleu.perl

[6] https://github.com/moses-smt/mosesdecoder

Parameters	Value
maximum sentence length	80
learning rate	0.0005
label-smoothing	0.1
optimizer	Adam
learning rate warmup	8000
training batch size	12800 tokens

Table 5: Training Parameters

Language Pair	BLEU Score
English-Malayalam	24.89
English-Tamil	7.00
English-Telugu	15.79

Table 6: Results on Validation data

5 Results

We made several models with different parameters and vocabulary sizes, Table 6 and Table 7 shows the results produced by the best models in each language pair for validation and test data respectively.

6 Conclusion and Future Works

In this paper we describe our submission to Machine Translation for Dravidian Languages (EACL 2021). As the quality of the sentences was not good, we had do a lot of preprocessing steps. So we also added other open source parallel corpora for our training. Our models are performing good on validation data but somewhat good on test data .

For future works, we would like to try pivoting methods and transfer learning methods. We would also like to introduce semantic features such Part of Speech Tags(POS), Named Entity Tags(NER), Lemmas etc. We can also use the language models for feature injection processes. Apart from this, we would also like to employ semi-supervised and unsupervised methods into these language pairs.

Language Pair	BLEU Score
English-Malayalam	8.43
English-Tamil	6.04
English-Telugu	6.25

Table 7: Results on Test data

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan V, Elizabeth Sherly, and John Philip McCrae. 2021. Findings of the shared task on Offensive Language Identification in Tamil, Malayalam, and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.