***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Neural Machine Translation from Bengali Language to English language and vice-versa

Author Name: Arindam Roy
Deptt of Computer Science, Assam University
Silchar, India
arindam_roy74@rediffmail.com

Abstract— Bengali ranks among the first ten spoken languages in the world with a native speaker numbering about 230 million people. With UNESCO declaring 21st February as International Mother Language Day to commemorate the laying down of lives by five Bangladeshi students for the cause of their mother tongue, Bengali has come into the radar of worldwide attention . Though significant amount of prose, poetry have been written in Bengali language and large number of newspapers in Bengali get published daily, technically it is still considered a Low Resource Language (LRL) unlike English or French which are High Resource Language (HRL). The reason is not far to seek as corpora in varied domains such as short stories, sports, politics, agriculture etc is less in number and even when they are available, the size is less. Machine translation (MT) is difficult to perform in Bengali as parallel corpora from Bengali to other languages and vice versa is few and far between and when they are available they suffer from the problems of size and quality. This work is aimed at implementing one state of the art model in Neural Machine Translation (NMT) which is called the self-attention transformer model to perform translation from English to Bengali and vice versa. Though a couple of research work has been published in the recent years on MT from English to Bengali, they are mostly domain specific. This paper does not focus on any specific domain for NMT from English to Bengali and as such may be conceived as a more of general domain NMT from English to Bengali which is more difficult than domain specific NMT. Performance evaluation of the model was done using BLEU version-4 vis-à-vis translations of well known English-Bengali MTsystems.

Keywords: Machine Translation , Neural Machine Translation, Self-Attention Transformer model, Low Resource Language, High Resource Language (HRL, BLEU version-4

I. Introduction

Neural Machine Translation (NMT) [1,2] is a comparatively newer method of Machine Translation which consists of neural networks which are trained to learn on vast amounts of parallel data. It has drawn considerable attention of the researchers because of its comparatively better performance compared to the earlier methods of MT. The widespread adoption of the NMT models stems from the fact that unlike the earlier long-dominant phrase based Statistical models, the neural models possess the novelty that the mapping from the input to the output is learned by training the model in a single big neural network. The NMT models give particularly encouraging results when large order parallel datasets are available which are of the size of millions of pairs of parallel sentences [3]. Of course NMT does not perform as well in the absence of large parallel datasets [4] Although compared to NMT, Statistical Machine Translation (SMT) performs better when large scale datasets are not available, still researchers want to construct NMT systems in such situations as well because NMT makes quantum jump in BLEU score as the data size increases while gains in SMT are at a linear rate only[4]

Paper [11] uses the self attention transformer model for translating English sentences to Bengali. Though in the recent couple of years some research work have been published on the use of transformer model for the job of translating English sentences to Bengali but they have mostly been trained on domain specific datasets and the evaluations, conclusions drawn also restricted itself to the specific domain. In our work we have not restricted ourselves to a specific domain because parallel Bengali to another language corpora and vice-versa are hard to come by and when they are available, they are small in size. Hence we have trained our model on a general purpose corpora using the self attention transformer mechanism.

Our experiments show that our model gives encouraging results over other models when the sentence size is small but when the sentence size gradually becomes larger and larger the performance degrades as is the case with other models. This may be due to the fact that as the sentence size increases, unique words also increase or the word polysemy may also increase.

II. Problem definition and Motivation

Over the last couple of years, various seminal research work has been conducted and proposed in the domain of Machine Translation by using the Neural Machine Translation models. The growth in the amount of research in the field of Neural Machine Translation has led to the newly proposed models being able to perform significantly better than the long-dominant Statistical Machine Translation, which used to to be the state-of-the-art even till the middle of the last decade.

The quality of the translations done by the NMT models is directly proportional to the size of the dataset which the model is trained on. Like other deep learning models, the NMT models require exceptionally huge datasets, consisting of millions of pairs of sentences in parallel form. However, time, money and skilled professionals are required to create such high quality datasets and also create a gold standard pair of parallel text which makes such high-quality data sets very expensive.

So in the absence of a large quantity and high-quality datasets most of the models being developed in low resource Indian languages is based on domain specific datasets. We were motivated to exploring how the self attention transformer model woks on a general purpose dataset and not on any domain specific dataset

III. LITERATURE SURVEY

With the increasingly promising results of NMT systems on large sized parallel corpora, research works on MT gradually shifted towards NMT and of late to further increase the success rate of NMT, hybrid models like phrase based SMT in combina-tion with NMT have also been reported.

The research works [1,2] underpinned the development of NMT systems. In these works they had proposed a Neural Network Language Model (NNLM). NNLM is a statistical Language Model where the a priori probability of a word sequence being the correct system response is estimated by a feed forward neural network.

Paper [4] extended the idea in [1,2] by proposing a statistical standard N-gram language model to be used in combination with NNLM.

When Recurrent Neural network (RNN) [6] was proposed it heralded a paradigm shift in the sense that the RNN was underpinned by continuous representations for words, phrases and sentences and did not rely on align-ments of phrasal translation units as was the case with phrase based SMT.

A hybrid RNN-SMT model proposed in [5] showed that performance of SMT systems improve considerably when a sequence of symbols was encoded into a fixed length vector representation and the encoder & decoder of RNN is used to maximize the probability of a target sequence.

Whenever large labeled training sets are available, Deep Learning Network (DNN) work well but they can only be used when inputs and targets can be encoded with vectors of fixed dimensionality. This is a serious drawback which Sutskever et al. [6] tackled by proposing a model where dependence on the sequence structure is minimal. Towards this end they proposed a model called Long Short Term Memory (LSTM) which is essentially a RNN language model except that it is conditioned on the input sequence.

Cho et al.[7] reported that RNN Encoder-Decoder coupled with recursive convolutional Network performs relatively well on short sentences without Out of Vocabu-lary (OOV) words but the performance rapidly declined with the increase in the length of the sentences

The problem of degrading quality of the translation of source sentences with the increase in the length of source sentence as suggested in [7] was also noticed in paper[8] who proposed new architecture that sought to address the issue. It proposed a model which soft-searched for a set of input words or their annota-tions computed by an encoder for generating the target word. So it freed the model to encode a whole sentence into a fixed-length vector and allowed the model to focus on information relevant to the next target word. They found that their conjecture yielded a positive result in the sense that NMT systems built by them based on their conjecture produced better results.

Again a paradigm shift happened in the world of NMT when [9] pro-posed a very novel architecture called the Transformer based solely on attention mechanism which fully discarded the ideas of RNN & CNN. They reported that the architecture proposed by them was superior in quality by virtue of the fact that it was more parallelizable and required significantly less time to train. The model proposed by them achieved 28.4 BLEU on the WMT 2014 English to German trans-lation task which was an improvement by over 2 BLEU points. The model achieved a brilliant BLEU score of 41.8 on the WMT 2014 English to French translation task which set a new record in this field. The model was trained in 3.5 days on 8 GPUs which is only a small fraction of the time required by best models reported at that time in literature.

NMT systems for English (En) to Bengali (Bn) fed with synthetic data was developed in [10]. To overcome the problem of data sparsity which takes place when a morphologically rich language like Bengali is used, various techniques like developing a word breaker for Bengali, synthetic data etc were used.

Monolingual data was incorporated in the source or target side in case of NMT in low resource scenario. Though it made output fluent but the drawback was the lack of accurate output. Back-translated monolingual data was used to expand the training data for low resource NMT. Here the available training data is initially used to train the model and only then the monolingual corpus is passed for translation. The size of the original parallel data has a direct bearing on the translation of the monolingual data. Increasing the size of training data through parallel phrases acquired from the original training data using SMT system was proposed in [12].

.

IV. Encoder and Decoder structure of transformer modelSelecting a Template

The Self Attention based transformer model was first proposed in[10]. This model has been used for English to Bengali MT task in the present paper.

Transformer neural network architecture consists of an encoder and decoder & its main function is sequence to sequence modeling. In the context of NLP a sequence can be ordered set of words to form sentences and so the transformer neural network architecture can do NLP tasks like English to other language translation and vice versa. The Transformer contains a stack of encoding components and a stack of decoding components whose numbers are the same.

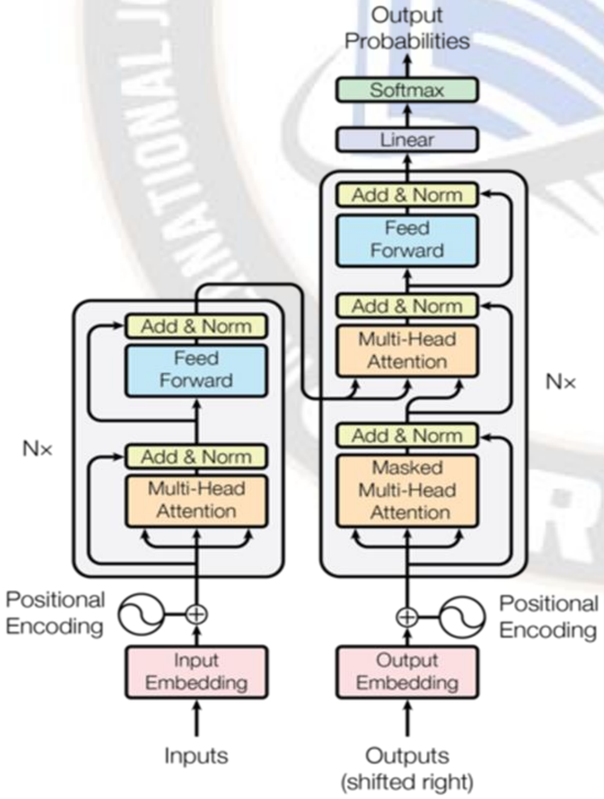

Figure 1 : Transformer Encoder-Decoder architecture, taken from Vaswani et al. (2017).

As far as structure is concerned , the encoders are all identical but they don't share same weights. There can be any number of encoders and decoders in the transformer model as indicated by Nx in Fig.1 but the number is normally kept between 6-8 in order to minimize complexity. Each encoder is made up of two sub Layers- i) Self Attention and ii) Feed Forward Neural Network.

The input to the encoder is first fed through the Self attention layer which while encoding a particular word looks at other words of a sentence. The output produced by the Self Attention layer at each position of a sentence is then fed to exactly same Feed Forward Neural Network. The decoder has both those layers as mentioned above, but there is an Encoder-Decoder attention layer in between them to focus on relevant parts of the input sentence. This is shown in Figure 2.

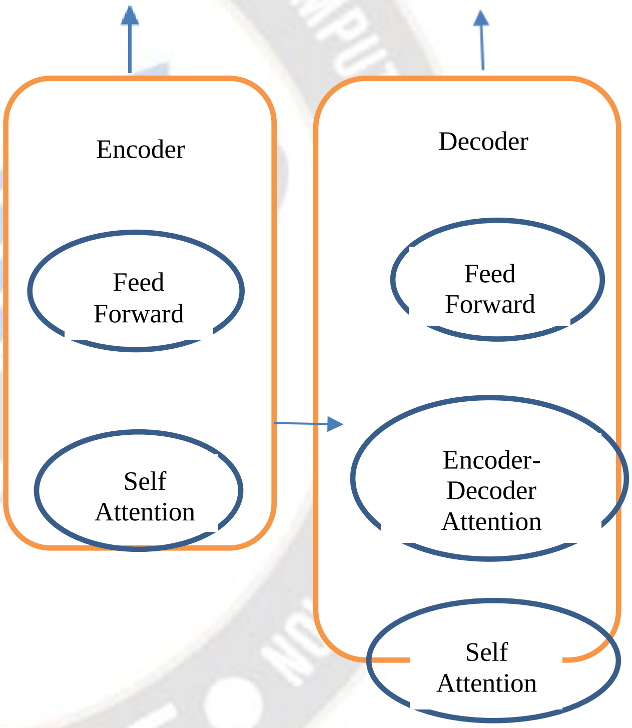

Figure 2: Encoder and Decoder sub layers in a Transformer Model.

Self-attention, Multi-head attention, Positional encoding besides Encoder-Decoder architecture forms the key components of the transformer model for MT. We now proceed to briefly discuss Self-attention, Multi-head attention, and Posi-tional encoding .

A. Self Attention

A Self Attention module is fed with n inputs and it produces n outputs. In a self attention module the inputs interact with each other and they find out a way of who they should pay more attention to. It is required in long term dependencies. Self attention can tell us which part of a sequence to look at while generating a specific output. So if we are doing an MT task, while generating a particular word, we can decide which part of the input sequence needs to be focused on.

Let us consider two input sequences we want to translate:-

i. The animal didn't cross the street as it was too tired.

ii. The animal didn't cross the street as it was very wide.

Here 'it' refers to animal in i) and 'it' refers to street in ii). It is easy for a human being to make this distinction but it is not that easy for a model.

In case of a transformer with Self Attention mechanism , when it processes each word, self attention allows it to look at other positions in the input sequence to help get a better encoding/decoding.

Self Attention module creates three vectors from encoder's input vector (X_i) for computing self attention.

These are:-

- Query vector (q_i)
- Key vector (k_i)
- Value vector (v_i)

These are created by multiplying the input with weight matrices W^Q, W^k, W^V learned during training. Then the scores of all words of input sentences are computed by taking dot product of query vector with key vector of respective words. The scores are then divided by the square root of the size of key vector. This is called scaled dot product attention.

Softmax function is then used to get a distribution of attention of each of the words in the sequence w.r.t the word under consideration.

Then each Value vector is by the score obtained by applying the Softmax function. This is done to keep the value(s) of words we want to focus on and keep the focus out of irrelevant words.

Finally the weighted value vectors are summed up which produces the output of self attention layer at a particular position which is called Vi (i stands for a particular position in the sequence.)

The mathematical formula proposed by Vaswani et al.[9] for calculating Self-attention is thus:-

Attention (Q, K, V) = $(Q/ \surd (d_{\_k}) * K^T) V$ where

where Q, K, V are the matrices containing the sets of queries, keys and values respectively and $1/(\surd d_k)$ is the scaling factor.

B. Multi Head Attention

Multi head attention are attention mechanisms in multiple layers of encoders and decoders. They are calculated exactly the same way as described in 3.1 with the only notable difference being that the learnable matrices W^Q, W^K and W^V being different in different layers. The advantage of this mechanism is that it expands the model's ability to focus on different positions. Different W^Q, W^K and W^V helps the model focus on different parts of the sentence which further enriches the processing of the current word.

C. Positional Encoding

Unlike CNN and RNN encoders, Self-attention based Transformer outputs do not depend on order of inputs. But order of input sequence conveys important information for MT tasks and language modeling in the sense that in which position a particular word is in a sentence imputes more meaning to it. The final input embedding is a concatenation of learnable embedding and positional embedding which becomes input to self attention layer. Therefore the role of positional encoding is to incorporate in the input embedding some value of where in the input sequence a specific input currently being processed is.

D. Feed-forward network

Each of the layers of encoder and decoder contain a fully connected position-wise feed-forward network. Position-wise feed-forward variant of feed-forward network is used to find out the position of a word in a sequence. ReLU activation function is used in the network.

E. A Holistic Transformer Model

At the end of discussion of the constituents of the Transformer model let us form an idea of the Transformer model as a whole in the light of Fig.1. There we can see that a concatenation of input embedding and positional embedding is first fed into the Multi head attention sub layer. The output of the Multi head attention layer is then added to the concatenation of input embedding and positional embedding which is passed via a split connection as shown in Fig.1 and then normalized. The normalized output from the Multi head attention layer is then fed to the Feed Forward Neural Network (FFNN) and this FFNN exists for every word of a sentence. The output of the FFNN is then supplied as input to the Multi head attention layer in the decoder side.

Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

In the decoder side of Fig.1 we can see there is one significant difference with the encoder side which is Masked Multi head attention layer is incorporated besides Multi head attention layer.

Now questions may arise why this is done. It is done because the architecture should not see any word beyond the currently processed word as the hypothesis is that the current word depends only on its immediate past and not on the future words but the self-attention mechanism allows to look at words before and beyond while processing the current word. So, for example, say we are doing an MT task from English to Bengali and we are processing the 3rd word of a sentence. We can use attention on the first two words but attention is forbidden on the remaining words because the model is not supposed to know them while predicting the 3rd word. This is done by masking the first Multi head attention layer in the decoder side and then the outputs are passed as it is to other multi head attention layers.

V. Dataset and experimental setup

A. *Dataset*

The dataset taken for this experiment is Shahjalal University Parallel Bengali-English corpus (SUPara) [13]. The dataset consists of approximately 70k pairs of sentences. The corpus consists of sentences from different domains including administrative, journalistic, and external communication texts.

The original dataset consists of two (.txt) files, one each for the source and the target language. In the scope of this work both the files have been merged into a single (.csv) file for better productivity in terms of output as a software system.

Among the decisions taken during training the model, the most basic one is that for better use of the allocated GPU, the dataset is trimmed down to sentences of length less than eighteen (18). The vocabulary of the English (source) language and that of Bengali (target) language is given in the Table- 1 below:

Language	Vocabulary Size
English	27163
Bengali	55418

TABLE-1:Vocabulary sizes of the source and target language.

The dataset is divided into three sets, namely, the training set, the validation set, and the testing set. The training set is used for training the transformer model; the test set is used to carry out the test to assess the performance of the trained network and the validation set is used to implement early stopping to get a better generalization of the model.

The sizes of the training, test and validation datasets are presented in Table - 2 below:

Dataset	No. of sentences
Training Set	52019
Testing Set	540
Validation Set	1078

TABLE -2 : Number of sentences in the training, testing and validation set.

B. *Experimental Setup:*

The experimental setup to run the self-attention model is the following:

The Dataset was hosted in Google Drive. The model was implemented in Google Colab environment which provides the GPU support needed to run such a high computation model like that of the transformer model. The setup used in the scope of this paper uses around 12GB of GPU, along with 12 GB of RAM; both of which are hosted remotely in Google Collab servers. The model is built using the Pytorch deep learning framework.

C. *Training Hyperparameters*

The hyper-parameters of the model used during training are as follows: Word embedding dimension is taken as 720, with multi-head attention dimension being 8. Three layers are stacked in the encoder and decoder. The Feed-Forward hidden dimension is 2048, and the batch size is 108.

VI. Results and Analysis

Bilingual Evaluation Understudy (BLEU) score is used to analyses the performance of the different models. BLEU compares the n grams of the predicted translation with the target translation to count the score.

For better analysis, we calculate and plot the BLEU unigram, bigram, trigram along with the standard BLEU 4-grams.

On the test set, the model achieves a BLEU (version 4) score of 10.52 . The graph below shows the BLEU score (unigram, bigram, trigram and 4-gram) achieved on the overall test set of 50 sentences:

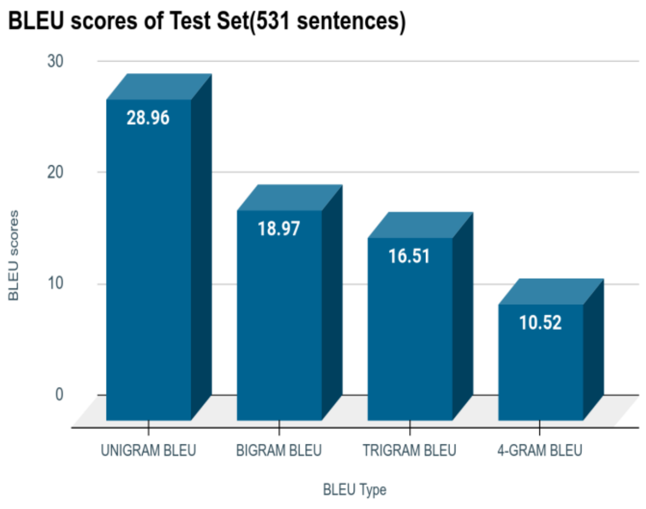

Figure 3: BLEU score achieved on a test set of 531 sentences

For further analysis of the model with respect to different length of the sentences, we divide the test set into subsets based on the length of the sentences in the subset. The sentence length of the subsets and the performance of the model on the respective subsets is given in the table-3 below.

We have calculated and documented the unigram, bigram and trigram BLEU scores along with the standard BLEU version-4 (4-gram).

Sub set	Length of sentence	No of sentences in subset	Unigram BLEU	Bigram BLEU	Trigram BLEU	4-gram BLEU
1	<5	86	48.81	40.34	36.78	27.43
2	<7	157	40.12	31.86	29.55	22.03
3	<9	237	36.70	27.29	24.49	17.01
4	<11	323	33.71	23.81	20.89	14.00
5	<15	454	31.83	21.65	19.11	12.71

TABLE -3: BLEU scores of subsets of Test set, when test set is divided based on length of the sentences

Based on the BLEU scores on the different test subsets, it can be concluded the transformer model trained on a low resource scenario is able to achieve good performance on relatively smaller sentence translation tasks, but, as the length of the sentences increases, the translation quality starts deteriorating

The performance of the model with varying length of the sentences is plotted in the graph below. The BLEU 4-gram values are considered as the performance gold standard.

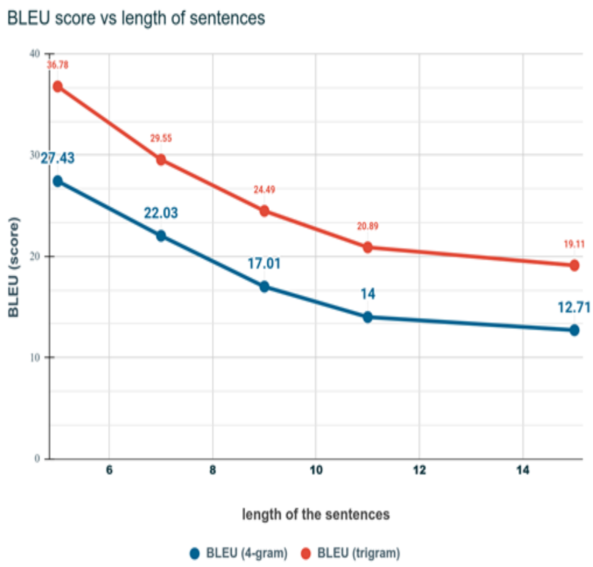

Figure 4: BLEU score vs length of the sentences

A. *Comparative Analysis*

In this section, the scores of the model presented in this paper are compared with some well-established and well-explored models

System reference	Type	BLEU score (best score)
Islam et al. (2010)	SMT	11.4
Pal et al. (2014)	SMT	13.17
Dandapat and Lewis (2018)	SMT,NMT	9.80
Banerjee et al. (2018)	PB-SMT	11.34
M. Akter et al. (2020)	PB-SMT, NMT	27.46
Our model	NMT	27.43

Comparing the results obtained in our experiment with the pre-existing models, we can conclude that the transformer model in our experiments not only achieves comparable scores on the overall test set, but significantly outperforms the

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

scores of the models, for the test subsets containing smaller sentences.

The BLEU score achieved on test subsets of small sentences, and the scores achieved on the test subsets of medium length sentences, outperforms the scores achieved by the well known SMT models.

However, the overall score achieved in [17], is the highest using a fairly large corpora of nearly 4.8 lakh sentences (0.5 million approx.) and also it is a hybrid MT scheme consisting of NMT injected with Phrase based SMT. But it may be seen that our model using only self attention transformer based NMT performs appreciably well when compared to the hybrid model proposed in [17].

Qualitative Analysis:

Transformer models have achieved state-of-the-art translations on benchmark datasets and thus have been widely adopted in various research as well as implemented on large scale by tech giants. In this section, some translations of sentences fed dynamically to the model are compared with the state of the art Google translation which uses a more sophisticated model trained on huge datasets.

Some of the examples of translation on sentences fed dynamically to the model are as below:

Example 1:
Sentence: Where are you coming from ?
Model prediction: আপনি কোথা থেকে এসেছেন?
Google Translation: তুমি কোথা থেকে আসছ

Example 2:
Sentence: Where are you going ?
Model prediction: তুই কোথায় যাবি?
Google Translation: আপনি কোথায় যাচ্ছেন?

Example 3:
Sentence:No one can prosper without diligence .
Model prediction: পরিশ্রম না করলে কেহ উন্নতি করতে পারে না।
Google Translation: পরিশ্রম ছাড়া কেউ উন্নতি করতে পারে না।

Example 4:
Sentence: With great power comes great responsibility .
Model prediction: বড় ক্ষমতা নিয়ে আসে ব্যাপক দায়িত্ব।
Google Translation: মহান শক্তি দিয়ে মহান দায়িত্ব আসে

VII. Conclusion and Future works

Neural Machine Translation has become the byword for automated machine translation with large-scale data, part of this wide use of NMT can be attributed to the efficiency of the self-attention based Transformer model. In this paper we have implemented the state of the art self-attention transformer architecture to translate English sentences to Bengali and vice versa in a low resource scenario and also, at length, dwelt on its performance, It is to be noted that the model has been implemented on a general dataset and not domain specific parallel corpora. Finally, BLEU score has been calculated for the model and a comparison has been carried out with other well known neural translation models.

On performing the qualitative analysis of the translations produced by the model and comparing the translations with the translations made by Google Translate, we have found that the model performs well in translating short length sentences from morphologically weak English language to a morphologically stronger Bengali language.

On analyzing the dataset [13] it has been observed that the dataset contains a lot of noises in terms of same words having both American and British spellings. Also, in case of Bengali, many of the words have different renditions in conformity with different dialects of Bengali language. By tuning the datasets to handle such noises, better performance can be achieved.

The model can be made to perform better by training the model on different pre-trained word embedding methods like word2vec, Glove and Byte Pair Encoding. Data augmentation may be used to augment the low resource dataset.

VIII. References:

[1] Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. "A neural probabilistic language model, Journal of machine learning research" vol 3 , 2003 pp 1137–1155.

[2] Bengio, Y., Schwenk, H., Senécal, J.-S., Morin, F., "Neural probabilistic language models. In Innovations in Machine Learning: Theory and Applications", 2006, pp 137–186

[3] Bahdanau, D., Cho, K., & Bengio, Y., "Neural machine translation by jointly learning to align and translate", ICLR, 2015

[4] Zamora-Martinez, F., Castro-Bleda, M. J., & Schwenk, H, "N-gram-based machine translation enhanced with neural networks for the French-English BTEC-IWSLT'10 task". In International Workshop on Spoken Language Translation (IWSLT)" , 2010, pp 45-52

[5] Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio Y, "Learning phrase representations using RNN encoder–decoder for statistical machine translation", In Proceedings of the Conference on Empirical Methods in Natural Language

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Processing (EMNLP) Doha, Qatar, 2014a, pp. 1724–1734

[6] Kalchbrenner, N., & Blunsom, P, "Recurrent continuous translation models. In Proceedings of the Conference on Empirical Methods in Natural Language Processing", 2013, pp. 1700–1709,

[7] E Sutskever, I., Vinyals, O., & Le, Q. V, "Sequence to sequence learning with neural networks", In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 27, 2014, pp. 3104–3112.

[8] Cho, K., van Merrienboer, B., Bahdanau, D., & Bengio, Y, "On the properties of neural machine translation: Encoder–decoder approaches", In Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar: Association for Computational Linguistics, 2014a, pp. 103–111

Article in a journal:

[9] Bahdanau, D., Cho, K., & Bengio, Y, "Neural machine translation by jointly learning to align and translate", ICLR, 2015

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, "Attention is all you need", . In I. Guyon, 2017

[11] Dandapat, S., & Lewis, W, "Training deployable general domain MT for a low resource language pair: English–Bangla", ACL anthology, 2018, pp 129-138.

[12] Sen. S, Hasanuzzaman. M, Ekbal. A, Bhattacharyya. P, and Way.A, "Neural Machine Translation of Low-resource

Languages using SMT Phrase Pair Injection", Neural Machine Translation of Low-resource Languages using SMT Phrase Pair Injection, 2018, pp 1-27

[13] Mumin et al. "SUPara: A Balanced English-Bengali Parallel Corpus", 2012

[14] Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., & Makhoul, J. , "Fast and robust neural network joint models for statistical machine translation. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics", vol 1, 2014, pp. 1370–1380

[15] Islam, M. Z., Tiedemann, J., & Eisele, A, "English to Bangla phrase-based machine translation", In Proceedings of the 14th Annual conference of the European Association for Machine Translation, May,2010

[16] Le, H.-S., Allauzen, A., & Yvon, F., "Continuous space translation models with neural networks", In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Montréal, Canada: Association for Computational Linguistics ,2012, pp 39–48.

[17] M. Akter, M. Shahidur Rahman, M. Z. Iqbal, and M. R. Selim, "English to Bangla machine translation ",Journal of Computer Science, volume 16 No 8, 2020, pp 1128-1138.

[18] Pal, S., Naskar, S. K., & Bandyopadhyay, S., "Word Alignment-Based Reordering of Source Chunks in PB-SMT", In LREC, May, 2014, pp 3565-3571

[19] Papineni, K., S. Roukos, T. Ward and W.J. Zhu, "Papine "Bleu: A method for automatic evaluation of machine translation", Proceedings of the 40th Annual Meeting of ACL, 2002, pp 311-318

[20] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A." Automatic Differentiation in Pytorch." 31st Conference on Neural Information Processing Systems (NIPS 2017), 4-9 December 2017, Long Beach, CA, USA pp 1-4