IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 26, Issue 5, Ser. 1 (Sept. – Oct. 2024), PP 17-25
www.iosrjournals.org

Neural Machine Translation Between English And Bangla: A Philosophical Survey Observing Architectures And Performances

MD. Abdul Awal Ansary
Department Of Computer Science And Engineering
RTM Al-Kabir Technical University, Bangladesh

Abstract:
Neural Machine Translation (NMT) is a state of the art of machine translations that uses neural network models to translate text from one language to another. Unlike traditional machine translation methods, which rely on statistical models and rule-based systems, NMT leverages the power of deep learning, specifically using techniques such as recurrent neural networks (RNNs), long short-term memory networks (LSTMs), or more recently, transformers. Neural Machine Translation represents a significant dive forward in the field of automatic translation, leveraging the latest advancements in deep learning to deliver more accurate and natural translations. The progress in NMT has been remarkable, driven by advancements in model architectures, multilingual training, pretraining techniques, increased computational power, and the availability of vast multilingual datasets and improved evaluation metrics. These developments have collectively enhanced the quality, efficiency, and applicability of NMT systems across various languages and domains. In this article we will explore the technologies and performances of NMT between English and Bangla Language pair from its launching periods to present.

Keywords: *Machine translation**,** Neural network, Architectures of neural network, Data set, Bangla-English language pair, BLEU score*

Date of Submission: 01-09-2024 Date of Acceptance: 09-09-2024

I. Brief History Of Neural Machine Translation

In the 1980s and 1990s, machine translation was in the sight of researchers exploring different methods (Waibel et al., 1991). Moreover, the models proposed by Forcada and Ñeco (1997) and Castaño et al. (1997) are striking relevant to the present dominant neural machine translation approaches. In fact, none of these models were trained enough on data sizes large to make reasonable results for anything but miniature examples. The computational complexity involved by far exceeded the computational resources of that era, and hence the idea was unrestricted for almost two decades.

During this intermediate period, data-driven approaches such as phrase-based statistical machine translation rose from unrecognizability to dominance and made machine translation a useful tool for information exchange, from information gisting to increasing the productivity of professional and effective translators.

The modern neural methods in machine translation started with the integration of neural language models into traditional statistical machine translation systems. The pioneering work by Schwenk (2007) showed large improvements in public evaluation campaigns. However, these ideas were slowly adopted, mainly due to computational concerns. The use of GPUs for training also posed a challenge for many research groups because of hardware or the experience to exploit it.

Moving beyond the use in language models, neural network methods crept into other components of traditional statistical machine translation, such as providing additional scores or extending translation tables (Schwenk, 2012; Lu et al., 2014), reordering (Kanouchi et al., 2016; Li et al., 2014) and preordering models (de Gispert et al., 2015), and so on. For instance, the joint translation and language model by Devlin et al. (2014) was influential since it showed large quality improvements on top of a very competitive statistical machine translation system.

More ambitious efforts aimed at pure neural machine translation, abandoning existing statistical approaches completely. Early steps were the use of convolutional models (Kalchbrenner and Blunsom, 2013) and sequence-to-sequence models (Sutskever et al., 2014; Cho et al., 2014). These were able to produce reasonable translations for short sentences, but fell apart with increasing sentence length. The addition of the attention mechanism finally yielded competitive results (Bahdanau et al., 2015; Jean et al., 2015b). With a few more refinements, such as byte pair encoding and back-translation of target-side monolingual data, neural machine translation became the new state of the art.

Within a year or two, the entire research field of machine translation went neural. To give some indication of the speed of change: At the shared task for machine translation organized by the Conference on Machine Translation (WMT), only one pure neural machine translation system was submitted in 2015. It was competitive, but outperformed by traditional statistical systems. A year later, in 2016, a neural machine translation system won in almost all language pairs. In 2017, almost all submissions were neural machine translation systems. Researchers focused on making NMT models more scalable and efficient, leading to the development of models that could handle larger datasets and longer sequences. Improvements in hardware acceleration, particularly using GPUs and TPUs, allowed for faster training times and larger model capacities. Google introduced a multilingual NMT system that could translate multiple languages within a single model, improving resource efficiency and performance on low-resource languages.

Major tech companies like Google, Facebook, and Microsoft integrated NMT into their translation services, providing users with more accurate translations and expanding the reach of NMT technology. Google Translate, in particular, made a significant shift from statistical machine translation (SMT) to NMT, greatly enhancing translation quality across many language pairs. Several open-source NMT frameworks were developed and released, such as OpenNMT by Harvard NLP and Tensor2Tensor by Google etc., making it easier for researchers and developers to experiment with and deploy NMT systems. NMT systems have the mechanisms to estimate the quality of translations and correct errors which improve reliability. Quality Estimation models predict the quality of translations without reference translations, allowing for better real-time use of NMT systems. Automatic error correction systems enhance the final output quality by fixing common translation errors.

A transformer is a deep learning architecture developed by Google and based on the multi-head attention mechanism, proposed in a 2017. Transformers have the advantage of having no recurrent units, and thus requires less training time than recurrent neural architectures, such as long short-term memory (LSTM),[7] and its later enhancement has been prevalently adopted for training large language models (LLM) on large (language) datasets, such as the Wikipedia corpus and Common Crawl.

At the time of writing, neural machine translation research is progressing at rapid pace. There are many directions that are explored ranging from core machine learning improvements such as deeper models to more linguistically informed models. More insight into the strength and weaknesses of neural machine translation is being gathered and will inform future work.

There is an extensive spread of toolkits available for research, development, and deployment of neural machine translation systems. At present, the number of toolkits is increasing, rather than consolidating. Some of the promising tools that are used for neural machine translation systems are given below-

Toolkits	Framework	Usage scale
Tensor2Tensor	TensorFlow	
TensorFlow/NMT	TensorFlow	
Fairseq	PyTorch	
OpenNMT-py	Lua, (Py)Torch, TF	
Sockeye	MXNet	
OpenSeq2Seq	TensorFlow	
Nematus	TensorFlow, Theano	
PyTorch/Translate	PyTorch	
Marian	C++	
NMT-Keras	TensorFlow, Theano	
Neural Monkey	TensorFlow	
THUMT	TensorFlow, Theano	
Eske/Seq2Seq	TensorFlow	
XNMT	DyNet	
NJUNMT	PyTorch, TensorFlow	
Transformer-DyNet	DyNet	
SGNMT	TensorFlow, Theano	
CythonMT	C++	
Neutron	PyTorch	

Table 1: Toolkits with respective framework and the amount of practices of different frameworks.

The remaining of the paper is organized as follows. Section **II**. provides an introduction of Neural machine translation and its history of development. Section **III**. introduces the architectures of NMT and the procedure of training and testing. Section **IV**. discusses the performances of different models between Bangla and English language pair based on existing architectures. Section **V.** represents the comparisons of performances regarding the section IV. Section **VI**. Conclusion and Section **VII**. Future works.

II. Introduction Of Neural Machine Translation

Neural Machine Translation (NMT) is a subfield of artificial intelligence and computational linguistics that focuses on developing algorithms capable of translating text from one language to another. Unlike traditional rule-based or statistical translation approaches, NMT utilizes deep learning techniques to generate translations that are more accurate and natural-sounding.

NMT models are built using deep neural networks, specifically Recurrent Neural Networks (RNNs) or Transformer architectures. These models are trained on vast amounts of bilingual data, such as sentence pairs in different languages, to learn the underlying patterns and structures of languages.

During the translation process, the NMT model takes an input sentence in the source language and passes it through multiple layers of neural networks. These networks analyse the sentence's context and generate a corresponding translation in the target language. The model's ability to consider the entire input sentence contextually enables it to produce more coherent and contextually accurate translations.

Advantages of Neural Machine Translation

1. Improved Translation Quality: NMT models have shown significant improvements in translation quality compared to traditional rule-based methods. They can capture nuances, idiomatic expressions, and context-specific meanings more effectively. That's not to say translations can be published without proofreading and editing. Linguistic Sign-Off (LSO) remains a crucial step in the process to ensure consistency.
2. Context-Aware Translations: NMT models have the ability to understand the context of the entire sentence, leading to more accurate and contextually appropriate translations. This is particularly beneficial for languages with complex grammar and sentence structures.
3. Faster Translation Speed: Neural Machine Translation eliminates the need for explicit pre-processing steps, making the translation process faster and more efficient. This is crucial in scenarios where real-time translation is required.
4. Adaptability and Scalability: NMT models can be easily adapted to different language pairs and domains by retraining them on specific data. This adaptability makes NMT highly scalable and applicable to various translation needs.

Some Challenges in Neural Machine Translation

1. Rare or Unseen Phrases: NMT models struggle with translating rare or unseen phrases since they heavily rely on the patterns learned during training. In such cases, the translations may not be accurate or may lack fluency.
2. Handling Ambiguity: Ambiguities present in the source language can pose challenges for NMT models, leading to inaccurate translations. Resolving ambiguity requires a deep understanding of the context, which remains a complex task for current NMT models.
3. Resource-Intensive Training: Training NMT models requires a significant number of computational resources and large datasets. This can limit the accessibility and applicability of NMT to certain languages or organizations with limited resources.

The prospective future of Neural Machine Translation

As technology continues to advance, Neural Machine Translation is expected to witness further advancements. Some potential future developments include:

1. **Improved Handling of Rare Phrases:** Future research aims to enhance NMT models' ability to handle rare or unseen phrases, ensuring more accurate and contextually appropriate translations.
2. **Multimodal Translation:** NMT models may evolve to incorporate other modalities, such as images or videos, to provide more comprehensive and accurate translations in various contexts.
3. **Domain-Specific Adaptation:** NMT models could be further optimized for specific domains or industries, enabling more accurate and tailored translations for specialized fields.
4. **Enhanced Language Generation:** Future NMT models may focus on generating translations that are not only accurate but also stylistically appropriate, considering factors like formality, tone, and cultural nuances.

III. The Architectures Of NMT And The Procedure Of Training And Testing

Feedforward neural network

A feedforward is one of the more basic forms of neural networks, and one can often use the architecture of a feedforward neural network to create more specialized networks. As the name suggests, feedforward neural networks feed data forward from input to output with no loops or circles. Although it's one of the simplest structures for neural networks, the hidden layers between input and output can still be complex. You can use this type of neural network for various tasks, such as pattern and image recognition, regression analysis, and classification.

How does a feedforward neural network architecture work?

A feedforward neural network has an input layer, followed by a series of hidden layers, and ends with an output layer. Data flows into the algorithm through the input and passes through the nodes in the first layer. The first layer of nodes computes the data, based on the node's weights and passes the calculation to the next layer of nodes. Each node in each layer connects to each node in the next layer, but the data can only flow towards the output.

Architecture of Feedforward Neural Networks

The architecture of a feedforward neural network consists of three types of layers: the input layer, hidden layers, and the output layer. Each layer is made up of units known as neurons, and the layers are interconnected by weights.

- **Input Layer:** This layer consists of neurons that receive inputs and pass them on to the next layer. The number of neurons in the input layer is determined by the dimensions of the input data.
- **Hidden Layers:** These layers are not exposed to the input or output and can be considered as the computational engine of the neural network. Each hidden layer's neurons take the weighted sum of the outputs from the previous layer, apply an activation function, and pass the result to the next layer. The network can have zero or more hidden layers.
- **Output Layer:** The final layer that produces the output for the given inputs. The number of neurons in the output layer depends on the number of possible outputs the network is designed to produce.

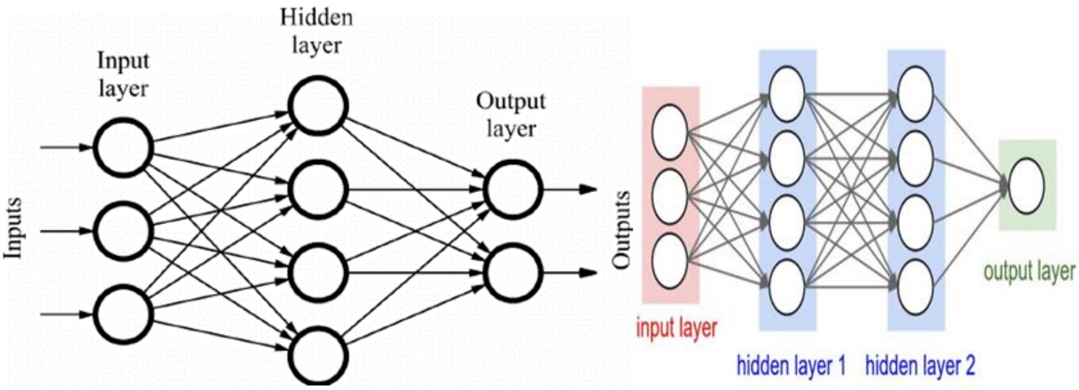

Fig1: Each neuron in one layer is connected to every neuron in the next layer, making this a fully connected network. The strength of the connection between neurons is represented by weights, and learning in a neural network involves updating these weights based on the error of the output.

Convolutional neural network (CNN)

Convolutional neural networks are particularly skilled at recognizing patterns and images, which makes them important for AI technology like computer vision, among other uses. For example, the US Postal Services uses neural networks to recognize handwritten zip codes. Convolutional neural networks are different from other networks because of their architecture and because the CNN nodes have shared weights and bias values, unlike feedforward or recurrent neural networks. They have shared weight because each node does the same job in a different input area, such as detecting the edge of an image.

How does a convolutional neural network architecture work?

In addition to input and output layers, convolutional neural networks contain two main types of hidden layers: convolutional and pooling. Convolutional layers filter the input, typically an image, to extract various features. This data then feeds into a pooling layer, simplifying the parameters but keeping important information. The process repeats many times, sometimes including other layers, such as a multilayer perceptron or a rectified linear unit for activation.

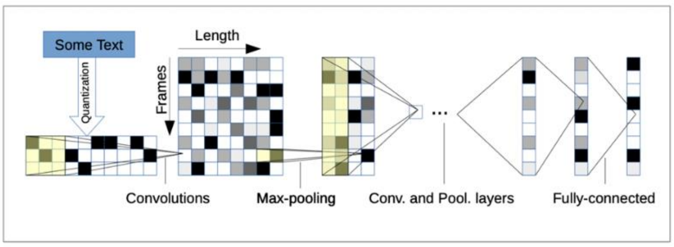

Fig2: Convolutional neural networks for text understanding (X. Zhang and LeCun 2015).

Recurrent Neural Network (RNN)

RNNs are designed to take sequences of text as inputs or return sequences of text as outputs, or both. They're called recurrent because the network's hidden layers have a loop in which the output and cell state from each time step become inputs at the next time step. This recurrence serves as a form of memory. It allows contextual information to flow through the network so that relevant outputs from previous time steps can be applied to network operations at the current time step.

How does a Recurrent neural network architecture work?

Depending on the use-case, one may want to set up RNN to handle inputs and outputs differently. For a project, one will use a many-to-many process where the input is a sequence of English words and the output is a sequence of French words (fourth from the left in the diagram below).

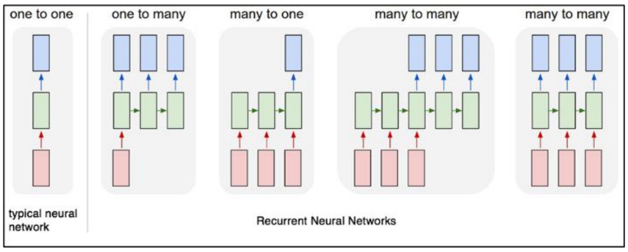

Fig3: Each rectangle is a vector and arrows represent functions. Input vectors are in red, output vectors are in blue and green vectors hold the RNN's state.

Recurrent networks have many variations. One of their famous versions is Long Short-Term Memories (LSTMs). LSTMs can handle long-term sequences. They have a cell state (horizontal straight line in figure below) and gates which all smooth the flow of information.

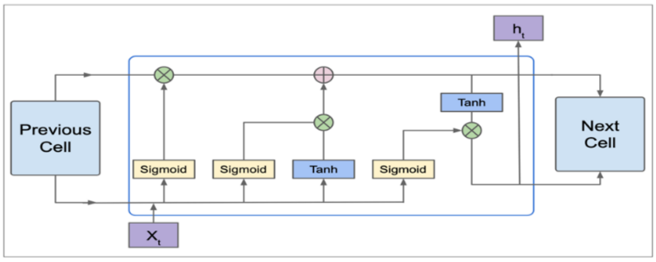

Fig4: Long Short-Term Memories (LSTMs).

Another slightly efficient version of LSTMs is gate recurrent Units (GRUs). LSTMs works great for basic sequence modelling problems but they are still limited in how far they can go.

Transformer Architecture

Transformer is a neural network architecture that can process sequential data such as texts, audios, videos, and images (as a sequence of image patches). Transformer does not use any recurrent or convolution layers. Its fundamental layer is called Attention. It also contain other basic layers such as fully-connected layers, normalization layer (LayerNorm mostly) (Ba, Kiros, and Hinton 2016), embedding layer, and positional encoding layer. We will see what each of those layers performs in next sections.

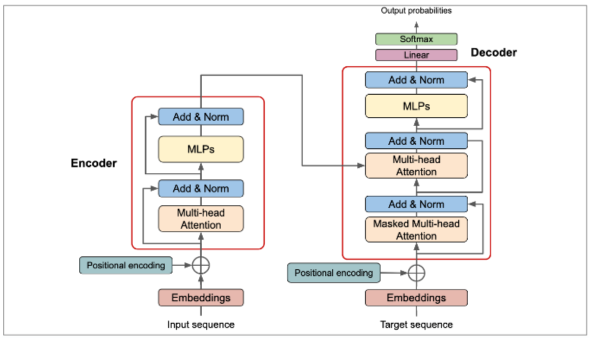

Figure 5: Transformer Architecture. Adapted from (Vaswani et al. 2017).

As we alluded to in the beginning, transformer was initially introduced for machine translation, a task that demands processing two sequences (both input and output are sequences). Thus, the transformer model had two parts: encoder for processing the input and decoder for generating the output. More about encoder, decoder, and other layers are discussed below.

Encoder

Encoder is one of the main blocks of the transformer architecture that is right at the input of input sequence. Encoder transforms input sequence into compressed representation. In the original transformer architecture, the encoder was repeated 6 times (this depends on overall size of architecture, it can be changed). Each encoder block has 3 main layers which are multi-head attention (MHA), layer norm, and Multi-Layer Perceptron (MLPs).

Multi-head attention and MLPs are referred to as sub-layers in the transformer. Between sublayers, there are layer normalization and dropout and residual connections in between (refer to diagram for correct flow of those layers).

The number of encoder layers was 6 as said previously. The more the number of encoder layers, the larger the model, and the more the model is likely to capture the global context of the input sequences hence resulting in better task generalization.

Decoder

The decoder is pretty much the same as encoder except additional multi-head attention that operated over the output of the encoder. The goal of the decoder is to fuse encoder output with the target sequence and to make predictions or to predict the next token.

The attention that takes the target sequence in decoder is masked to prevent the current token (being processed) from attending to subsequent tokens in the target sequence. If the decoder has access to a full target sequence, this would basically be cheating and can result in model that cannot generalize beyond the training data.

Decoder is also typically repeated the same times as encoder. In the original transformer, the number of decoder blocks were also 6 blocks.

IV. The Performances Of Different Architectures In Neural Machine Translations Of Different Bangla-English Language Pairs:

We have reviewed 32 translators of neural machine translation where we found that different architectures are used to translate from English language to Bangla language or vice versa. The mostly used architectures are Transformer, Basic RNN, RNN with Encoding, Bidirectional RNN, RNN Encoder and Decoder, Bidirectional with Encoding RNN, RNN- LSTM, RNN – GRU, RNN with Attention Mechanism. We have grouped these architectures into 3 major groups- a) Transformer b) RNN with additional technologies and c) RNN with Attention. We have represented the BLEU scores of English and Bangla language pairs with respect to different architectures and data sets in the following tables.

Translator #	**Model used architectures**	**Languages**	**Dataset**	**BLEU**	**BLEU On Average**
1	Transformer	**English to Bangla**	Supara (70861parallel sentences)	25.44	20.85
2	Transformer	**English to Bangla**	SUPara, GlobalVoices (197338 parallel sentences)	16.26	

Table 2: BLEU scores of translations from English language to Bangla language using Transformer architectures.

In the Table 2, the architecture used for translation where the source language is English and the target language is Bangla is Transformer. The BLEU score and dataset of individual translator is mentioned. The average BLEU score in translation from English to Bangla is 20.85.

Translator #	**Model used architectures**	**Languages**	**Dataset**	**BLEU**	**BLEU On Average**
1	Transformer	Bangla to English	LMPC, SIPC, PTB, SUPara , AmaderCAT (419109 parallel sentences)	18.99	19.71
2	Transformer	Bangla to English	Supara (70861parallel sentences)	21.42	

3	Transformer	Bangla to English	ILMPC (324,366 parallel sentences)	18.73	

Table 3: BLEU scores of translations from Bangla language to English language using Transformer architectures.

The architecture Transformer is used for translation where the source language is Bangla and the target language is English. The BLEU scores and datasets of translators are mentioned in the Table 3. The average BLEU score in translation from Bangla to English is 19.71.

Translator #	Model used architectures	Languages	BLEU	BLEU On Average
1	Transformer	English to Bangla	20.85	20.28
2	Transformer	Bangla to English	19.71	

Table 4: On average BLEU score of translations between English and Bangla using Transformer architecture. (bidirectional)

Using Transformer architecture, the average BLEU score of bidirectional translation between English and Bangla language pair mentioned in Table 2 and Table 3 is 20.28.

Translator #	Model used architectures	Languages	Dataset	BLEU	BLEU On Average
1	RNN- BiLSTM	Bangla to English	5204 English and Bengali bilingual parallel sentences	47.46	23.60
2	RNN- BiGRU	Bangla to English	5204 English and Bengali bilingual parallel sentences	43.21	
3	RNN- LSTM	Bangla to English	5204 English and Bengali bilingual parallel sentences	41	
4	RNN- GRU	Bangla to English	5204 English and Bengali bilingual parallel sentences	39	
5	Bidirectional RNN-LSTM	Bangla to English	SUPara (70861parallel sentences)	19.76	
6	Bidirectional RNN-LSTM	Bangla to English	ILMPC, SIPC, PTB, SUPara, AmaderCAT (419109 parallel sentences)	19.24	
7	RNN- LSTM	English to Bangla	19959 English Bangla parallel sentences (Aviation domain)	39.97	
8	RNN- BiLSTM + BN-Emb	Bangla to English	SUPara (70861parallel sentences)	19.44	
9	RNN- BiLSTM + BN-Emb, EN-Emb	Bangla to English	SUPara (70861parallel sentences)	19.24	
10	RNN- BiLSTM	Bangla to English	ILMPC (324,366 parallel sentences)	15.94	
11	RNN- BiLSTM + BN-Emb	Bangla to English	ILMPC (324,366 parallel sentences)	16.21	
12	RNN- BiLSTM + BN-Emb, EN-Emb	Bangla to English	ILMPC (324,366 parallel sentences)	16.36	
13	RNN- BiLSTM	Bangla to English	ILMPC+SIPC+PTB	15.24	
14	RNN- BiLSTM + BN-Emb	Bangla to English	ILMPC+SIPC+PTB	15.56	
15	RNN- BiLSTM + BN-Emb, EN-Emb	Bangla to English	ILMPC+SIPC+PTB	15.62	
16	Classical NMT	Bangla to English	8000 parallel sentences (Literature-based)	8.56	
17	Classical NMT	Bangla to English	11500 parallel sentences (Literature-based and custom)	9.28	
18	Classical NMT	Bangla to English	More than 1 million sentence pairs	13.43	
19	Classical NMT	English to Bangla	4,84,131 sentence pairs	26.76	
20	NMT with pre-trained embedding	English to Bangla	4,84,131 sentence pairs	26.92	
21	NMT with synthetic monolingual data	English to Bangla	4,84,131 sentence pairs	27.46	

Table 5: BLEU scores of translations between Bangla and English language pairs using RNN and classical NMT architectures.

The BLEU scores of Bangla and English language pairs translations are mentioned in Table 5 where the architecture of neural machine translation is Recurrent Neural Network (RNN) with additional technologies and classical NMT. The average BLEU score of Bangla-English language pairs translation is 23.60. It is observed that the architecture of classical NMT provides the most poor-quality translation whose BLEU score is 8.56 and the best quality translation is provided by the architecture of RNN-BiLSTM whose BLEU score is 47.46.

Translator #	Model used architectures	Languages	Dataset	BLEU	BLEU On Average
1	Bidirectional RNN-Encoder and Decoder Attention Mechanism	English to Bangla	28,927 images and the same number of corresponding English-Bengali parallel sentences (Multimodal)	43.90	26.95
2	Bidirectional RNN-Encoder and Decoder Attention Mechanism	English to Bangla	28,927 English-Bengali parallel sentences	40.90	
3	RNN with Attention Mechanism	Bangla to English	SUPara, GlobalVoices (197338 parallel sentences)	22.38	
4	RNN with Attention Mechanism	English to Bangla	SUPara, GlobalVoices (197338 parallel sentences)	15.57	
5	RNN-Attention Mechanism with BPE (Byte Pair Encoding)	English to Bangla	SUPara, GlobalVoices (197338 parallel sentences)	16.26	
6	RNN-Attention Mechanism with BPE (Byte Pair Encoding)	Bangla to English	SUPara, GlobalVoices (197338 parallel sentences)	22.68	

Table 6: BLEU scores of translations between Bangla and English language pairs using RNN architecture with attention mechanism.

The language pairs mentioned in Table 6 used RNN with attention mechanism for neural machine translation. The BLEU score of individual language pair translation is mentioned independently and the average BLEU score of all language pairs is 26.95. The lowest BLUE score of translation from English to Bangla is 15.57 indicates the poorest translation in Recurrent Neural Network with attention mechanism whereas the best BLEU score of 43.90 is provided by the architecture of Bidirectional RNN-Encoder and Decoder Attention Mechanism from English to Bangla translation.

V. Discussion On Performances Regarding Comparisons

In the above five tables, we have represented 3 major architectures named Transformer, RNN with additional technologies and RNN with attention mechanism. Using Each architecture, Bangla and English language pairs are translated and their BLEU scores are mentioned. It is also calculated the average BLEU score of language pairs translation of each architecture. The BLEU scores of Transformer, RNN with additional technologies and RNN with attention mechanism are 20.28, 23.60 and 26.95 respectively. Considering the average BLEU scores, it is found that RNN with attention mechanism mentioned in Table 6 generated the best quality of translation whereas the Transformer architecture generated the lowest quality translation. It is also mentioned that the datasets and amount of data or sentence pairs are not same for all architectures. Bur considering the qualities translations of different architectures applied in Bangla-English language pairs, we can say that model with RNN with attention mechanism provides the best translation.

VI. Conclusion

Neural Machine Translation is a revolutionary technology that has transformed the field of language translation. With its ability to produce accurate, contextually appropriate translations, NMT is bridging the language gap and facilitating effective communication across cultures and languages. In our paper we have observed that among three major architectures of neural machine translation, RNN with attention mechanism provides the best quality translation on average of different Bangla-English language pairs. But the findings may be different with respect to datasets and the amount of text of datasets. Although Neural Machine Translation faces challenges, ongoing research and advancements of computations continue to push the boundaries of NMT, making it a promising field for the future.

VII. Future Work

The chronological observation shows that the architectures of neural machine translation are updated with new technologies to increase the accuracy of bilingual translation. The researchers will enhance the models until to ensure 100% accurate translation. Low resource languages are suffered more to achieve the expected translation. Since Bangla is a low resource language, the translation quality between Bangla and other languages is comparatively poor regarding other languages. Our future works is to develop a bilingual translator for Bangla and English languages pair using neural network.

References

[1] Jordi Armengol-Estapé, Marta R. Costa-Jussà (2021), Semantic And Syntactic Information For Neural Machine Translation - Injecting Features To The Transformer. Https://Doi.Org/10.1007/S10590-021-09264-2

[2] Zhixing Tan, Shuo Wang, Et Al. (2020). Neural Machine Translation: A Review Of Methods, Resources, And Tools. Www.Keaipublishing.Com/En/Journals/Ai-Open

[3] Philipp Koehn (2017). Neural Machine Translation. Arxiv:1709.07809v1

[4] Brown, T. B., Mann, B., Ryder, N., Et Al. (2020). Language Models Are Few-Shot Learners. *Arxiv Preprint Arxiv:2005.14165*.

[5] Vaswani, A., Shazeer, N., Parmar, N., Et Al. (2017). Attention Is All You Need. *Advances In Neural Information Processing Systems*, 5998-6008.

[6] Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The Long-Document Transformer. *Arxiv Preprint Arxiv:2004.05150*.

[7] Liu, Y., Ott, M., Goyal, N., Et Al. (2020). Roberta: A Robustly Optimized Bert Pretraining Approach. *Arxiv Preprint Arxiv:1907.11692*.

[8] Tang, Y., Tran, C., Li, X., Et Al. (2020). Multilingual Translation With Extensible Multilingual Pretraining And Finetuning. *Arxiv Preprint Arxiv:2008.00401*.

[9] Johnson, M., Schuster, M., Le, Q. V., Et Al. (2017). Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Transactions Of The Association For Computational Linguistics*, 5, 339-351.

[10] Lample, G., & Conneau, A. (2019). Cross-Lingual Language Model Pretraining. *Advances In Neural Information Processing Systems*, 32.

[11] Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding Back-Translation At Scale. *Proceedings Of The 2018 Conference On Empirical Methods In Natural Language Processing*, 489-500.

[12] Fadaee, M., Bisazza, A., & Monz, C. (2017). Data Augmentation For Low-Resource Neural Machine Translation. *Proceedings Of The 55th Annual Meeting Of The Association For Computational Linguistics*, 567-573.

[13] Specia, L., Scarton, C., & Paetzold, G. (2018). Quality Estimation For Machine Translation. *Synthesis Lectures On Human Language Technologies*, 11(1), 1-162.

[14] Junczys-Dowmunt, M., Grundkiewicz, R., & Heafield, K. (2018). Approaching Neural Grammatical Error Correction As A Low-Resource Machine Translation Task. *Proceedings Of The 2018 Conference Of The North American Chapter Of The Association For Computational Linguistics: Human Language Technologies*, 595-606.

[15] Koehn, P., & Knowles, R. (2017). Six Challenges For Neural Machine Translation. *Proceedings Of The First Workshop On Neural Machine Translation*, 28-39.

[16] Gu, S., Chen, Q., Li, X., Et Al. (2018). Meta-Learning For Low-Resource Neural Machine Translation. *Proceedings Of The 2018 Conference On Empirical Methods In Natural Language Processing*, 3622-3631.

[17] Rei, R., Søgaard, A., & Ruder, S. (2020). Comet: A Neural Framework For Mt Evaluation. *Proceedings Of The 2020 Conference On Empirical Methods In Natural Language Processing*, 2685-2702.

[18] Haifeng Wang, Hua Wu, Zhongjun He, Liang Huang, Kenneth Ward Church (2022), Progress In Machine Translation. Www.Elsevier.Com/Locate/Eng

[19] Sainik Kumar Mahata*, Dipankar Das And Sivaji Bandyopadhyay (2019), Mtil2017: Machine Translation Using Recurrent Neural Network On Statistical Machine Translation, Https://Doi.Org/10.1515/Jisys-2018-0016

[20] Denny Britz , Anna Goldie, Minh-Thang Luong, Quoc Le (2017), Massive Exploration Of Neural Machine Translation Architectures, Https://Github.Com/Google/Seq2seq/

[21] Shuoheng Yang, Yuxin Wang, Xiaowen Chu (2020), A Survey Of Deep Learning Techniques For Neural Machine Translation, Arxiv:2002.07526v1 [Cs.Cl]

[22] Janhavi R. Chaudhary, Prof. Ankit C. Patel (2018), Bilingual Machine Translation Using Rnn Based Deep Learning, Ijsrset | Volume 4 | Issue 4 | Print Issn: 2395-1990 | Online Issn: 2394-4099

[23] Shohei Iida, Ryuichiro Kimura, Hongyi Cui, Po-Hsuan Hung, Takehito Utsuro, Masaaki Nagata (2019), A Multi-Hop Attention For Rnn Based Neural Machine Translation, Proceedings Of The 8th Workshop On Patent And Scientific Literature Translation

[24] Debajit Datta, Preetha Evangeline David, Dhruv Mittal, Anukriti Jain (2020), Neural Machine Translation Using Recurrent Neural Network, International Journal Of Engineering And Advanced Technology (Ijeat) Issn: 2249 – 8958, Volume-9 Issue-4, Doi: 10.35940/Ijeat.D7637.049420

[25] Mohammad Wasil Saleem, Sandeep Uprety (2021), Neural Machine Translation With Attention, Https://Www.Researchgate.Net/Publication/355917108

[26] Felix Stahlberg (2020), Neural Machine Translation: A Review, Journal Of Artificial Intelligence Research 69 (2020) 343-418

[27] Antonio Valerio Miceli Barone, Jindrˇich Helcl, Rico Sennrich, Barry Haddow, Alexandra Birch (2017), Deep Architectures For Neural Machine Translation, Proceedings Of The Conference On Machine Translation (Wmt), Volume 1: Research Papers, Pages 99–107

[28] Md. Abdul Awal Ansary (2024), Neural Machine Translation: A Philosophical Survey Observing Architectures And Performances, Www.Iosrjournals.Org, E-Issn: 2278-0661,P-Issn: 2278-8727, Volume 26, Issue 4, Ser. 3 (July. – August. 2024), Pp 01-09

[29] Argha Chandra Dhar, Arna Roy, Et Al. (2021), Bangla English Machine Translation Using Attention-Based Multi-Headed Transformer Model, Journal Of Computer Science 2021, 17 (10): 1000.1010, Doi: 10.3844/Ajbbsp.2021.1000.1010

[30] Nipun Paul, Ishmam Faruki, Et Al. (2023), Bengali-English Neural Machine Translation Using Deep Learning Techniques, 2023 International Conference On Electrical, Computer And Communication Engineering (Ecce)

[31] Sahinur Rahman Laskar, Pankaj Dadure, Et Al. (2022), English To Bengali Multimodal Neural Machine Translation Using Transliteration-Based Phrase Pairs Augmentation, Proceedings Of The 9th Workshop On Asian Translation, Pages 111–116

[32] Mohammad Abdullah Al Mumin, Md Hanif Seddiqui, Et Al. (2019), Neural Machine Translation For Low-Resource English-Bangla, Journal Of Computer Science 2019, 15 (11): 1627.1637, Doi: 10.3844/Jcssp.2019.1627.1637

[33] Saptarshi Paul, Bipul Shyam Purkhyastha (2020), English To Bengali Neural Machine Translation System For The Aviation Domain, Infocomp, V.19 , N. 2, P. 78-97, 2020

[34] Md. Arid Hasan, Firoj Alam, Et Al. (2019), Neural Machine Translation For The Bangla-English Language Pair, 22nd International Conference Of Computer And Information Technology (Iccit), 18-20 December, 2019

[35] Md. Arid Hasan, Firoj Alam, Et Al. (2019), Neural Vs Statistical Machine Translation: Revisiting The Bangla-English Language Pair, International Conference On Bangla Speech And Language Processing (Icbslp), 27-28 September 2019