Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.035814
Article

Neural Machine Translation Models with Attention-Based Dropout Layer

Huma Israr[1,*], Safdar Abbas Khan[1], Muhammad Ali Tahir[1], Muhammad Khuram Shahzad[1], Muneer Ahmad[1] and Jasni Mohamad Zain[2,*]

[1]School of Electrical Engineering and Computer Science (SEECS), National University of Science and Technology, Islamabad, Pakistan
[2]Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Kompleks Al-Khawarizmi, Universiti Teknologi MARA, 40450, Shah Alam, Selangor, Malaysia
*Corresponding Authors: Huma Israr. Email: hisrar.dphd17seecs@seecs.edu.pk; Jasni Mohamad Zain. Email: jasni67@uitm.edu.my
Received: 05 September 2022; Accepted: 14 December 2022

Abstract: In bilingual translation, attention-based Neural Machine Translation (NMT) models are used to achieve synchrony between input and output sequences and the notion of alignment. NMT model has obtained state-of-the-art performance for several language pairs. However, there has been little work exploring useful architectures for Urdu-to-English machine translation. We conducted extensive Urdu-to-English translation experiments using Long short-term memory (LSTM)/Bidirectional recurrent neural networks (Bi-RNN)/Statistical recurrent unit (SRU)/Gated recurrent unit (GRU)/Convolutional neural network (CNN) and Transformer. Experimental results show that Bi-RNN and LSTM with attention mechanism trained iteratively, with a scalable data set, make precise predictions on unseen data. The trained models yielded competitive results by achieving 62.6% and 61% accuracy and 49.67 and 47.14 BLEU scores, respectively. From a qualitative perspective, the translation of the test sets was examined manually, and it was observed that trained models tend to produce repetitive output more frequently. The attention score produced by Bi-RNN and LSTM produced clear alignment, while GRU showed incorrect translation for words, poor alignment and lack of a clear structure. Therefore, we considered refining the attention-based models by defining an additional attention-based dropout layer. Attention dropout fixes alignment errors and minimizes translation errors at the word level. After empirical demonstration and comparison with their counterparts, we found improvement in the quality of the resulting translation system and a decrease in the perplexity and over-translation score. The ability of the proposed model was evaluated using Arabic-English and Persian-English datasets as well. We empirically concluded that adding an attention-based dropout layer helps improve GRU, SRU, and Transformer translation and is considerably more efficient in translation quality and speed.

Keywords: Natural language processing; neural machine translation; word embedding; attention; perplexity; selective dropout; regularization; Urdu; Persian; Arabic; BLEU

1 Introduction

Machine translation (MT) is a field of computational linguistics that uses software to translate text from one natural language (NL) into another natural language (NL). The digital revolution's impact and the ever-growing scientific and political interests in knowledge sharing have sparked significant interest in MT. According to Ethnologue, there are 7,117 living languages worldwide [1]. Unlike well-resourced languages, less work has been done on translating under-resourced and minority languages. This is primarily because linguistic resources such as rich corpus are unavailable for these languages. Over the last few years, there has been a growing interest and awareness among the scientific community and locally among the advocates of minority languages in sustaining and expanding the existing resources and translating under-resourced languages [2].

Urdu is a low-resource, morphologically rich, and complex language [3]. Yet, a few attempts have been made to develop Urdu MT. Over the past few decades, various researchers have explored Urdu MT, but they were significantly more focused on translating English to Urdu. Apparently, no significant work has been conducted on translating Urdu to English. The early effort towards an Urdu MT system adopted the rule-based machine translation (RBMT) technique. RBMT uses handwritten linguistic rules for both languages in its translation process [4–6]. Translation results produced by RBMT-based translators are subjectively analyzed and claimed to give more accurate results. RBMT-based translators require considerable effort to prepare morphological, syntactic, and/or semantic rules for both the languages and the additional rules to handle case phrases. Later advances in RBMT-based translator used additional rules to handle case phrases, verb positions, and transfer approach [7]. However, with the growing availability of linguistic corpora, the corpus-based (CB) approach to MT has strengthened and proved to be more effective and useful than the RBMT approach. Example-based machine translation (EBMT) [8] is a CB approach. For translation, EBMT systems use sentence fragments rather than individual words [9]. EBMT system shows its limitation once the source text exceeds the imposed limits. Working with massive bilingual corpora, statistical machine translation (SMT) uses statistical concepts of probabilities while translating source sentences into target language sentences [10]. Few researchers adopted an SMT-based system for Urdu translation [11,12], while others investigated phrase-based SMT using reordering models [13].

Neural network (NNET) models have seen an incredible resurgence in recent years, obtaining state-of-the-art results in computer vision [14–16], speech recognition [17–19], bio-medical science [20], and many other tasks. More recently, they have shown substantial improvements in MT with the potential of addressing many shortcomings of traditional MT systems. Sutskever et al. performed translation using a large, single, and end-to-end (E2E) trainable neural network [21]. These E2E models have significantly been improved recently through the use of RNN, LSTM [22], and GRU [23], organized into the encoder-decoder architecture that is capable of persisting data over several time steps. The encoder part encodes all source-side semantic details into fixed-size vectors. The decoder then generates a new sequence in the target language, conditioned on the encoder's representation. However, using a fixed-length vector is inadequate to represent a source sentence as there is no synchrony between input and output sequences, and it lacks the notion of alignment. To solve the problem of failed word alignment, a well-known attention mechanism is introduced [24]. Attention-based NMT models automatically recognize source words relevant to the next target word and give

high attention to these source words in computing the context vector. Luong et al. presented different single-layer multiplicative attention mechanisms (local and global) for RNNs-based NMT models [25]. In 2017, Gehring et al. [26] proposed a convolutional sequence-to-sequence (seq2seq) model that used an attention mechanism to compute intermediate encoder and decoder states. Through empirical demonstration, they concluded that the convolutional approach discovers the sequence's compositional structure more easily. Vaswani et al. proposed a network structure called Transformer. The Transformer used a fixed context length with multi-head attention and claimed to capture long-distance relationships better than any other recurrent architecture [27]. Today neural translators have reached a level of reliability and efficiency in many bilingual texts. However, perfection is still a long way. There is still room for improvement in the existing NMT models.

The main contributions of this paper are as follows:

1. This study is the first comprehensive attempt to use the NMT model for Urdu-to-English translation. It performs a comparative analysis of attention-based LSTM and Bi-RNN, GRU, CNN, Transformer, and SRU for Urdu-to-English MT.
2. This paper also explores the problem of failed word alignment and over-translation issues in the translation output. A novel NMT model with an attention-based dropout layer has been proposed to alleviate the identified problem.
3. This study also pioneers employing the concept of attention-based dropout layer in the context of NMT. The results obtained have been quantitatively and qualitatively analyzed in comparison to the results obtained from the NMT model without the attention-based dropout layer.
4. Specifically, more experimental results of Urdu-to-English translation using NMT models and more detailed and rigorous explanations on using NMT models with attention-based dropout layer are presented. Additionally, the newly proposed model for translation has also been validated using Persian-English and Arabic-English datasets.

The rest of the paper is organized as follows: A list of abbreviations used in the article is given in Table 1. Section 2 presents related work, while Section 3 presents the methodology to elaborate the working of the baseline system. It is followed by Section 3.8, which is our proposed model for MT. Section 4 explains the experimental setup, and Section 5 presents our proposed work's experimental results, followed by the conclusion and future work in Section 6.

Table 1: List of abbreviations

#	Acronyms	Description	#	Acronyms	Description
1.	NMT	Neural machine translation	8.	WER	Word error rate
2.	RNN	Recurrent neural network	9.	GRU	Gated recurrent unit
3.	LSTM	Long short-term memory	10.	SRU	Statistical recurrent unit
4.	CNN	Convolutional neural network	11.	ADL	Attention-based dropout layer
5.	BLEU	Bilingual evaluation understudy	12.	TER	Translation edit rate
6.	UTEM	Under-translation evaluation metric	13.	OTEM	Over-translation evaluation metric
7.	Bi-RNN	Bidirectional recurrent neural network	14.	GLUE	General language understanding evaluation

2 Related Work

This part of the paper consists of Urdu-related work to English MT and recent MT advancements using different datasets, neural translational approaches, machine learning techniques, and technologies. The main features of each technique or method are presented in Tables 2 and 3.

Table 2: Summary-1 of the literature review

Domain	Ref.	Languages	Dataset	Research findings
Urdu NMT	[28]	English-Urdu	Own English-Urdu dataset	BELU Score
	[29]	English-Urdu English-Hindi	Own English-Urdu dataset	0.5903 and 0.6054 BLEU
	[30]	Roman Urdu-English	Own Roman Urdu dataset	86% and 89% accuracy
	[31]	English-Urdu	Own English-Urdu dataset	Improved BLEU Score
English NMT	[32]	English-German	WMT14, WMT17	54.74 BLEU and 54.15 TER
	[33]	English-Urdu English Hindi	TED Talks, Europarl, News-Commentary	+1.98 and +1 BLEU increase
	[34]	English-German English-French	WMT14	39.06 and 36.06 BLEU
	[35]	English-German English-French	WMT14	41.12 and 41.67 BLEU
	[36]	English-German Chinese-English	WMT14, NIST	Increase in BLEU score
	[37]	English-German English-French	Publicly available	Alignment error rate (27.1%)
	[38]	English-German	WMT14	Increase in BLEU score
	[39]	English-German	WMT16	Increase in BLEU score
Image processing	[38]	Object detection	CIAFR-10, Image-net	87.03% and 88.13% accuracy
	[39]	Object localization	CUB-200-2011	62.29% and 76.97% accuracy

Table 3: Summary-2 of the literature review

Reference	Model used	Research focus	Evaluation parameter
[29]	NNET for rules generation	ANN with L-M algorithm	n-gram BLEU, METEOR score
[30]	LSTM, GRU	NMT implementation	Accuracy, WER, CER, and BLEU
[31]	LSTM and SMT	Comparison of SMT and NMT	n-gram BLEU score
[33]	Transformer	Selective attention approach	n-gram BLEU Score
[34]	GRU, LSTM, RNN-based ATR-NMT	New recurrent network	Accuracy, WER, CER, and BLEU
[35]	LSTM, RNMT+	New modification layers in RNN	Accuracy, WER, CER, and BLEU
[36]	Transformer	Novel attention mechanism	BLEU score, accuracy
[37]	Transformer	Additional alignment layer in RNN	Alignment error rate
[38]	Transformer	Model compression using dropout	BLEU, accuracy
[38]	ResNet, wide ResNet	Targeted dropout	Accuracy
[39]	CNN, wide ResNet	Attention dropout	Accuracy
[40]	Transformer	Drops part of units in each layer	Perplexity, accuracy, BLEU score

2.1 Urdu-to-English MT

In the literature, we can find a few research works attempting to use NNET to translate English to Urdu and vice-versa. Andrabi et al. [28] proposed English to Urdu MT that employed an LSTM-based encoder-decoder model. Analysis of the translation showed improvement in translation quality and BLEU score. Khan et al. proposed a multilingual MT system to translate English into Urdu and Hindi [29]. The proposed system consisted of two modules: one for crafting translation rules and an artificial neural network (ANN) module for translation. Module one takes a sentence in the source language and returns the grammatical structure of the target language sentences. The rules for the source and the target language are then encoded for the ANN. ANN module is implemented using back-propagation with Levenberg-Marquardt (LM) algorithm and is trained on bilingual dictionaries for English-Hindi and English-Urdu language pairs. The system's n-gram BLEU score was 0.5903 for English to Hindi and 0.605 for English to Urdu. The Metric for Evaluation of Translation with Explicit ORdering (METEOR) score was 0.796 for English-Hindi and 0.808 for English to Urdu. The NNET system used by [29] is trained on language rules for both the input and target sentences.

Khan et al. presented a state-of-the-art NMT model for Roman Urdu-to-English and English-to-Roman Urdu transliteration. The implemented solution used the E2E network. Attention-based

Bi-RNN with LSTM unit and GRU-based seq2seq transliteration models with different hyper-parameters were built and trained. To evaluate the model accuracy, they calculated WER, CER, and BLEU. The trained model achieved 86% and 89% accuracy on the transliteration of English to Roman Urdu and vice versa, respectively. The NMT system proposed by [30] was trained on small corpora.

Rauf et al. [31] proposed domain-specific SMT and NMT models to translate English into Urdu. They trained two separate models, SNMT and NMT, using datasets from four different biomedical, religious, technological, and general domains. Trained NMT was a two-layered encoder-decoder model with global attention. They used a standard test set from Indic, Treebank, Transliteration, and FlickrGooglTrans for model evaluation. The BLEU score of the translation produced by the NMT was lower than the corresponding SMT system.

2.2 NMT Models

The artificial intelligence translation software with deep learning ability does not have powerful translation ability but gradually improves translation by manipulating specific parameters and different architectural phenomena. The most crucial factor in NLP and translation systems is the alignment of sentences in different languages, especially for longer sentences, and the use of the attention mechanism as a semantic feature extractor & for capturing long-range dependencies.

Zhang et al. [32] studied the over-translation issue in the vanilla attention network. They investigated the degenerative translation performance due to variance decrease in the context vector, which ultimately led the model incapable of distinguishing different translation predictions. In their work, they proposed an extension to the attention network by inserting an additional gating layer. The proposed model was named the GRU-gated attention model (GRU-GAtt) for NMT. GRU-GAtt can produce translation-sensitive source representations that increase the variance in context vectors, and this increased variance has the discriminating power to predict the next target word. They evaluated GRU-GAtt and its variant and concluded that GRU-GAtt significantly outperforms the vanilla attention-based NMT.

Maruf et al. [33] conducted a detailed study on context-aware NMT. These context-aware models use previous sentences as context during translation. They adopted a selective attention approach to focus on relevant sentences, choose the keywords in those sentences, and then compute the sparse attention dynamically for each query word. They proposed flat attention and integrated them into the encoder or decoder of the Transformer model. Experimental results showed that choosing the selective attention approach improves the overall BLEU score for translating TED Talks, News-Commentary, and Europarl.

Zhang et al. [34] proposed an addition-subtraction twin-gated recurrent network (ATR) for NMT, focusing on simplifying recurrent units in RNN-based NMT. The proposed ATR-NMT is comprised of a single-layer encoder and a decoder. ATR-NMT retains only two weight matrices. One was computed over the input and the other over history. The layer used addition and subtraction operations between the weighted history and input to estimate the input and forget gate. Using this twin-gated mechanism avoids the vanishing gradient problem. It keeps the essential weight metric, making ATR-NMT efficient in physical memory usage and running speed. The results showed that the proposed model yields competitive results compared to regular GRU/LSTM-based NMT.

Chen et al. [35] presented new RNN architecture, RNN-based NMT model (RNMT+). The proposed model was based on GNMT with modification layers. The encoder network of RNMT+ consisted of an additional six bidirectional LSTM layers and regular unidirectional LSTM layers. The outputs of the forward and backward layers were concatenated at the end of each bidirectional layer.

The decoder part of the RNMT+ consisted of eight unidirectional LSTM layers. The context vector was computed through multi-head additive attention, and fed directly into the decoder and softmax layers. The concept of random dropout, label smoothing, and weight decay after certain epochs were used. Upon evaluation, the newly proposed model outperformed all three fundamental architectures on the benchmark WMT'14 dataset. Hao et al. [36] proposed a novel attention mechanism incorporating rich syntactic information in multi-head attention. Generally, attention computed by NMT lacks explicit phrase information. Multi-granularity self-attention combined effective N-gram phrase representation with attention. Experimental results showed that the proposed model captured useful phrase information better than regular multi-head attention. Zenkel et al. proposed a naïve Transformer architecture that defined an additional alignment layer above encoder attention. This extra layer encouraged the attention in the network to learn to attend to source words that correspond to the current target word [37].

Typically NNET based model has a large number of learnable parameters, which sometimes cause over-fitting. Having an enormous number of NNET parameters also causes an increase in the network's computational and storage requirements. Much work has been done on proposing and developing strategies to compress NNET. Among many, one approach is dropout regularization to handle scarification. Gomez et al. [38] introduced a new concept of targeted dropout strategies to compress neural networks. In their proposed methodology, they ranked the weights and units according to some measure of importance. Dropout was then applied to those sets of units that were a-priori considered least useful. They analyzed the model performance with regular dropout, where units were dropped randomly. It was observed that by using targeted dropout, the network became highly robust and easy to implement. It improved test accuracy as well. [39] Proposed the concept of attention dropout for the localization of weakly supervised objects in images. A CNN with an added attention-based dropout layer was implemented on the CUB-200-2011 and ImageNet-1k datasets. The empirical demonstration concluded that adding an attention-based dropout layer has achieved state-of-the-art localization accuracy. Zehui et al. Explored DropAttention idea on a wide range of tasks. They proposed two dropout methods in which they randomly dropped "column" in the attention weight matrix or randomly dropped "element" in the attention weight matrix. They conducted experiments on classification and machine translation tasks and concluded that dropping random attention effectively improves generalization and reduces over-fitting neural networks [40].

It is evident from the literature review that no prior concrete study is available about using the NMT model for Urdu-to-English translation, and work done for the Urdu language has hugely relied on conventional natural language processing (NLP) techniques. The use of deep learning techniques for machine translation in Urdu is still in its inception. Besides, much work has been done on using a neural approach towards English to German and French translation. Several new neural transnational approaches, techniques, and models are proposed [41], but there is still room for improvement in the existing neural translational models. Our proposed system will address the existing gap identified in the literature by adopting the neural approach for Urdu-to-English translation and suggest a novel NMT model for translation using the concept of structural dropout.

3 Methodology

In this section, details of tools and implemented RNN-based NMT model as shown in Fig. 1 and the proposed model as shown in Figs. 2 and 3 have been discussed. Table 4 shows a list of the variables with their description.

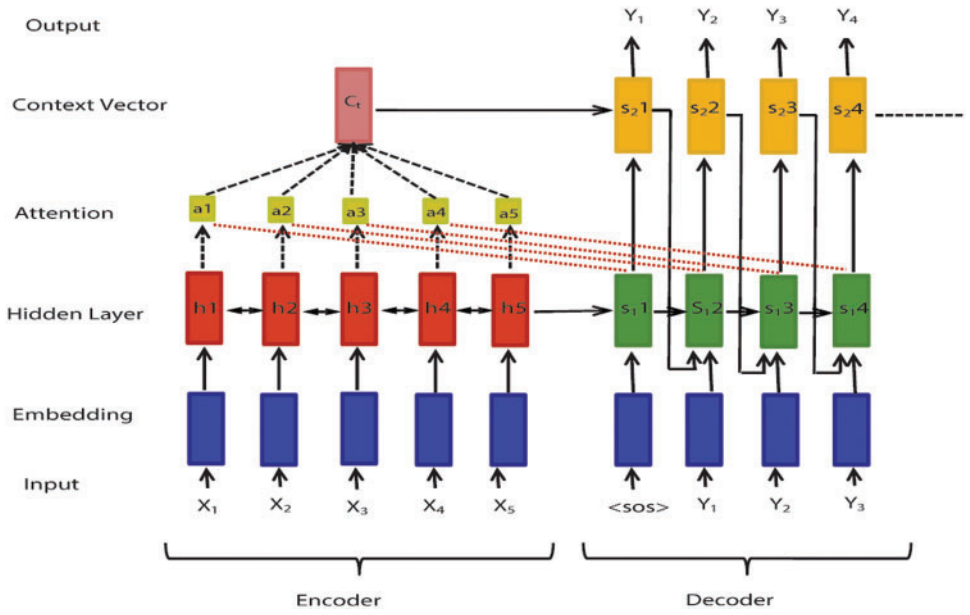
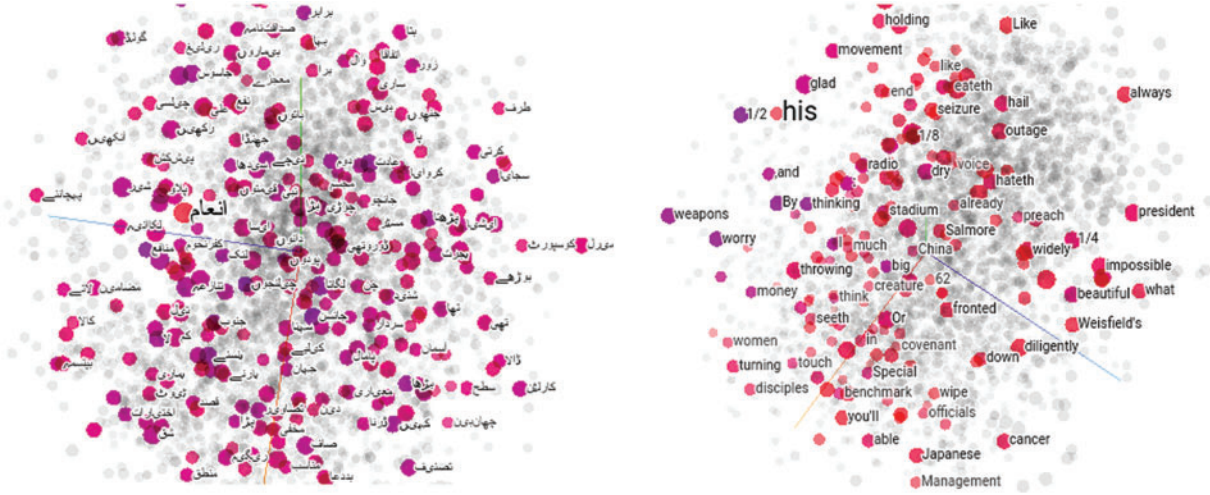


Figure 1: Neural machine translation model (NMT)



(a) Urdu Embedding (b) English Embedding

Figure 2: Urdu-English learned embedding

3.1 Tools

OpenNMT (Open-Source Neural Machine Translation) is an open-source toolkit developed at MIT. Based on pure seq2seq models [42], NMT system is built upon the Torch/Py-Torch mathematical toolkit. This technology maintains model modularity and readability while providing good translation accuracy. The system is user-friendly and supports significant research extensibility. We trained the SRU model on GPU-NVIDIA 160 TITAN Xp 12 GB and all other RNN-based NMT models using

Dell Precision 7920 with 56 core, 62 GB System memory, Intel(R) Xeon(R) Gold 5120 CPU @ 2.20 GHz.

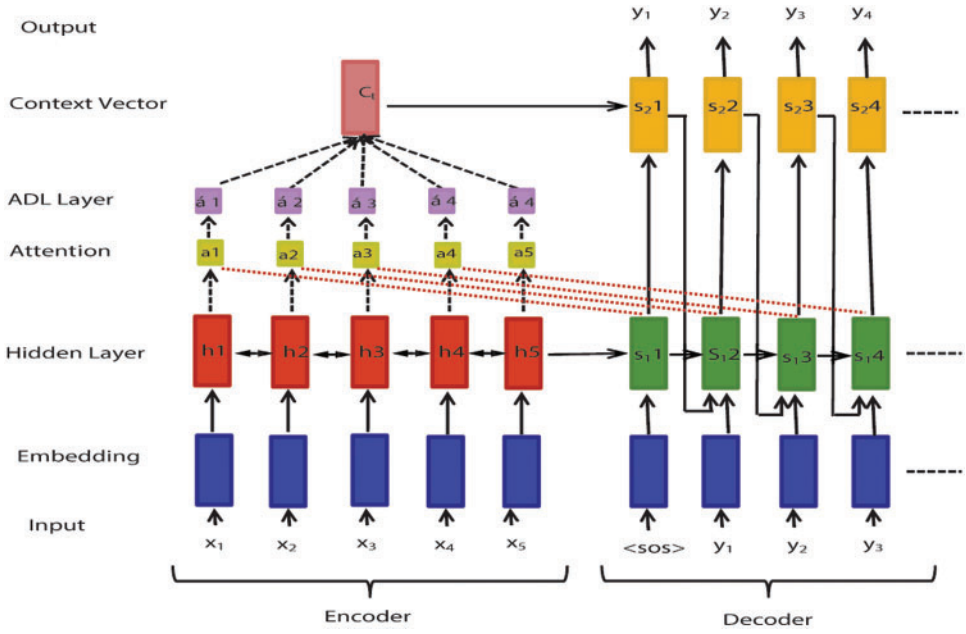


Figure 3: Neural machine translation (NMT) model with an attention-based dropout layer

Table 4: Variables abbreviation and description

#	Variable	Description	#	Variable	Description
1.	x	Source sentence	7.	C	Context vector
2.	y	Target sentence	8.	E	Alignment score
3.	h	Encoder hidden state	9.	$\|V\|$	Vocabulary
4.	s	Decoder hidden state	10.	a	Attention score
5.	α	Alignment model	11.	Emb_{src}	Source embedding
6.	$\acute{a}$	Alignment map at ADL-layer	12.	Emb_{tgt}	Target embedding

3.2 Neural Translation Model

An NMT model with RNN-based encoder-decoder architecture takes source sentence $x = (x_1, x_2,\ldots,x_n)$ represented as word embedding vectors and generates a target sentence $y = (y_1, y_2, ..y_n)$ of length n shown in Fig. 1. The encoder RNN produces an encoding of the source sentence. Encoding of the source sentence provides an initial hidden state for the decoder RNN. Decoder RNN is like a language model that generates a target sentence, conditioned on the source sentence's encoding as in Eq. (1).

$$P(y|x) = P(y_1|x)P(y_2|y_1, x)P(y_3|y_1, y_2, x)\ldots P(y_T|y_1, \ldots, y_{T-1}, x) \quad (1)$$

While $P\ (y|x)$ is the probability of the next target word, given a target word so far and source sentence x.

3.3 Embedding

Each word in the input layer is represented by a real vector called "word embedding", which is produced by embedding each word in the vocabulary into a real space of a fixed dimension. Based on the notion of word similarity, a low-dimensional vector representation represents each word. Similar words have similar vectors and lie close to each other in the vector space. Learned embedding for the Urdu-English dataset is shown in Fig. 2.

3.4 Encoder

The encoder is a bi-directional NNET with recurrent units that read an input sequence $x = (x_1, x_2, \ldots, x_n)$ of length n, pre-process input sequence to generate an embedding matrix. It calculates a forward sequence of hidden state and $\overrightarrow{h} = \left(\overrightarrow{h_1}, \overrightarrow{h_2}, \ldots, \overrightarrow{h_n}\right)$ a backward sequence $\overleftarrow{h} = \left(\overleftarrow{h_1}, \overleftarrow{h_2}, \ldots, \overleftarrow{h_n}\right)$ with the help of the embedding matrix. The encoder RNN consists of either a gating unit, e.g., GRU or LSTM, OR a non-gating unit, e.g., SRU. For a bidirectional RNN, hidden states from feed-forward layer $\overrightarrow{h}$ and feed-back layer $\overleftarrow{h}$ are concatenated to form the annotation vector $\overleftarrow{h_j}$.

$$h_t = RNN([Emb_{src}\, x_j],\ h_{T-1})\ h_1,\ h_2 \text{ are encoder hidden states at time step } t,\ \varepsilon\ R^h \tag{2}$$

Multiple layers can be stacked using each resulting output sequence $h = (h_1, h_2, \ldots, h_n)$ as the input to the next RNN layer. The final hidden state vector contains all the encoded information from the previously hidden representations and inputs. An encoder computes a representation for each source sentence.

3.5 Decoder

The decoder is a separate RNN that generates a target sequence $y = (y_1, y_2, \ldots, y_n)$ word-by-word by computing conditional probability P(y/x).

$$\log P\,(y/x) = \sum_{i=1}^{n} logP\,(yi|y < i, s) \tag{3}$$

The decoder is more like a conditional language model that predicts the next word of the target sentence y, and its predictions are conditioned on the source sentence x. The decoder uses input feeding where a context vector š is concatenated with the previous word's representation as input to the RNN.

$$s_t = RNN\ ([Emb_{tgt}\ y_{t-1},\ s_{t-}{}'_1]\ ,\ s_{T-1})s_1,\ s_2 \text{ are decoder hidden states at time step } t,\ \varepsilon\ R^s$$

Each word y_i is predicted based on a decoder recurrent hidden state s_t, the previously predicted word $y_{(t-1)}$ and, a context vector c_t.

$$s'_t = \tanh\,(watt[s_t,\ c_t] + b) \tag{4}$$

The decoder generates a translation of one word at a time by decomposing the conditional probability as in Eq. (4):

$$p\,(y_t|y_{t-1},\ x) = Softmax\,(V,\ s_t + b) \tag{5}$$

3.6 Attention

NMT captures word alignment through its attention mechanism, which models the probability that y_i is aligned to x_j. For each step, the attention score e^t is computed such that.

$$e^t = [s^T h_1,\ s^T h_2,\ ....s^T h_N]\ \varepsilon\ R_h \tag{6}$$

The weight of each annotation at hidden node h_j is computed through an alignment model $\alpha^t i$. The alignment model is a single-layer feed-forward NNET that the network jointly learned through back-propagation. A softmax function is used to get the attention distribution α^t which is a probability distribution and sums to 1.

$$\alpha^t = Softmax\,(e^t)\ \varepsilon\ R^h \tag{7}$$

α^t takes a weighted sum of the encoder's hidden states to generate the attention output a_t

$$c_t = \sum\nolimits_i \alpha_1^t\,.\,h_j \tag{8}$$

The context vector c_t is computed with an attention mechanism scoring the previous decoder state $s_{(t-1)}$ and each encoder state h_j.

$$a_t = \frac{exp\ \left(score\ \left(s_{t-1},\ h_j\right)\right)}{\sum_k exp\ (score\ (s_{t-1},\ h_k))} \tag{9}$$

$$Score\,(s_{t-1},\ h_j) = s_{(t-1)}^T\ W_{hj} \tag{10}$$

3.7 Output

The output layer is a simple feed-forward neural network with a softmax function. It takes hidden state representation s_t, weights associated with it, the bias as the input, context vector c^t, and produces a vector containing a score for each token in the target vocabulary. Through a linear transformation, these scores can be interpreted as a probability distribution over the target vocabulary |V| that defines an index over target tokens v_j.

$$p\,(y_t = v_j|x,\ y_{<i}) = Softmax\,(g\,(y_{t-1},\ s_t,\ c_t)) \tag{11}$$

During decoding, the beam search strategy is used to choose k hypothesis with the highest scores p (y_t), i.e., the probability of the sequence at each time step. The score is Log probabilities and is negative or 0. The max of two log probabilities is, the greater one is better.

3.8 Objective Function

During training, a decoder takes an input sequence x and selects a certain weight vector w, calculates the scores according to the model, and produces an output y. Generally, a loss function is defined over the network to analyze the effect of the score. The loss function provides a numerical indicator that when a particular weight w is used, how "good" or "bad" a translation is generated. To achieve better translation, the training and testing objective must be defined to reduce the loss. For a given source and target pairs (x, $y_{(1:n)}$), minimize the negative log-likelihood (NLL) of each word independently, condition on gold history ($y_{(1:\,t-1)}$).

$$\text{NLL Loss} = -\sum\nolimits_t \ln \mathrm{p}\,(\mathrm{w_t} = \mathrm{y_t}|\mathrm{y_{1:t-1}}, \mathrm{x},\ \theta) \tag{12}$$

Θ is the hidden states of the encoder and decoder.

3.9 Proposed Attention-Based Dropout Layer

Inspired by the work of [39], a new layer is introduced over the top of the attention layer, as shown in Fig. 3. The dropout layer prevents co-adaptations and generalizes the model to unseen data.

In attention-based architectures, elements are randomly dropped out of the softmax in the attention equation. In contrast to the conventional dropout mechanism, which randomly drops units or connections, this separate Attention-based dropout layer applies a structured dropout method. ADL drops selective parts within attention whose probability exceeds a certain threshold, thus encouraging the model to learn the features with low probability.

The attention-based dropout layer (ADL) has two main hyper-parameters, threshold and drop rate. The drop rate parameter defines the frequency at which the drop mask is applied, and the threshold controls the region's size to be dropped. Using these two parameters, ADL computes a dropped mask and an importance map from input attention.

$$IMap = f\left(\acute{a}_i^t\right) = \frac{1}{1+e^{-\acute{a}_i^t}}$$

$$\text{Drop\_Mask} = (a'')$$

$$\text{Drop\_Mask} = Min\,(a'') \begin{cases} if\ \text{a}'' < Threshold \\ 0 = \qquad Otherwise \end{cases}$$

Drop mask and importance map components are computed from the attention map generated by the softmax layer. Both components play opposite roles. The drop mask penalizes the most discriminative part. The importance map rewards the most discriminative part to increase the prediction power of the model. Penalizing the computed attention part with high probability enables the model to focus on the less discriminating part and cover the sentence integral extent.

Algorithm 1: Proposed Attention-based dropout layer

Input: Attention_score = X, drop_prob = 0.75, drop_thr = 0.90 attn_mdfid
Output: attn_mdfid
function get_imp_map(X)
    return sigmoid(X)
end function
function get_drop_mask(X, drop_thr)
    max_val = max(X)
    thr_val = max_val * drop_thr
    return (X < thr_val)
end function
function select_component(imp_map, drop_mask, drop_prob)
    random_tensor = a random variable uniformly distributed on drop_prob & 1+drop_prob
    binary_tensor = random_tensor converted into float

(Continued)

Algorithm 1: Continued

```
        return (1-binary_tensor ) ∗ imp_map + binary_tensor ∗ drop_mask
    end function
    imp_map = get_imp_map(X)
    drop_mask = get_drop_mask(X, drop_thr)
    final_map = select_component(imp_map, drop_mask, drop_prob)
    attn_mdfid = X ∗ final_map
```

The importance map (IMap) is generated by applying sigmoid activation to the attention. Drop mask (DM) is obtained by using a drop threshold of 0.90 to the attention. The drop mask or importance map is stochastically selected during each training step. The final map is computed by applying either the drop mask OR the importance map randomly. Using a drop rate of 0.75, the selected map is applied stochastically to the attention map by spatial-wise multiplication. The context vector is computed on the newly computed attention map shown in Fig. 4. During the testing phase, the attention-based dropout layer is deactivated.

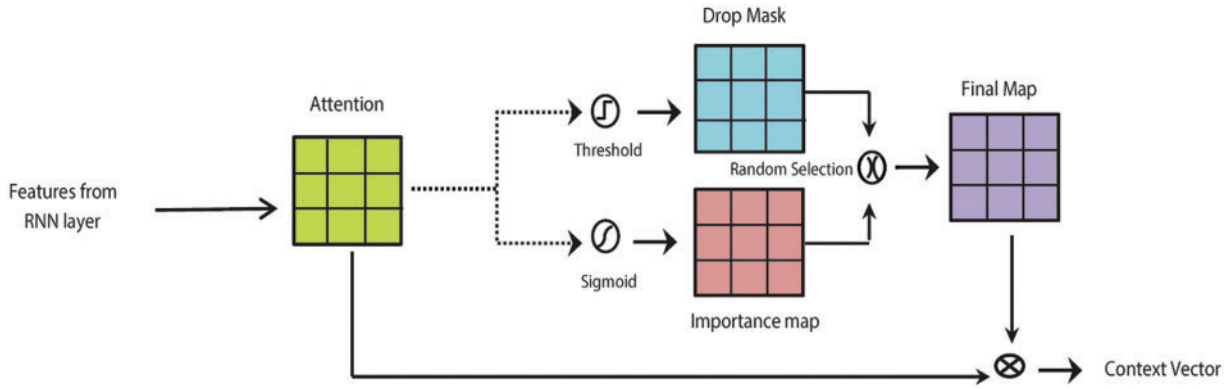


Figure 4: Key details of the attention-based dropout layer

4 Experimental Setup

This part of the paper consists of the details of the data set, trained model, and hyperparameter.

4.1 Dataset

We have developed an Urdu-English dataset repository from freely available online resources for research purposes, e.g., sentences from OPUS: the open parallel corpus [43]. These are the roughest available free corpora. Longer, medium, and short-sized sentences from various fields like sports, politics, religion, science, and education are selected. Cleaning the dataset and making it compatible and sentence aligned is the very first step. Additionally, we have harvested some data by crawling the web where bilingual texts are available. To analyze the behavior of the proposed model, additional small Arabic-English and Persian-English datasets are collected from OPUS, Kaggle [44], and other Internet resources.

Table 5 shows the detail of the Urdu-English datasets used in the experiments and their corresponding vocabulary size. The dataset consists of a 90 K train, 5 k validation, and a 5 k unique test set. Arabic-English and Persian-English datasets consist of 25 k train, 5 k validation, and test sentences.

The source and target sentences are contained in two separate files, a new line separates each sentence, and each line in the files corresponds to each other. Parallel sentences from the train set are shown in Table 6. Translated sentences are tokenized during pre-processing, and a vocabulary is constructed for both the source and target languages. A special token is appended </s> at the end of every sentence, which identifies the end of the sentence. The out-of-vocabulary words are mapped to the special token <unk>.

Table 5: Urdu-English corpus description

Corpus	Language set	Number of sentences	Number of total words	Number of unique words
Train set	Urdu	90000	2335503	29504
	English	90000	2033574	38001
Validation set	Urdu	5000	113181	8714
	English	5000	99406	10060
Test set	Urdu	5000	135744	6863
	English	5000	116364	9093

Table 6: Example sentences from Urdu-English train set

Source sentence	Target sentence
ہم جیت گئے ۔	We won.
میں موسم بہار کی چھٹیوں کے دوران کام کرنے جارہا ہوں ۔	I am going to work during the spring vacation.
اگر تم اپنے کپڑوں کا دھیان رکھو گے تو وہ ز یادہ عرصہ چلے گے ۔	If you take care of your clothes, they will last longer.
پاکستان کی تشکیل میں جناح کی اہمیت یادگار تھی ۔	Jinnah's importance in the creation of Pakistan was monumental.

4.2 Encoder Types

We have trained six different RNNs, each with LSTM units, Bi-directional LSTM units, GRUs, CNN, SRU, or Transformer. The primary objective of the experiment is to fairly compare all NMT models and their performance on the Urdu-to-English dataset. The standard parameters of all trained models are described in Table 7.

Table 7: Standard parameter of all NMT models used in the experiment

S.No.	Common parameter for all trained model	Configuration
1.	Number of stacked encoder/decoder layers	2
2.	Number of hidden units	512
3.	Word embedding (size of the dictionary of embedding)	512
4.	Dropout ratio	0.3
5.	Batch size	64
6.	Decay rate	0.5
7.	Train steps	10000
8.	Validation steps	5000

4.2.1 Long Short-Term Memory

LSTM, the special architecture of RNN, proposed by [45], is an efficient gradient-based algorithm. It is capable of learning long-term dependencies quickly and remembers information over longer time steps. An LSTM cell consists of four regulated gates that control the flow of information within the cell.

We have built an RNN-based encoder-decoder model with global attention proposed by [25], and used an LSTM-RRN for the encoder and standard fully batched RNN decoder with attention. Global attention takes a matrix and a query vector and computes a parameterized convex matrix combination based on the input query using the dot product. Stochastic gradient descent is used as an optimizer with an initial learning rate of 1. For translation output, beam search is used. It keeps track of the k-most probable partial translations on each step. Google Neural Machine Translator (GNMT) global scorer is used for ranking output. The model was refined until the accuracy and perplexity score did not change much on the development set.

4.2.2 Gated Recurrent Unit

Proposed by Cho et al. [23], the main idea behind GRU was to make each recurrent unit adaptively capture different time scales' dependencies. GRU is simpler than LSTM and has fewer parameters. Like LSTM, the GRU has two gating units that modulate the flow of information inside the unit. Unlike LSTMs, GRUs does not need the cell layer to pass values along. An RNN-based encoder-decoder model with a global attention mechanism that uses a GRU-RRN for the encoder was trained. Adam optimizer is used, with an initial learning rate at 0.00100 and a decay rate of 0.5 at decay step 50000. The used batch size is 64.

4.2.3 Bi-Directional RNN

Bi-RNN is an NMT model in which the forward and backward LSTM layers are stacked on top of each other, that read the exact feature representations from a single feed-forward neural network. For the encoder part of the Bi-RNN model, LSTM units with global attention are used. Stochastic gradient descent with an initial learning rate (a) $= 1$ is used as an optimizer. The learning rate was halved after every 50 percent steps.

4.2.4 Convolution Neural Network

A CNN-based encoder-decoder is an NMT model that encodes the sentence to real-valued vectors. A convolutional layer extracts local features by sliding a window of a specified length over the sentence and performs the convolution operation within each sliding window. Afterward, it manipulates all local features via a max-pooling operation and a hyperbolic tangent function to obtain fixed-sized sentence vectors. For experimentation, a CNN-based encoder-decoder model is used. CNN encoder-decoder consists of 2 gated convolution layers with multi-step attention and a window of size (3, 1), stride = (1, 1) and padding = (1, 0). A key matrix calculates attention weight with the query vector and sums it on the value matrix. The dropout ratio for the network is set to $p = 0.3$. The initial learning rate is 0.00100 with a decay rate of .0005.

4.2.5 Statistical Recurrent Unit

SRU is a unique encoder-decoder architecture that uses an un-gated SRU Cell to learn long-term dependencies and compute the likelihood of the words in source and target languages. SRU Unit, also called simple summary statistic [46], calculates the exponential moving average over time. We built an un-gated SRU-based encoder-decoder model with global attention with tanh and RelU activation. The encoder consists of 512-dimensional SRU units. We used the Adam optimizer, with an initial learning rate of 0.001 and a decay rate of 0.5 at the decay step.

4.2.6 Transformer

A Transformer model consists of a stacked RNN encoder and a decoder block. The encoder block of a Transformer is composed of a two layers feed-forward network with residual norm layer and multi-head self-attention. Word embedding with positional encoding is used as input to the network. The model Transformer uses a separate attention and context dropout mechanism. We have trained a four layers Transformer with multi-head attention and positional encoding. Adam and adam_beta2 with value 0.998 are used as optimizers with an initial learning rate of 2, dropout ratio of 0.1, and Noam decay method.

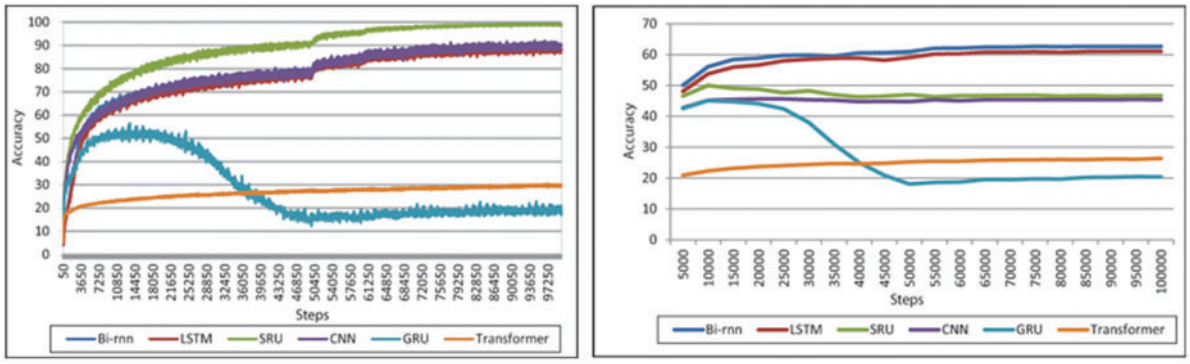
4.3 Experiment Observation

Each model's learning behavior was observed, and step-wise accuracy and perplexity (PPL) were reported during the training and validation process. BLEU score was used to evaluate whether the features increase the translation performance and the translation quality of the NMT system. For neural MT, perplexity is a useful measure of how well the model can predict a reference translation given the source sentence. It is the indicator of whether input features provide benefits to the models. We reported the best validation set perplexity of each experiment. The training and validation accuracy of each model is shown in Figs. 5 and 6.

Bi-RNN, LSTM, CNN, and SRU show a smooth curve, while GRU shows a significant drop in the training and validation curve accuracy. The validation perplexity of GRU and SRU show many fluctuations and a high increase in PPL score in the training curve.

Bi-RNN and LSTM achieved 62.6% and 61% accuracy on the development set and 49.67 and 47.14 BLEU scores, respectively, as shown in Table 8. On the final training step, SRU and CNN achieved 46.6% and 45% accuracy. Table 9 shows the trained model results on the test set. The system achieved the MT-evaluation scores for around 5000 test Urdu sentences translated into English. The average n-gram BLEU score is 49.67 for Bi-RNN, 47.14 for LSTM, 0.77 for GRU, 21.80 for CNN, and 28.61 for SRU. GLUE-Corpus score was 0.506 for Bi-RNN, 0.485 for LSTM, 0.051 for GRU, and

0.196 for CNN. METEOR score achieved was 0.429 for Bi-RNN, 0.413 for LSTM, 0.064 for GRU, and 0.209 for CNN. The system's best n-gram BLEU score was 49.67, GLUE Corpus score was 0.506, METEOR score was 0.429, and Rouge score of 0.732 by Bi-RNN. The Bi-RNN and LSTM models outperform SRU, GRU, CNN, and Transformer.



(a) Training Accuracy of NMT models (b) Validation Accuracy of NMT models

Figure 5: Training and validation accuracy of the baselines system over 100 k steps

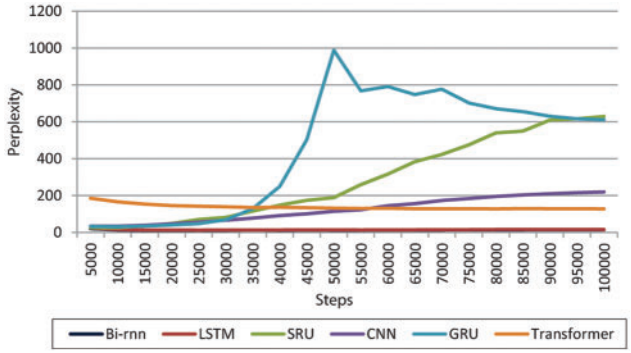


Figure 6: Perplexity of the baseline systems on validation step

Table 8: Validation accuracy and perplexity on final training step

Test sentence	Model	Bi-RNN	LSTM	CNN	GRU	SRU	Transformer
Validation set	Accuracy	62.633	61.039	45.361	20.361	46.672	26.164
	Perplexity	14.771	16.070	221.11	610.572	629.27	137.945

It is observed from the proposed system results that the presence of spelling mistakes, grammatical mistakes, and single-word translations having different word representations in the target set affects the model accuracy. For example, translation for the word حکومت, "The Government" and "The government" generates two diffident feature vectors. If case marking is improved in the train set, it will produce more efficient results.

Table 9: Trained models complete statistics on the test set, ∗ represents the best-achieved score by any model

Result on test-set	Bi-RNN	LSTM	CNN	GRU	SRU	TRF
Translation time in Min	40*	43	205	39	–	48
BLEU score	49.67*	47.14	21.80	00.77	28.61	1.27
GLUE-corpus score	0.506*	0.485	0.196	0.051	0.308	0.039
GLEU sentence average score	0.530*	0.511	0.287	0.078	0.342	0.049
ROUGE_L	0.732*	0.715	0.474	0.234	0.565	0.156
METEOR	0.429*	0.413	0.209	0.064	0.337	0.048
Over-translation OTEM/2	0.031*	0.033	0.397	0.044	0.233	0.249
TER	0.342*	0.361	0.826	0.826	0.764	1.005
WER	9.000*	10.00	21.00	21.00	20.00	24.00
Precision:	0.807*	0.794	0.512	0.455	0.555	0.172
Recall:	0.789*	0.768	0.395	0.131	0.693	0.107
f1:	0.798*	0.781	0.446	0.203	0.617	0.132
fMean:	0.791*	0.772	0.409	0.147	0.668	0.114
Bleu_1:	75.80*	74.00	37.40	05.50	52.90	12.80
Bleu_2:	64.70*	62.50	30.10	02.50	41.60	04.20
Bleu_3:	56.40*	54.00	25.40	01.30	34.10	02.10
Bleu_4:	49.67*	47.14	21.80	00.80	28.61	01.27

5 Results & Discussion

In this section, we discuss the results of the NMT model with an added attention-based dropout layer.

Table 5 shows the results of training the NMT model with an attention-based dropout layer. The NMT models with the added attention-based dropout layer show improvement over baseline systems, especially in GRU. The perplexity and accuracy of GRU, GRU-ADL during training are shown in Figs. 7 and 8 shows much less fluctuation in the training curve.

During training, the recorded validation set accuracy for GRU is 20.36%, and GRL-ADL is 60.48%. GRU-ADL shows a 40% increase in accuracy, making it an almost competitive alternative to Bi-RNN and LSTM. Adding ADL to SRU contributed to the model validation accuracy by increasing 1.2% and the Transformer 0.2%. The new model with added ADL further outperforms in terms of perplexity. Summarized in Table 10 Bi-RNN, LSTM, perplexity decreases by 1 PPL point. Likewise, the perplexity of CNN decreases by almost 2 PPL points.

The total real-time required to translate a test set of 5000 sentences is computed on the same machine. It was observed that NMT models with added attention-based dropout layer show a decrease in the translation speed, as shown in Table 11.

Adding a new attention-based dropout layer has contributed to the over-translation OTEM/2 score on the test set. Table 12 represents the OTEM/2 score obtained on the test set. All NMT models with added new layers showed a reduction in OTEM/2 score, especially in GRU. The over-translation

score for GRU was reduced from 0.044 to 0.034. Additionally, it was noted that the under-translation UTEM/4 (under-translation evaluation metric) [47] for GRU was reduced from 1.88 to 0.52, and for Transformer, it was reduced from 1.354 to 0.942. The UTEM/4 score can be seen in Table 13.

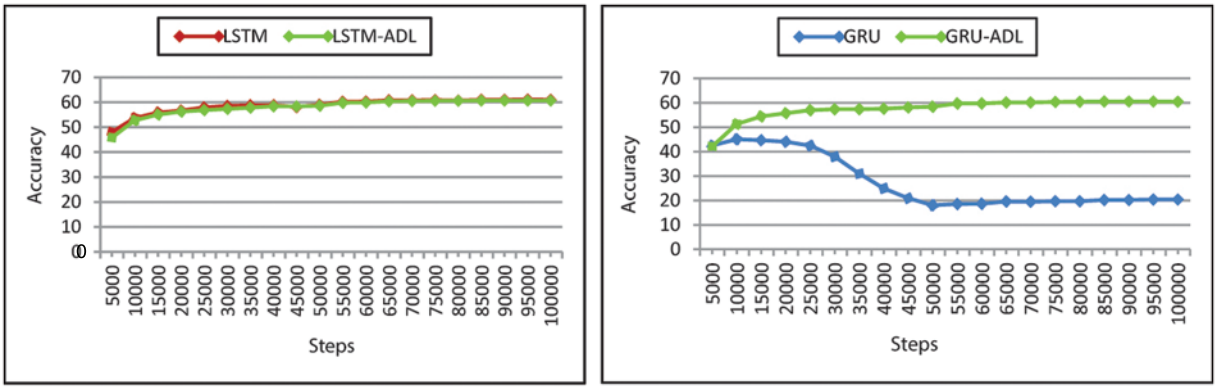


Figure 7: Comparison of the validation accuracy of the NMT models and NMT models with attention-based dropout layer

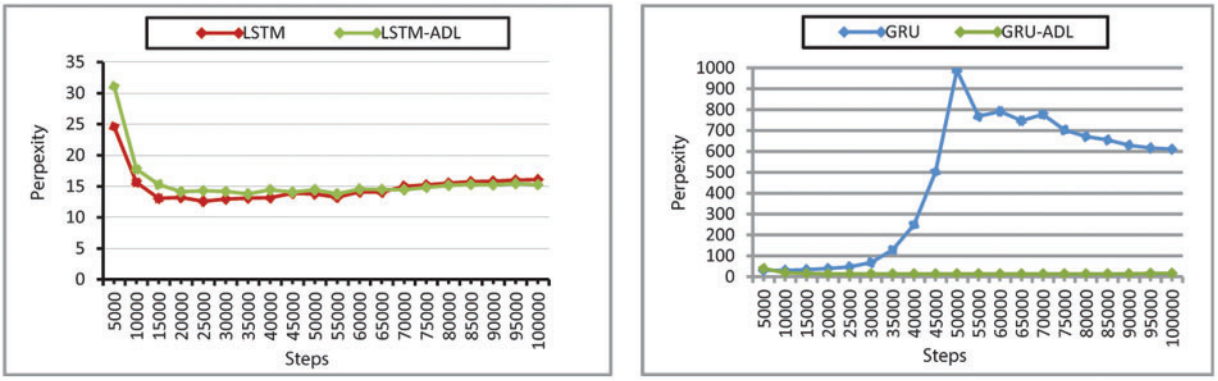


Figure 8: Comparison of the validation perplexity of the NMT and NMT models with attention-based dropout layer

Table 10: Accuracy and perplexity of NMT models and NMT models with attention-based dropout layer on final training step. ↑ represent an increase while ↓ represent the decrease in the score

	NMT model	Bi-RNN	LSTM	CNN	GRU	SRU
Accuracy	Without ADL	62.633	61.039	45.361	20.361	46.672
	With ADL layer	62.549	60.718	41.345	60.48↑	47.75↑
Perplexity	Without ADL	14.771	16.070	221.11	610.572	629.27
	With ADL layer	13.629↓	15.23↓	219.25↓	15.69↓	326.9↓

Table 11: Translation time in minutes taken by each model on test set

NMT model	Bi-RNN	LSTM	GRU	CNN	Transformer
Without ADL	40	43	39	205	48
With ADL layer	39↓	41↓	38↓	184↓	55

Table 12: Over-translation OTEM/2 score of each model on test set

NMT model	Bi-RNN	LSTM	GRU	CNN	SRU	Transformer
Without ADL layer	0.031	0.033	0.044	0.397	0.249	0.233
With ADL layer	0.030↓	0.032↓	0.034↓	0.380↓	0.119↓	0.616↑

Table 13: Over-translation OTEM/2 score of each model on test set

NMT model	GRU	Transformer
Without ADL layer	1.879	1.354
With ADL layer	0.515↓	0.942↓

Table 14 summarizes the proposed model's statistics and performance on the test set. With added ADL, GRU, SRU, and Transformer have improved BLEU, GLEU ROUGE_L, METEOR, Precision and Recall score, and a decrease in TER and WER Score. The n-gram BLEU, GLUE, precision, and Recall scores are slightly reduced by Bi-RNN, LSTM, and CNN.

Table 14: Trained NMT models with attention-based dropout layer complete statistics on test set. ↑ and ↓ represent increase or decrease in the score in comparison to the baseline system

Result on test-set	Bi-RNN-ADL	LSTM-ADL	CNN-ADL	GRU-ADL	TF-ADL	SRU-ADL
BLEU score	47.26	44.69	21.54	44.8↑	2.02↑	32.4↑
GLUE-corpus	0.490	0.475	0.192	0.467↑	0.040↑	0.342↑
GLUE-sentence average score	0.515	0.491	0.279	0.494↑	0.059↑	0.364↑
ROUGE_L score	0.723	0.703	0.466	0.703↑	0.181↑	0.578↑
METEOR	0.419	0.407	0.204	0.407↑	0.060↑	0.333
TER	0.344	0.369	0.859	0.371↓	1.257	0.636↓
WER	9.000	10.00	21.00	10.00↓	31.00↓	16.00↓
Precision:	0.802	0.789	0.510	0.786↑	0.138↑	0.637↑
Recall:	0.777	0.760	0.387	0.761↑	0.141↑	0.668

(Continued)

Table 14: Continued

Result on test-set	Bi-RNN-ADL	LSTM-ADL	CNN-ADL	GRU-ADL	TF-ADL	SRU-ADL
f1:	0.789	0.774	0.440	0.773↑	0.139↑	0.652↑
fMean:	0.781	0.764	0.402	0.765↑	0.140↑	0.663↑
Bleu_1:	73.80	72.10	36.70	72.20↑	16.20↑	60.50↑
Bleu_2:	62.60	60.30	29.40	60.50↑	06.60↑	47.30↑
Bleu_3:	54.12	51.70	24.67	51.80↑	03.50↑	38.70↑
Bleu_4:	47.26	44.69	21.54	44.80↑	02.02↑	32.40↑
BLEU score	47.26	44.69	21.54	44.8↑	2.02↑	32.4↑

A potential explanation of the results is that all NMT models are trained to reduce word-level negative log-likelihood (NLL) but evaluated with a BLEU-like metric that is based on n-grams overlap between the output and reference translations. These measures only give us a good score when the trained model generates an expected output and shows a reduction in the score even when a valid but unexpected translation is produced. The result for the three exemplary sentences and the produced translation are shown in Tables 15–17.

Table 15: Sentence 1 translation

SENT: 1	ملک میں لانگ مارچ روکنے کے لئے ہزاروں پولیس کارکنوں کو تعینات کیا ۔ حکومت نے	Ngram BLEU
GOLD: 1	The Government deployed thousands of Police workers to stop long march in the country.	Score
Bi-RNN	The government deployed thousands of police workers to put a long march in the country.	36.70
Bi-RNN-ADL	The government deployed thousands of police workers to stop long march in the country.	35.63
LSTM	The government deployed thousands of police workers in order to stop long march stop in the country.	21.77
LSTM-ADL	The government deployed thousands of policemen to stop long march in the country.	26.17
GRU	The government will be a great deal on Russia?	0.00
GRU-ADL	The Government deployed thousands of policemen to halt long march in the country.	0.002
CNN	The government deployed has deployed to prevent thousands of police workers to prevent the thousands of police workers to prevent the thousands of police workers to prevent a long march to prevent long.	0.00
CNN-ADL	The government set for thousands in to the country.	0.005

For the exemplary sentence shown in Table 15, "حکومت نے ملک میں لانگ مارچ روکنے کے لئے ہزاروں پولیس کارکنوں کو تعینات کیا" translation produced by Bi-RNN is "The government deployed thousands of police workers to put a long march in the country", and translation produced by Bi-RNN using ADL is "The government deployed thousands of police workers to stop long March in the country". The n-gram BLEU score for the translation of Bi-RNN is 36.70 and the n-gram BLEU score for the translation of Bi-RNN using ADL is 35.63. Despite the low BLEU score, translation produced by Bi-RNN using ADL is more valid and nearer to the gold standard.

For sentence 2, Table 16 translation produced by CNN and CNN-ADL model has n-gram BLEU scores 0.00 and 0.005. The METEOR score for CNN-produced translation is 0.380, and the METEOR score for CNN-ADL produced translation is 0.282. Even with an improved BLEU score, for CNN-ADL METEOR score shows a reduction.

Table 16: Sentence 2 translation

SENT: 2	کیا آپ تھوڑا سا اونچی آواز میں بول سکتے ہیں ؟	Ngram BLEU
GOLD: 2	Can you speak a little louder?	
Bi-RNN	Can you speak a little high voice?	54.11
Bi-RNN-ADL	Can you speak a little bit loudly?	54.11
LSTM	Can you speak more loudly?	0.006
LSTM-ADL	Can you speak a little more loudly?	54.11
GRU	Can you speak?	0.006
GRU-ADL	Can you speak a bit high?	43.47
CNN	Speak in you a bit more loudly?	0.00
CNN-ADL	Can you speak out loudly.	0.005

For sentence 3, translation in Table 17 produced by CNN model has an n-gram BLEU score 0.00 and the CNN-ADL model BLEU score is 0.002. For sentence 3: 1-gram BELU score is 0.251 for CNN and 0.286 for CNN-ADL. GLUE Corpus and average sentence scores are 0.121 for CNN translation and 0.152 for CNN-ADL translation, METEOR score for CNN translation is 0.189, and for CNN-ADL translation, 0.105. For CNN translation Precision, Recall, f1, fMean score is 0.914, 0.4, 0.557 and 0.437, respectively. For CNN-ADL translation Precision, Recall, f1, fMean score is 0.539, 0.219, 0.311, and 0.240. Although there is an apparent increase in the n-gram BLEU score, 1-gram BLEU score and GLUE Corpus and average sentence score for CNN-ADL produced a translation of sentence 3. METEOR, Precision, Recall, f1, fMean score shows a reduction for CNN-ADL.

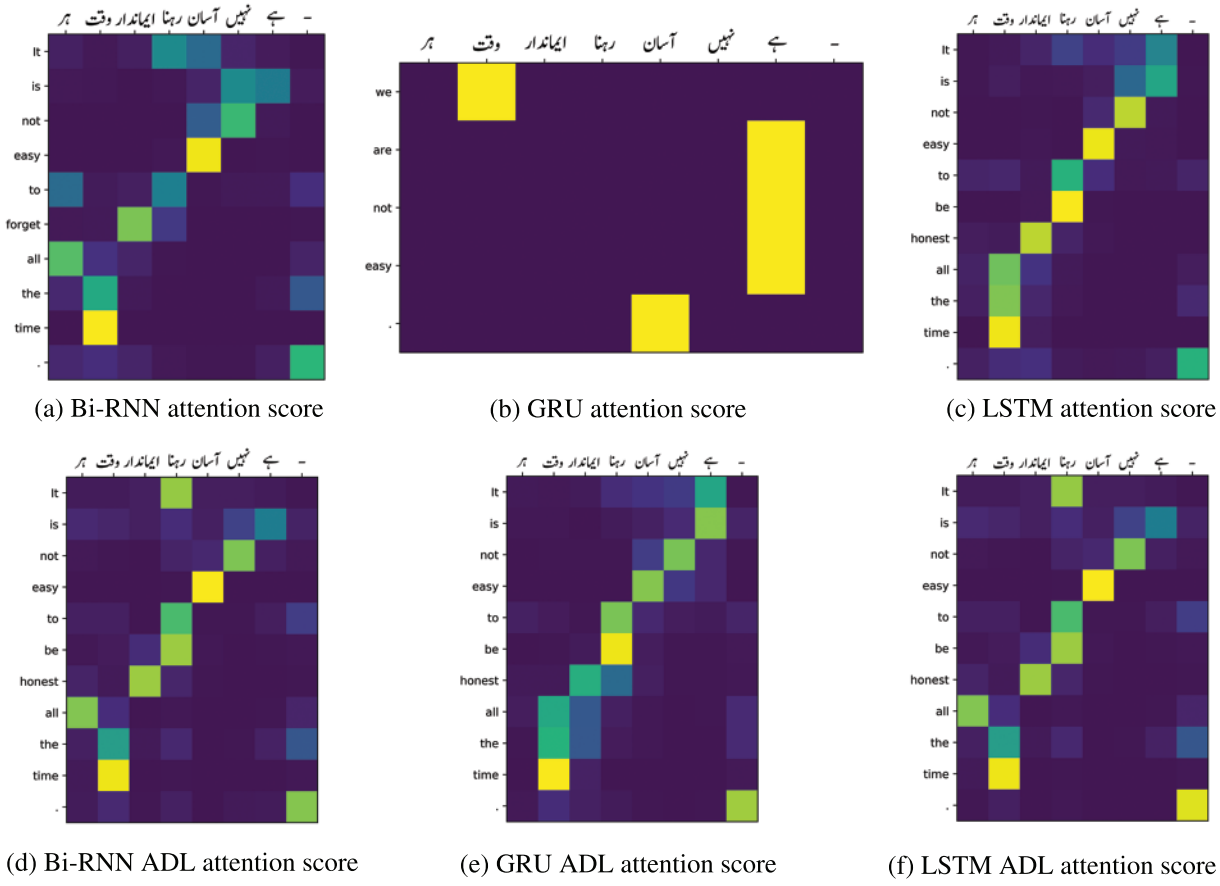
Table 17: Sentence 3 translation

SENT: 3	کلاسیکی معاشیات میں ، یہ خیال کیا جاتا ہے کہ معیشت بنیادی طور پر دولت کے علم کے بارے میں ہے ۔	Ngram BLEU
GOLD: 3	In classical economics, it is believed that the economy is primarily about the knowledge of wealth.	Score
Bi-RNN	In the classic economics, it is believed that the economy is mainly about wealth knowledge.	51.39
Bi-RNN-ADL	In classic economics, it is believed that the economy is mainly about wealth knowledge.	50.55
LSTM	In the classic economics, it is thought that the economy is mainly about knowledge.	30.51
LSTM-ADL	In classic economics, it is thought that the economy is mainly about the knowledge of wealth.	54.75
GRU	The beautiful has been also	0.00
GRU-ADL	& In classic economics, it is believed that the economy is mainly about wealth.	50.95
CNN	In the classic economics of the wealth.	0.00
CNN-ADL	In the classic situation, it is about.	0.002

Fig. 9 is the visual representation of attention weights at the final layer for the sentence "رہنا آسان نہیں ہے ہر وقت ایماندار" from the test set with reference translation "It is not easy to be honest all the time".

The attention score shows the clearest alignment for Bi-RNN and LSTM. For Bi-RNN in Fig. 9a shows high alignment scores on "ہر", "وقت", "ایماندار" and "آسان". Bi-RNN shows a low attention score for words "رہنا", "نہیں" and "ہے". The words "رہنا", "نہیں" and "ہے" in the same sentence is attended by Bi-RNN with ADL and have produced higher attention scores for corresponding translation "it", "is", "not", and "be", thus shifting the heat to the exact position, showing strong patterns along the diagonal, and has produced the most apparent alignment then Bi-RNN.

The LSTM and LSTM with ADL have produced the same correct translation, "It is not easy to be honest all the time." for the test sentence. The attention score of LSTM and LSTM with ADL show clear alignment and higher scores at exact positions as in Figs. 9c and 9f. The LSTM with ADL, in comparison to LSTM has reconsidered and strengthened all the words by increasing their attention score, except for the word "رہنا". The translation of which is assigned to the end of the sentence rather than to "to". The translation produced by the GRU for the same test sentence is "We are not easy." Compared to the reference translation, it suffers from under-translation. In Fig. 9, the attention score produced by GRU shows incorrect translation for all words, poor alignment, and lack of a clear structure for the whole sentence. The GRU has produced the correct output with the ADL model, re-ordered the position, and reflected a high score for all the exact words and their translation.



(a) Bi-RNN attention score (b) GRU attention score (c) LSTM attention score

(d) Bi-RNN ADL attention score (e) GRU ADL attention score (f) LSTM ADL attention score

Figure 9: Attention score of an exemplary sentence at the final layer

We investigated the effect of adding an attention-based dropout layer in NMT models for Arabic-English and Persian-English translation and reported the results in Tables 18 and 19. From these results, we can see that applying ADL increases the performance of the models by contributing to both perplexity and accuracy. For the Arabic-English dataset, we find the perplexity of the NMT model without ADL on the validation set is 142.94, 133.31, 599.5, 15620.5 for Bi-rnn, LSTM, GRU, and CNN, respectively. Adding ADL layer to the models, the recorded validation set perplexity decreases by almost 21.33, 15.02, 262, and 3237.3 PPL points for Bi-rnn, LSTM, GRU, and CNN. We additionally find that the ADL can improve the accuracy of GRU. During training, the recorded validation set accuracy for GRU is 38.24%, and GRL-ADL is 40.29%. GRU-ADL shows a 2.05% increase in accuracy.

For the Persian-English dataset adding ADL, the recorded validation set perplexity is decreased by almost 136.48, 770.1, and 6691.3 PPL points for Bi-rnn, GRU, and CNN, respectively. During training, the recorded validation set accuracy for GRU is improved by 1.59%.

On Arabic-English and Persian-English datasets, Tables 20 and 21 provide the qualitative evaluation of translated sentences. From the results, we consistently observe that the NMT model with ADL captures the word alignment better than the NMT model without ADL. For example, as seen from the translation of the Persian sentence, the translation produced by the LSTM is “I need a lot of cloth to make a good dress”. The translation produced by LSTM with ADL, “I need a lot of cloth to make a

long dress." is more meaningful and near to the ground truth in the target sentence. For the exemplary sentence from the Arabic dataset shown in Table 17, the translation produced by GRU is "Can you speak slowly?" and the translation produced by GRU-ADL, "Can you speak slowly, please?". The GRU has produced the correct output with the ADL model and reflected a high score for all the exact words and their translation.

Table 18: Accuracy and perplexity of NMT models and NMT models with attention-based dropout layer on final training step for Arabic dataset. ↑ represent an increase while ↓ represent the decrease in the score

	NMT model	Bi-RNN	LSTM	GRU	CNN
Accuracy	Without ADL	45.80	45.31	38.24	39.22
	With ADL layer	45.19	44.58	40.29 ↑	35.77
Perplexity	Without ADL	142.94	133.31	599.5	5620.5
	With ADL layer	121.61↓	118.29↓	337.5↓	2383.1↓

Table 19: Accuracy and perplexity of NMT models and NMT models with attention-based dropout layer on final training step for Persian dataset. ↑ represent increase while ↓ represent decrease in the score

	NMT model	Bi-RNN	LSTM	GRU	CNN
Accuracy	Without ADL	34.97	34.9	30.81	28.58
	With ADL layer	35.48↑	33.8	32.4↑	24.77
Perplexity	Without ADL	595.57	501.29	1284.8	7843.7
	With ADL layer	459.09↓	521.75	514.7↓	1152.4↓

Table 20: Persian-English example sentence translation

SENT: 1	من برای درست کردن یک لباس بلند به مقدار ز یادی پارچه احتیاج دارم .	Ngram BLEU
GOLD 1:	I need a lot of cloth to make a long dress.	Score
Bi-RNN	I need a lot of cloth to make a lot of dress.	70.16
Bi-RNN-ADL	I need a lot of cloth to make a long dress.	1.00
LSTM	I need a lot of cloth to make a good dress.	76.92
LSTM-ADL	I need a lot of cloth to make a long dress.	1.00
GRU	I need a lot of cloth to make a dress.	77.73
GRU-ADL	I need a lot of cloth to make a long dress.	1.00
CNN	I need a lot of cloth to need a week.	55.49
CNN-ADL	I need a lot of cloth to make a little cloth.	72.93

Table 21: Arabic-English example sentence translation

SENT: 1	يمكنك أن تتحدث ببطئ من فضلك ؟	Ngram BLEU
GOLD 1:	Can you speak slowly, please?	Score
Bi-RNN	Can you speak slowly, please?	64.35
Bi-RNN-ADL	Can you speak somewhere, please?	61.14
LSTM	Can you tell me slowly?	0.00
LSTM-ADL	Can you speak of tea, please?	0.005
GRU	Can you speak slowly ?	53.20
GRU-ADL	Can you speak slowly, please?	64.35
CNN	Can you speak Japanese?	0.006
CNN-ADL	Can you speak Japanese well?	0.006

Table 22 shows the trained model results on the test set. The MT-evaluation BLEU scores achieved by the system for around 5,000 test Persian sentences translated into English Bi-RNN, GRU, and CNN showed +0.21, +0.94, and +0.49 points improvement over the NMT model without attention-based dropout layer.

Table 22: BLEU score comparison of Arabic-English and Persian-English NMT Model on test set

Language-set	NMT model	Bi-RNN	LSTM	GRU	CNN
Persian-English	Without ADL	18.02	16.92	16.47	13.62
	With ADL layer	18.23	17.47	17.41	14.11
Arabic-English	Without ADL	17.37	20.23	13.95	11.57
	With ADL layer	19.54	18.36	17.97	13.06

For 5,000 tests, Arabic sentences translated into English Bi-RNN, GRU, and CNN showed +2.17, +4.04, and +1.49 points improvement, respectively. Surprisingly, adding an attention-based dropout layer to Bi-RNN and GRU contributed to the perplexity, accuracy, and BLEU score for Arabic and Persian translation to English.

6 Conclusion & Future Work

In this paper, we explored the different architectural implementations of the NMT models designed to enable barrier-free access and multilingual translation support for less privileged languages. This work aims to unearth research opportunities to address issues and challenges using the NMT model for Urdu language translation. NMT models with attention mechanisms trained iteratively, with scalable datasets, make precise predictions on unseen data and yield competitive results by achieving high accuracy and good BLEU Score. However, these state-of-the-art well-trained NMT models tend to produce repetitive output more frequently and challenge the existing greedy approximate inference algorithms. Attention scores produced by Bi-RNN and LSTM produces clear

alignment, while GRU shows incorrect translation for words, poor alignment, and lack of a clear structure. We consider refining the attention-based models by defining an additional attention-based dropout layer. Unlike the concept of random dropout, which focuses on randomly dropping part of units/elements or complete attention, these targeted dropouts from within attention handle failed word alignment and over-translation issues and helped to generalize the model on unseen data. Empirical demonstration and comparison with counterparts show improvement in the resulting translation system's quality and decreased perplexity and over-translation score. We demonstrated that the newly proposed architecture's translation quality surpasses the general NMT model with an attention mechanism. Although the NMT model with added attention-based dropout layer outperforms other NMT models with attention mechanisms, it still suffers from under translation. We hope our work will motivate NMT researchers to investigate the problem further and propose new techniques. Secondly, based on our findings translation results produced by the Transformer are the worst. Therefore, another important avenue for future work is to identify a set of optimal parameters for the Transformer to produce better translation for Urdu-English datasets. We believe these recommendations will be positively received by the NMT research community.

Funding Statement: This work was supported by the Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Universiti Teknologi Mara, Shah Alam, Selangor. Malaysia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Simons and C. Fennig, *Ethnologue: Languages of Asia*, 20th ed., Dallas, Texas: SIL International Dallas, 2017.

[2] A. Fraisse, Z. Zhang, A. Zhai, R. Jenn, S. F. Fishkin *et al.,* "A sustainable and open access knowledge organization model to preserve cultural heritage and language diversity," *Information-An International Interdisciplinary Journal*, vol. 10, no. 303, pp. 10, 2019.

[3] S. Mukund, G. Debanjan and R. Srihari, "Using cross-lingual projections to generate semantic role labeled annotated corpus for Urdu-a resource poor language," in *Proc. 23rd Int. Conf. on Computational Linguistics (Coling 2010)*, Beijing, China, Association for Computational Linguistics, pp. 797–805, 2010.

[4] H. Kaji, "An efficient execution method for rule-based machine translation," in *Proc. 12th Conf. on Computational Linguistics (Coling '88), Association for Computational Linguistics*, USA, vol. 2, pp. 824–829, 1988.

[5] H. Masroor, M. Saeed, M. Feroz, K. Ahsan and K. Islam, "Transtech: Development of a novel translator for Roman Urdu to English," *Heliyon*, vol. 5, no. 5, pp. e01780, 2019.

[6] N. Ata, B. Jawaid and A. Kamran, "Rule based English to Urdu machine translation," in *Proc. Conf. on Language and Technology (CLT'07)*, Peshawar, Pakistan, 2007.

[7] S. C. Kak, "The paninian approach to natural language processing," *International Journal of Approximate Reasoning*, vol. 1, no. 1, pp. 117–130, 1987.

[8] J. Hutchins, "Example-based machine translation: A review and commentary," *Machine Translation*, vol. 19, no. 3, pp. 197–211, 2005.

[9] M. Zafar and M. Asif, "Interactive English to Urdu machine translation using example-based approach," *International Journal on Computer Science and Engineering*, vol. 1, no. 3, pp. 275–282, 2009.

[10] P. Brown, J. Cocke, S. Pietra, V. Pietra, F. Jelinek *et al.,* "A statistical approach to language translation," in *Proc. 12th Conf. on Computational Linguistics (Coling '88), Association for Computational Linguistics*, USA, vol. 1, pp. 71–76, 1988.

[11] S. Khan and R. B. Mishra, "Statistical machine translation system for English to Urdu," *International Journal of Advanced Intelligence Paradigms*, vol. 5, no. 3, pp. 182–203, 2013.
[12] U. Singh, V. Goyal and G. Singh, "Urdu to Punjabi machine translation: An incremental training approach," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 4, pp. 227–237, 2016.
[13] B. Jawaid and D. Zeman, "Word-order issues in English-to-Urdu statistical machine translation," *The Prague Bulletin of Mathematical Linguistics*, vol. 95, no. 1, pp. 87–106, 2011.
[14] H. Chefer, S. Gur and L. Wolf, "Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers," in *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Montreal, QC, Canada, pp. 397–406, 2021.
[15] X. Jiang, Z. Xiao, B. Zhang, X. Zhen, X. Cao *et al.,* "Crowd counting and density estimation by Trellis encoder-decoder networks," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 6133–6142, 2019.
[16] R. Vallea, J. M. Buenaposadab and L. Baumelaa, "Cascade of encoder-decoder CNNs with learned coordinates regressor for robust facial landmarks detection," *Pattern Recognition Letters*, vol. 136, no. 2, pp. 326–332, 2020.
[17] J. Salazar, K. Kirchhoff and Z. Huang, "Self-attention networks for connectionist temporal classification in speech recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, pp. 7115–7119, 2019.
[18] R. Masumura, T. Tanaka, T. Moriya, Y. Shinohara, T. Oba *et al.,* "Large context end-to-end automatic speech recognition via extension of hierarchical recurrent encoder-decoder models," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, pp. 5661–5665, 2019.
[19] H. Lee, W. H. Kang, S. J. Cheon, H. Kim and N. S. Kim, "Gated recurrent context: Softmax-free attention for online encoder-decoder speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 710–719, 2021.
[20] A. Selvaraj and E. Nithiyara, "A convolutional encoder-decoder residual neural network for liver tumour segmentation," *Neural Process Letters*, vol. 54, no. 3, pp. 1–20, 2022.
[21] I. Sutskever, O. Vinyals and Q. Le, "Sequence to sequence learning with neural networks," in *Proc. Advances in neural information processing systems*, Montreal, Quebec, Canada, pp. 3104–3112, 2014.
[22] J. Chung, Ç. Gülçehre, K. Cho and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," CoRR abs/1412.3555. arXiv: 1412.3555. 2014. [Online]. Available: http://arxiv.org/abs/1412.3555
[23] K. Cho, B. Merrienboer, D. Bahdanau and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," CoRR abs/1409.1259. arXiv: 1409.1259. 2014. [Online]. Available: http://arxiv.org/abs/1409.1259
[24] D. Bahdanau, K. Cho and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. on Learning Representations-ICLR*, San Diego, CA, USA, 2015.
[25] M. Luong, H. Pham and C. D. Manning, "Effective approaches to attention-based neural machine translation," 2015. [Online]. Available: http://arxiv.org/abs/1508.04025
[26] J. Gehring, M. Auli, D. Grangier, D. Yarats and Y. Dauphin, "Convolutional sequence to sequence learning," in *Proc. 34th Int. Conf. on Machine Learning*, vol. 70, pp. 1243–1252, 2017.
[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30. Long Beach, CA, USA: Curran Associates, Inc., pp. 5998–6008, 2017.
[28] S. A. B. Andrabi and A. Wahid, "Machine translation system using deep learning for English to Urdu," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–11, 2022.
[29] S. Khan and I. Usman, "Model for English to Urdu and Hindi machine translation system using translation rules and artificial neural network," *The International Arab Journal of Information Technology*, vol. 16, no. 1, pp. 125–131, 2019.

[30] A. Khan and A. Sarfaraz, "RNN-LSTM-GRU based language transformation," *Soft Computing*, vol. 23, no. 24, pp. 13007–13024, 2019.
[31] S. A. Rauf, S. Abida, N. e Hira, S. Zahra, D. Parvez *et al.,* "On the exploration of English to Urdu machine translation," in *Proc. 1st Joint Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU) and Collaboration and Computing for Under-Resourced Languages European Language Resources Association*, Marseille, France, pp. 285–293, 2020.
[32] B. Zhang, D. Xiong and J. Su, "A GRU-gated attention model for neural machine translation, 2017. CoRR abs/1704.08430,". ArXiv: 1704.08430. [Online]. Available: http://arxiv.org/abs/1704.08430
[33] S. Maruf, A. F. T. Martins and G. Haffari, "Selective attention for context-aware neural machine translation," in *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics*, Minneapolis, Minnesota, Minneapolis, MN, USA, vol. 1, pp. 3092–3102, 2019.
[34] B. Zhang, D. Xiong, J. Su, Q. Lin and H. Zhang, "Simplifying neural machine translation with addition-subtraction twin-gated recurrent networks," in *Proc. Conf. on Empirical Methods in Natural Language Processing*, Brussels, Belgium, pp. 4273–4283, 2018.
[35] M. Chen, O. Firat, A. Bapna, M. Johnson, W. Macherey *et al.,* "The best of both worlds: Combining recent advances in neural machine translation," in *Proc. 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, Australia, pp. 76–86, 2018.
[36] J. Hao, X. Wang, S. Shi, J. Zhang and Z. Tu, "Multi-granularity self-attention for neural machine translation," in *Proc. Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing*, Hong Kong, China, pp. 887–897, 2019.
[37] T. Zenkel, J. Wuebker and J. DeNero, "Adding interpretable attention to neural translation models improves word alignment," 2019. [Online]. Available: http://arxiv.org/abs/1901.11359
[38] A. Gomez, I. Zhang, K. Swersky, Y. Gal and G. Hinton, "Learning sparse networks using targeted dropout," CoRR abs/1905.13678. arXiv: 1905.13678. URL, 2019. http://arxiv.org/abs/1905.13678
[39] J. Choe and H. Shim, "Attention-based dropout layer for weakly supervised object localization," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 2219–2228, 2019.
[40] L. Zehui, P. Liu, L. Huang, J. Chen, X. Qiu *et al.,* "DropAttention: A regularization method for fully-connected self-attention networks," arXiv preprint arXiv: 1907, 2019.
[41] Z. Tan, S. Wang, Z. Yang, G. Chen, X. Huang *et al.,* "Neural machine translation: A review of methods, resources, and tools," *AI Open*, vol. 1, pp. 5–21, 2020.
[42] G. Klein, Y. Kim, Y. Deng, J. Senellart and A. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proc. 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 2017. URL [Online]. Available: https://doi.org/10.18653/v1/P17-4012
[43] J. Tiedemann, "Parallel data, tools and interfaces in OPUS," in *Proc. 8th Int. Conf. on Language Resources and Evaluation (LREC'12), European Language Resources Association (ELRA)*, Istanbul, Turkey, vol. 2012, pp. 2214–2218, 2012.
[44] S. Moustafa, Kaggle 2020. [Online]. Available: https://www.kaggle.com/samirmoustafa/arabic-to-english-translation-sentences
[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computing*, vol. 9, no. 8, pp. 1735–1780, 1997.
[46] J. Oliva, B. Póczos and J. Schneider, "The statistical recurrent unit," in *Proc. 34th Int. Conf. on Machine Learning*, Sydney, NSW, Australia, vol. 70, pp. 2671–2680, 2017.
[47] J. Yang, B. Zhang, Y. Qin, X. Zhang, Q. Lin *et al.,* "Otem&Utem: Over- and under-translation evaluation metric for NMT," in *Proc. Int. Conf. Natural Language Processing and Chinese Computing*, Hohhot, China, vol. 11108, pp. 291–302, 2018.