International Journal of Mathematical, Engineering and Management Sciences
Vol. 9, No. 5, 1067-1088, 2024
https://doi.org/10.33889/IJMEMS.2024.9.5.056

Ram Arti
Publishers

Improved Urdu-English Neural Machine Translation with a fully Convolutional Neural Network Encoder

Huma Israr
School of Electrical Engineering and Computer Science (SEECS),
National University of Sciences and Technology (NUST), Islamabad, Pakistan.
Corresponding author: hisrar.dphd17seecs@seecs.edu.pk

Muhammad Khuram Shahzad
School of Electrical Engineering and Computer Science (SEECS),
National University of Sciences and Technology (NUST), Islamabad, Pakistan.
E-mail: mkhuram.shahzad@seecs.edu.pk

Shahid Anwar
Department of Information Engineering Technology (IET),
National Skills University, Islamabad, Pakistan.
E-mail: shahidanwar@nsu.edu.pk

(Received on March 5, 2024; Revised on May 7, 2024 & June 26, 2024; Accepted on July 25, 2024)

Abstract
Neural machine translation (NMT) approaches driven by artificial intelligence (AI) has gained more and more attention in recent years, mainly due to their simplicity yet state-of-the-art performance. Despite NMT models with attention mechanism relying heavily on the accessibility of substantial parallel corpora, they have demonstrated efficacy even for languages with limited linguistic resources. The convolutional neural network (CNN) is frequently employed in tasks involving visual and speech recognition. Implementing CNN for MT is still challenging compared to the predominant approaches. Recent research has shown that the CNN-based NMT model cannot capture long-term dependencies present in the source sentence. The CNN-based model can only capture the word dependencies within the width of its filters. This unnatural character often causes a worse performance for CNN-based NMT than the RNN-based NMT models. This study introduces a simple method to improve neural translation of a low-resource language, specifically Urdu-English (UR-EN). In this paper, we use a Fully Convolutional Neural Network (FConv-NN) based NMT architecture to create a powerful MT encoder for UR-EN translation that can capture the long dependency of words in a sentence. Although the model is quite simple, it yields strong empirical results. Experimental results show that the FConv-NN model consistently outperforms the traditional CNN-based model with filters. On the Urdu-English Dataset, the FConv-NN model produces translation with a gain of 18.42 BLEU points. Moreover, the quantitative and comparative analysis shows that in a low-resource setting, FConv-NN-based NMT outperforms conventional CNN-based NMT models.

Keywords- Neural machine translation (NMT), Urdu, English, BLEURT-20 score, Perplexity, Convolutional neural network (CNN).

1. Introduction

Neural machine translation (NMT) models significantly advanced the field of natural language processing (NLP) by constructing a single neural network (NN) that enables end-to-end learning (Cho et al., 2014b; Sutskever et al., 2014). A typical NMT model consists of two basic parts. First, an encoder network that takes the input sentence as embeddings and produces a dense representation in the form of a vector. Second, a decoder network that uses the encoder's output and employing an attention mechanism to generate the translation (Bahdanau et al., 2014). Most NMT models that exist today still use the classical encoder-decoder framework. To enhance NMT systems further, different NMT model have been built based on classical encoder-decoder framework. Among those, the model proposed by Gehring et al.

(2017) and Vaswani et al. (2017) have significantly enhanced MT capabilities by proposing innovative NMT model. A key feature of NMT is its enhanced ability to handle unseen text. This is due to advanced word embedding techniques that capture word similarities and help with generalization (Veliz et al., 2021). Furthermore, NMT models incorporate attention mechanisms to effectively manage changes in word order over larger distances within sentences.

Recurrent Neural Networks (RNNs)-based NMT model have long anchored state-of-the-art in language translation. Despite their prominence, RNN-based NMT models face many challenges. These include vanishing gradients, forgetting information over long distances, and difficulty in retaining long-term dependencies (Cho et al., 2014b). To address these issues, more complex architectures like LSTMs and GRUs were proposed by Bahdanau et al. (2014) and Luong et al. (2015). However, RNNs have maintained their superiority on NL applications. This supremacy was due to their ability to model sequential data with strong predictive power. However, recent advances in deep learning are changing this historical dominance of RNNs in NLP applications (Johri et al., 2021).

In parallel, Convolutional Neural Networks (CNNs) have proven remarkably effective across diverse machine learning applications like image processing and computer vision (LeCun et al., 2015). Additionally, CNNs have showcased their potential by creatively using their celebrated convolutional operations for text data processing. As highlighted by Gehring et al. (2017), CNN-based NMT models require less memory to train. This improves computational efficiency and makes them more suitable for deployment in large-scale MT systems. Furthermore, CNNs can effectively extract local features from input sequences and capture local dependencies through the use of convolutional filters. They also utilize dilated convolutions to capture dependencies over wider contexts within the text (Gehring et al., 2017).

In the early exploration of Convolutional Neural Networks (CNN) for MT, a CNN-based encoder was proposed by Kalchbrenner et al. (2014) for encoding input representations. They used an RNN decoder to generate translations. This seminal work thus inspired further research into hybrid CNN-RNN models for language applications. In subsequent research efforts to explore CNN-RNN encoder-decoder approaches, Cho et al. (2014a) proposed a hybrid CNN-RNN model. They used a gated convolutional network as the encoder and an RNN as a decoder. However, their proposed CNN-RNN model performed worse when compared to RNN-based models. Another significant advancement in the development of convolutional architectures for NMT was made by Kaiser et al. (2017). They introduced a fully CNN-based NMT system with Extended Neural GPU framework. Subsequently, Gehring et al. (2017) achieved one of the early breakthroughs in convolutional NMT. Their work demonstrated that a purely CNN-based encoder could attain a translation quality similar to RNN-based models of that era. Concurrently, Kalchbrenner et al. (2016) independently put forth a ByteNet-style convolutional architecture for NMT. The proposed approach established a new benchmark for character-level translation. However, it faced challenges with word-level translation.

CNN-based NMT models clearly have an advantage over their RNN-based counterpart, especially in terms of training speeds. The fundamental structure of CNNs facilitates parallel calculations. The calculations are often performed by multiple filters (Yang et al., 2020). Using multiple filters accelerates the training process as they receive and transform input representations. Additionally, due to their inherent parallel processing nature, The CNN-based NMT model avoids the issue of vanishing gradients. The vanishing gradient problem used to affect RNN-based NMT models. However, earlier CNN-based translation models suffered from critical limitations that effected final quality of translation produced. The original CNN-based model uses convolutional filter. It can only consider relationships within the fixed width of its filters. It does not capture dependencies between words that are farther apart in the input sequence. Due to this inherent limitation in the CNN-based model often resulted in lower performance when compared to RNN-based models (Pandey and Roy, 2023).

To address the earlier limitations, researchers have developed more advanced convolutional NMT architectures. For example, Kaiser and Bengio (2016) introduced a depth-wise separable convolution approach for NMT. The SliceNet model developed using this technique achieved quality comparable to their earlier work. In a separate study, Gehring et al. (2017) proposed adding an attention mechanism into a CNN-based translation system. This enhanced approach achieved even better results than RNN-based models.

The CNN-based NMT model is different from traditional RNN-based NMT (with LSTM or GRU cells) and transformers in various ways. RNNs use sequential processing of input data, which makes them slower. In contrast, CNN-based model processes input data in parallel through the use of convolutional layers. This approach speeds up training and inference times (LeCun et al., 1998). Additionally, unlike transformers, the CNN-based model does not explicitly encode positional information present in the input data. Instead, the model implicitly captures the positional information through spatial relationships. Moreover, CNN-based NMT models are often more memory-efficient compared to RNNs and transformers, especially for translation tasks involving long input sequences (Kalchbrenner et al., 2016). This efficiency stems from the parallel nature of convolutional operations and the ability to process input data in a fixed-size window.

However, both RNN-based and CNN-based NMT systems achieved success only after being trained on massive datasets containing millions of parallel sentence pairs. Such extensive data resources exist only for a few well-resourced language pairs. These pairs commonly comprise English paired with major European languages, such as French, German, Arabic, or Chinese (Krasadakis et al., 2024). In reality, the amount of parallel data available is vastly insufficient or completely unavailable for most of the world's low-resource languages. This includes language pairs such as Urdu-English. The absence of large-scale training resources for the vast majority of language combinations remains a major challenge (Haddow et al., 2022).

In light of the challenges presented by the availability of large parallel corpora, researchers have shown a growing interest in expanding NMT. They aim to extend the application of NMT model to more diverse language typologies beyond English-centered pairs. The literature survey indicates considerable efforts in exploring CNN-based NMT models for translation. While past studies have applied convolutional architectures to various languages and domains, there is still room for real-world applicability of NMT to under-resourced languages like Urdu.

In this study, we explore applying a fully convolutional neural machine translation (NMT) model to the low-resource language pair Urdu-English and evaluate the quality of translations generated by the model. The novel contributions presented in this paper are:

- We develop an Urdu-English machine translation model using a fully convolutional neural network (FConv-NN)-based encoder-decoder architecture with an attention mechanism in a low-resource setting.
- We evaluate the machine translation model through multiple quantitative metrics.
- We performed a comparative evaluation of translations produced by a standard CNN-based NMT model versus our proposed FConv-NN-based NMT model.

Following the introduction, rest of the paper is structured as follows: Section 2 contains the literature review that this paper contributes, Section 3 introduces the necessary concepts and details about CNN and fully convolutional neural network (FConv-NN) for the translation task, Section 4 reviews the result generated by proposed FConv-NN and, Lastly, Section 5 presents the conclusions and discusses some challenges associated with the FConv-NN research, and outlines potential areas for future work.

Table 1. Key findings from literature review.

S. No.	Reference	Model used	Languages	Task	Research findings
1.	Sánchez-Cartagena et al. (2024)	RNN, Transformer	Czech, German, Spanish, Turkish- English	Translation	Improved translation quality
2.	Israr et al. (2023)	CNN, LSTM, GRU, Transformer	Urdu-English, Arabic-English, Persian-English	Translation	Improved Accuracy, BLEU, Over-Translation, Recall, F1
3.	Meng et al. (2015)	CNN, inCNN, TagCNN	NIST Chinese-English	Translation	Improved BLEU score
4.	Kalchbrenner et al. (2014)	CNN	English-to-German WMT	character-to-character machine translation	Robust Translation Time, Improved BLEU
5.	Costa-Jussa and Fonollosa (2016)	CNN	German-English WMT	Character-based NMT	Improved BLEU score
6.	Karpukhin et al. (2019)	CNN, Transformer	IWSLT 2016 De-En, Fr-en, Cs-En	Character-based NMT	Improved BLEU score
7.	Tang et al. (2018)	CNN, RNN	newstest sets & ContraWSD DE-FR DE-EN	Translation	Reduced perplexity, Improved Accuracy
8.	Wang et al. (2019)	CNNW, CCNC	Mongolian-Chinese CWMT'2009	Translation	Improved BLEU score
9.	Vylomova et al. (2016)	CNN	Europal Est-En, Ro-En, Ru-En	Translation	Improved BLEU score, word position, and METEOR score.
10.	Beddiar et al. (2021)	CNN, LSTM	AskFm, Formspring, Olid, Warner & Waseem, Wikipedia Toxic comment	back Translation	Improved F1 score, accuracy, Precision and Recall
11.	Tiwari et al. (2020)	LSTM, CNN	HindEnCorp and (CFILT), IIT Bombay Hindi-English	Translation	Improved BLEU score,
12.	Naeem et al. (2023)	RNN, LSTM, GRU	English-Urdu, English-German	Translation	Improved Accuracy
13.	Tan et al. (2022)	LSTM, CNN	Malay-English, French-English	Translation	Improved Accuracy
14.	Hu (2023)	LSTM, CNN	English-Chinese	Translation	Fluency, Continuous coverage
15.	Dowlagar and Mamidi (2021)	Seq2seq based CNN	English to Hinglish	Code Switch Machine Translation	Improved BLEU score and translation quality
16.	Hujon et al. (2023)	Transformer	English-Khasi	Code-Switch Machine Translation	CNN Code-switching Machine Translation

2. Related Work

This section provides an overview of the relevant studies conducted on applying CNNs to machine translation task and recent developments in CNN-based models for MT. **Table 1** summarizes the key aspects of different techniques and methods from the literature.

The Transformer, RNN-based system and ConvS2S system are established advanced NMT frameworks. They are typical representatives of NMT systems (Mohamed et al., 2024). Training NMT systems for under-resourced language pairs, for which the number of parallel corpora is smaller than those available for prevailing language pairs, may be challenging (Sánchez-Cartagena et al., 2024). Recently, Urdu NLP researchers started training Urdu variants of these models (Naeem et al., 2023). Israr et al. (2023) conducted a thorough investigation into the effectiveness of RNN-based, CNN-based, and Transformer models for Urdu-English machine translation. They trained six different NMT models alongside a conventional CNN model. The highest reported BLEU score was 47 in experiments translating Urdu to English. Detailed comparisons show that the CNN model produced significantly lower BLEU scores and produced translations of low quality compared to other approaches. Quantitative analysis revealed that the CNN translations tended to be repetitive. CNN translations frequently contain repeated words, indicating a lack of diversity in the generated text. Based on these results, the study concluded that among the models evaluated, the CNN approach performed the worst for the task of Urdu-English translation.

Classical CNN for MT tasks typically employs a simple strategy of convolution followed by pooling. Applying pooling directly after convolution overlooks the complex relationships between different parts of a sentence. This approach is less ideal for modeling sentences. To address this issue, Meng et al. (2015) proposed two variants of convolutional models for machine translation tasks: tagCNN and inCNN. The tagCNN takes a source sentence as input and produces a summarized representation called a tag. The inCNN model fuses this summarized representation with the context of the entire source sentence to form a unified representation. The models were tested on Chinese-English translation tasks. The results showed an increase in performance by 1.08 BLEU points compared to the translation generated by the baseline model.

A common challenge for a typical CNN-based NMT model is effectively capturing and conveying the meaning expressed by a sequence of words in a sentence. To address this issue, Kalchbrenner et al. (2014) introduced a character-level CNN-based model called "ByteNet". ByteNet addresses this challenge by incorporating the dilation in the convolution layers. The dilation overcomes the need for excessive memorization seen in other character-level language models. This approach also allows ByteNet to process long sequences efficiently in linear time. In experiments, the proposed ByteNet model outperformed state-of-the-art RNN-based models, proving the effectiveness of the dilation technique.

CNN-based NMT models work well for morphologically rich languages. However, Costa-Jussa and Fonollosa (2016) showed that the CNN-based NMT model is efficient for processing languages that lack complex morphology. The author proposed a modified version of classical CNN-based encoder/decoder architecture. Their model used character-based embeddings as input to convolutional layers. The convolutional layers produced output in the form of a vector. The vector is then forwarded to two highway layers, which maintains flow of information across the network. When tested on an English-German dataset the modified model showed a notable improvement of up to +0.3 BLEU points.

In real-world scenarios the noise and data imperfections are common. To improve NMT model performance in the presence of noise, Karpukhin et al. (2019) proposed a transformer-based MT model. The proposed model utilizes a CNN-based character encoder. The model is trained on IWLST16 data with augmented noise. The experimental results showed that training the model on data with a moderate amount of random synthetic noise can improve the model's robustness.

Theoretically, CNN-based NMT struggle with capturing long-range dependencies. Their filters only cover a limited local context. The performance of CNN decline when the distance between the subject and verb exceeds size of the filters. To empirically verify this phenomenon, Tang et al. (2018) compared CNNs and self-attentional networks to RNNs and their ability to connect distant words. The CNNs and self-attention networks are better at modeling long-range relationships since they have shorter network paths. Tang et al. evaluated experimentally CNNs and self-attentional networks on two tasks: subject-verb agreement, which requires capturing long-distance relations. Secondly, word sense disambiguation. Training and evaluation of the model are conducted using the DE-FR dataset (Tang et al., 2018).

Wang et al. (2019) introduced a CNN-based NMT model for Mongolian-Chinese translation. Mongolian has a rich morphological structure. To address this challenge the authors used byte-pair encoding (BPE) to segment Mongolian words into subword units. In experiments on Mongolian-Chinese translation, the best-reported result was 35.37 BLEU points. A direct comparison showed that representing Mongolian as subword units using BPE performed better than full word unit. Their findings set a new milestone in Mongolian-Chinese translation research.

Vylomova et al. (2016) address the challenge of translating morphologically rich languages by developing models that learn character-level word representations. They introduce CNN alongside another RNN-based model for obtaining word embeddings from character/morpheme decompositions. These representations are integrated into a novel MT Model. Evaluation of target translation produced while translating several morphologically complex languages to English yields consistent gains over baseline and improved the BLEU scores by 1 to 1.5 points.

In an effort to identify hate speech and cyberbullying within online content, Beddiar et al. (2021) put forward using back translation and paraphrasing methods to augment training data. Convolutional neural networks and LSTM models were trained on the original and expanded datasets. They were then evaluated on five public cyberbullying and hate speech corpora. The results indicated that augmenting the datasets using back translation and paraphrasing before training CNN model training resulted in enhanced performance.

Endeavoring to use and evaluate the NMT model for low-resource Asian languages, Tiwari et al. (2020) trained and compared two sequence-to-sequence NMT models - LSTM Seq2Seq and ConvS2S for the English-to-Hindi language pair. In another study, Naeem et al. (2023) evaluated a number of deep neural NMT models for low-resource languages. It tested RNN, LSTM, and GRU models for English-to-Urdu translation. They examined how these models perform while translating a language pair with limited data. Ahmed et al. (2024) later introduced a hybrid model that combines Transformer encoder with CNN. Their research is focused on NER for Urdu language. To improve the performance of the model, they used a deep learning approach enhanced with BERT embeddings. In a subsequent effort, Tan et al. (2022) implemented CNN for Malay-English and French-English translation task. They compared the performance of their proposed CNN model to LSTM with an attention mechanism. The goal of this comparison was to analyze the performance of the CNN model against the LSTM model with respect to the two language pairs. In addition, Hu (2023) explored Chinese-English translation using LSTM and CNN-based models. Their research provided a thorough examination of how effectively these models managed the complexities of English and Chinese. This study showed the relative strengths and weaknesses of LSTM and CNN-based model for English-Chinese translation.

Dowlagar and Mamidi (2021) proposed a gated-CNN model for English-to-Hinglish code-switched MT. The proposed model effectively handled the complexities of code-switching. The research found that gated-CNN models work better for English-Hinglish code-switched MT than recurrent-based NMT models. Similarly, Hujon et al. (2023) developed a Seq2Seq CNN-based model for English to Khasi. Their study showed that the CNN are more effective than RNN-based model for code-switching MT tasks. The convolutional layers in CNN-based effectively captured the contextual representations compared to RNN-based NMT models. Laskar et al. (2022) experimented with a CNN-based NMT model for translating Assamese, a low-resource language, to English. To address the problem of limited data, they used synthetic parallel data. The process automatically generates parallel sentences and adds them into train set. This approach resulted in improved translation accuracy for both English-to-Assamese and Assamese-to-English translations.

The literature review shows substantial research applying Convolutional Neural Networks (CNNs) to MT. Besides, much work has been done on translating resource-rich languages like English, French, and Chinese translation. However, NMT for morphologically rich, under-resource Urdu-English pair is still in its early stages. A few studies have explored neural approaches, techniques, and models for Urdu-English translation. There remain opportunities to enhance the performance of NMT systems further, particularly for morphologically rich, low-resource language Urdu translation. This research aims to address the gaps identified in the literature by adopting neural techniques for Urdu-English translation. Specifically, it proposes a CNN-based neural machine translation model to handle this language pair.

3. Methodology

In this section, the details of the tools, the CNN-based NMT model, and the implemented FConv-based NMT model, as shown in **Figure 1** and **Figure 2** are discussed. **Table 2** lists the variables used to describe the methodology and their description.

Table 2. List of the variables used, each with its description.

S. No.	Variable	Description	S. No.	Variable	Description
1.	*x*	source sentence	7.	*e*	embedding vector
2.	*y*	target translation	8.	*a*	attention score
3.	*p*	position embedding	9.	*c*	context vector
4.	*h*	hidden state/ decoder output	10.	*b*	bias
5.	*z*	encoder output	11.	*d*	decoder state summary
6.	*k*	kernel width	12.	*g*	previous target element

3.1 CNN Model

A CNN-based encoder-decoder architecture for MT takes source sentence x= (x_1, x_2, ..., x_m), represented as word embedding and positional embedding. This model generates a target sentence y = (y_1, y_2, ..., y_n) in the target language, as illustrated in **Figure 1** and **Figure 2**. The encoder produces an encoded representation of the source sentence, and the decoder generates the target sentence by conditioning it on the encoding of the source sentence. This is accomplished by calculating the probability of the next target word based on the previously generated target words and the source sentence x, as described in Equation (1):

$$P(Y_{i+1}|y_1,\dots y_i\ ,x) \tag{1}$$

***Word embedding*:** Multi-layer CNN for NMT models takes as input a source language sequence x and is projected into distributed space, represented by w = (w_1, w_2 ...,w_m). Here, each word w_j is a column vector in matrix D ε R_f. f is the embedding dimension, with V being the size of the vocabulary. To maintain word order information, the absolute positions of words in x are mapped to a dense vector p. The generated position embedding is p = (p_1, p_2, ..., p_m), where p_j ε R_f. Word and position embeddings are then combined to form the overall word representation e_j, which is an element-wise vector.

$$e_j^{(d)} = w_j^{(d)} + p_j^{(d)} \tag{2}$$

***CNN Encoder*:** When constructing a convolutional seq2seq model, the initial element is the Encoder. The encoder consists of simple blocks of convolution operation. The convolutional seq2seq model designed for NMT shares this block structure between its encoder and decoder components.

$$W^{(2dxkd)}.\ e^{(kd)}_{j..j+k-1} \tag{3}$$

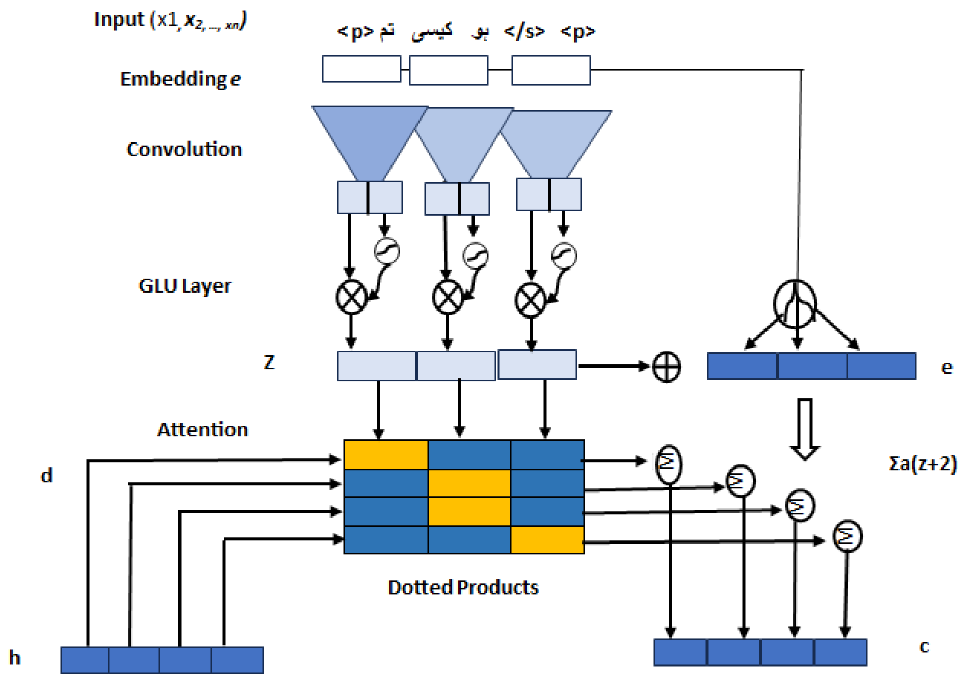


Figure 1. Seq2Seq convolution neural network (CNN) encoder.

where, *k* is the width of the kernel and *d* is the dimension of the embeddings. To enhance the learning and representation capabilities, a nonlinearity called the Gated Linear Unit (GLU) is applied to the output of the convolution layer. The GLU implements a straightforward gating mechanism on the convolution output, denoted as Y = [A B].

$$v[AB] = A \otimes \sigma(B) \tag{4}$$

The gates in the Gated Linear Unit (GLU) regulate the contextual importance of the inputs A. Following that, a Residual Connection is incorporated from the input of each convolution to the output of the block, resulting in the creation of a combined vector. For the l^{th} bock, the combined vector is denoted as. $z_j^l = (z_1^l, z_2^l, ..., z_m^l)$ in the encoder and $h_i^l = (h_1^l, h_2^l, ..., h_n^l)$.

The combination of a convolutional layer and a GLU layer is commonly referred to as a block. In a CNN-based NMT implementation, different numbers of these blocks can be utilized. The number of blocks employed can vary depending on the particular implementation or architecture.

Decoder: The decoder in CNN-based NMT exhibits a solid resemblance to the encoder. Like encoder, the decoder receives the output sequence y and sends the tokens through the embedding layer to obtain word and positional embeddings. The positional vectors in the decoder keep track of a section of the input sequence that the model is currently dealing with. The word and positional embeddings are combined using element-wise summation and then fed into a fully convolutional layer, which is subsequently connected to the next convolutional layer.

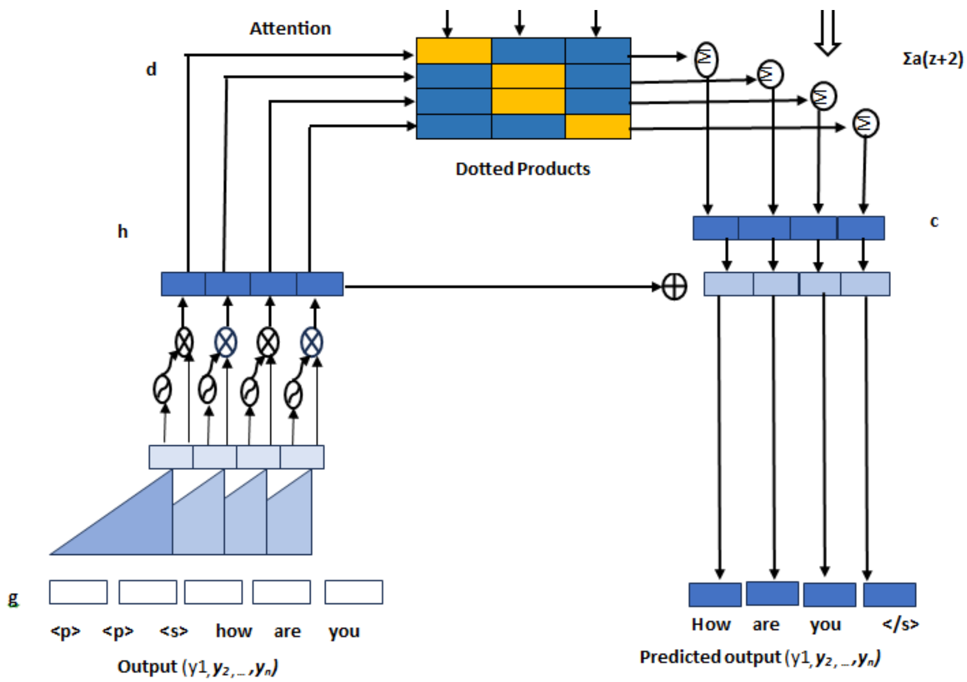


Figure 2. Seq2Seq convolution neural network (CNN) decoder.

In the decoder, the convolutional layer is supplemented with two extra inputs from the encoder. These inputs serve to incorporate the encoder's information into the decoder. Additionally, the embedding vector is passed as a residual connection to the convolutional layer.

Lastly, the model calculates a probability distribution over the T potential next target elements y_{i+1}. This is achieved by transforming the top decoder output h_i^l through a linear layer with weights and bias b.

$$h_i^l = v(W_l[\ h_{i-k/2}^{l-1},\ .\ .\ h_{i+k/2}^{l-1}\ ]\ b_w^l) + h_i^{l-1} \tag{5}$$

The top decoder output is transformed with a linear layer with weights W_o and bias b_o, respectively.

$$P(Y_{i+1}|y_1,\ldots y_i, x) = softmax(W_o\ h_i^l + b_o) \in R^T \tag{6}$$

***Attention*:** The CNN-based NMT model computes separate attention at each decoding step by combining the current hidden state with an embedding of the previous target element to get the current decoder state summary.

$$d_i^l = w_d^l h_i^l + b_d^l + g_i \tag{7}$$

While g_i is the previous target element.

The attention weights for the current decoder layer are calculated by taking the dot product between the summarized decoder state and each encoded representation from the final encoder block.

$$a_{ij}^l = \frac{exp(d_i^l, z_j^u)}{\sum_{t=1}^{m} exp(exp(d_i^l, z_t^u)} \tag{8}$$

The conditional input for the current decoder layer is calculated as a weighted sum of the encoded representations and input token embeddings. This weighted sum is then added to the output of the corresponding decoder layer.

$$c_i^l = \sum_{j=1}^{m} a_{ij}^l \ (z_j^u + e^j) \tag{9}$$

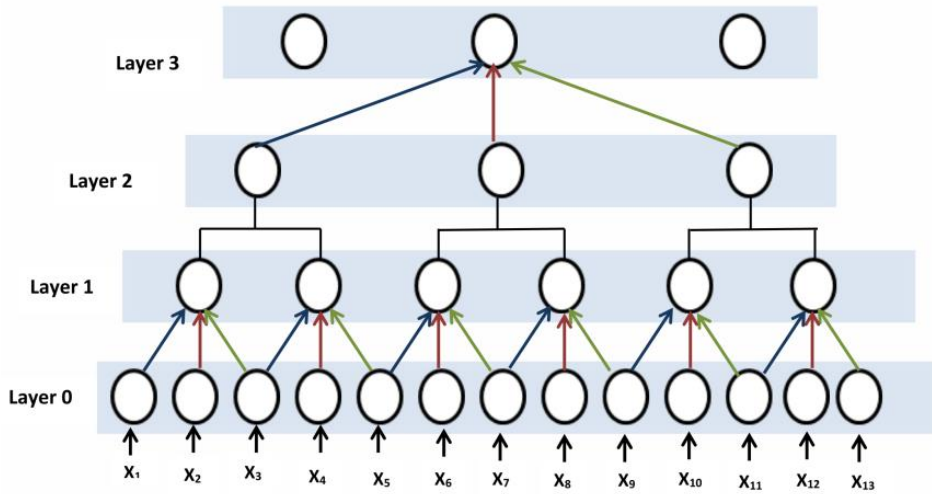
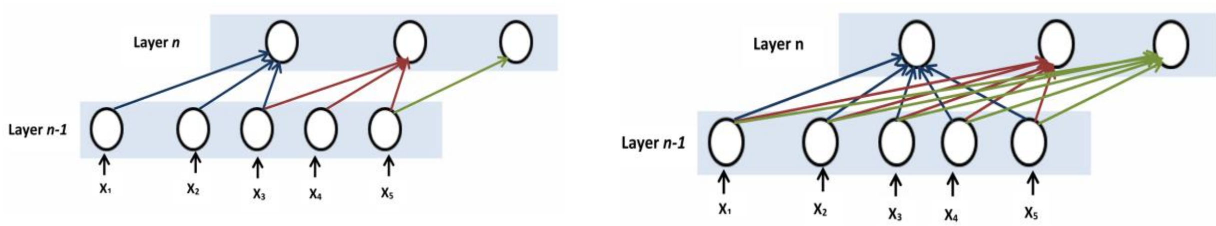


Figure 3. Generic architecture of CNN for NMT.

3.2 Generic CNN & FConv-NN Model

The fundamental structure of a standard CNN encoder, depicted in **Figure 3**, consists of a sequence of layers. In general, the output of one convolutional block is usually passed through an additional linear layer before being fed into the next convolutional layer.



Convolution layer with window size three

Fully connected final layer

Figure 4. Fully convolution layer vs. Fully connected convolutional layer.

Layer-0**:** The input layer accepts word embeddings as input. In this work, the maximum input sequence length is set to 512 words. Any shorter sequences are padded with zeros at the beginning to reach the fixed length.

Layer-1-CNN**:** The CNN consists of alternating convolutional and max pooling layers to generate feature maps. The convolutional layer extracts local features by performing a convolution operation within sliding windows of size three that are applied across the input sequence, as shown in **Figure 4**. As the window slides over each word and its neighbors, contextual features are extracted from localized regions of the sentence. In the context of CNN-based NMT, the vector from the previous layer is passed through a fully connected layer to transform it into a specific dimension. The hidden layers flatten the features. Fully connected CNN loses spatial information as it passes through dense layers.

Fully connected CNNs are generally harder to train due to the high number of parameters in dense layers. The network performs extensive computation at each layer. The final layer transforms the input into a vector before making the predictions, thereby increasing computational complexity and memory requirement.

***Layer-1-FConv*:** In a fully convolutional CNN, the network consists entirely of convolutional layers. The final layers of an FConv-NN usually consist of convolutional layers with a global pooling operation, such as global average pooling. In contrast to CNN, that consists of alternating convolutional and max pooling layers, the convolutional layer in FConv-NN extracts local features by performing a convolution operation within sliding windows. These sliding windows are applied across the input sequence, as shown in **Figure 4**. As the window slides over each word and its neighbors, contextual features are extracted from localized regions of the sentence. To handle inputs/outputs of varying lengths, the convolutions in FConv-NN are strided/padded. The strided/padded convolutions reduce/expand spatial dimensions as needed. The convolutional operation is applied throughout the network at different layers. This property helps FConv-NN preserve spatial information and capture local dependencies effectively. The final convolutional layer generates encoded sequences. This encoded sequence is then used by the decoder to generate the output sequences.

One potential benefit of using FConv-NN is that it is translation invariant i.e. features at different locations of the input have the same meaning. This helps encode relative positional relationships in sequences better than CNNs. Additionally, the FConv-NN is more efficient in terms of parameters. A fully convolutional CNN has fewer parameters as the network consists entirely of convolutional layers.

3.3 Tools

The training and evaluation of the proposed model is conducted using FAIRSEQ, an open-source sequence modeling toolkit built on PyTorch. Specifically, the researchers utilized Fairseq-py's PyTorch implementation of FConv models. Training and testing are performed on a system containing an Intel®CoreTM i9-9900K CPU @3.60GHZ along with an NVIDIA GeForce GTX 1650 SUPER GPU.

3.4 Model Configuration

One of the critical aspects of training any NMT model is selecting the optimal hyper-parameters and model parameters. Model parameters refer to the weights and biases that can be initialized and updated through data learning. In contrast, hyper-parameters cannot be directly estimated from the learning process and must be defined before training a model (Bunrit et al., 2023). According to research by Bengio (2012) and Goodfellow et al. (2016) two key hyper-parameters impacting the performance of fine-tuning CNN models are the initial learning rate and the number of training epochs. To maximize the translation quality of the convolutional model, these hyper-parameters are carefully tuned. Specifically, the learning rate is set to 0.25 with a dropout of 0.2 to help optimize the training process. The model is trained for 15 epochs.

Table 3 contains the model parameters and hyper-parameter for the FConv-NN model used in the experiment. The model parameters and hyper-parameters for the CNN and CNN-ADL models used in the experiment are drawn from the research conducted by Israr et al. (2023).

Table 3. Model parameter / Hyper-parameter for FConv-NN Model used in the experiment.

S. No.	Parameter / Hyper-parameter	Configuration	S. No.	Parameter / Hyper-parameter	Configuration
1.	Number of hidden units	512	10.	beam	5
2.	Word embedding	512	11.	Minimum loss scale	0.0001
3.	Dropout ratio	0.2	12.	Weight decay	0.0
4.	Train Epoch	15	13.	Momentum	0.99
5.	Learning rate	0.25	14.	Clip norm	0.1
6.	Force Anneal	50	15.	Label smoothing	0.1
7.	Number of Encoder / Decoder Layers	15	16.	Nbest	1
8.	Optimizer	Nag	17.	Localsgd frequency	3
9.	Criterion	Label smoothed cross entropy	18.	Slowmo base algorithm	localsgd

Table 4. Sample sentences from Urdu-English training corpus.

Source Sentence in Urdu	Target Translation in English
مدد .	Help.
لینگویج پروسیسنگ آرٹفشل انٹیلی جنس کی ایک شاخ ہے۔	Language processing is a branch of artificial intelligence.
کرسٹیانو رونالڈو کا شمار دنیا کے مہنگے ترین اور سب سے زیادہ کمانے والے کھلاڑیوں میں ہوتا ہے	Cristiano Ronaldo is one of the most expensive and highest earning players in the world.
مسٹر صدر ، خواتین و حضرات ، آرٹیکل 170 کے مطابق ، میں گروپ آف یورپی‌پیپلز پارٹی اور یورپی ڈیموکریٹس کی جانب سے درخواست کرتا ہوں کہ قرارداد اور رپورٹ کے بارے میں حتمی ووٹ اگلے مکمل اجلاس تک ملتوی کردیا جائے	Mr. President, ladies and gentlemen, in accordance with Article 170, I request on behalf of the Group of the European People's Party and European Democrats that the final vote on the resolution and the report be postponed until the next plenary sitting.

3.5 Dataset

An Urdu-English parallel corpus of 10,000 sentences, as collected by Israr et al. (2023), is utilized. The corpus contains sentences from news, sports, and movies dialogues that are commonly used in everyday conversations. It comprised a 90,000 unique sentence in the training set, a 5,000 sentence in the validation set, and a 5,000 sentence in the test set. The source (Urdu) and target (English) sentences are stored in separate files, with each line corresponding to the translation in the other file, as shown in **Table 4**. Individual sentences in both files are separated by new lines.

3.6 Evaluation Methods

To measure the quality of translation, we select several automatic evaluation metrics. These include BLEU (Papineni et al., 2002), perplexity (PPL) (Jelinek et al., 1977), Accuracy, METEOR (Banerjee and Lavie, 2005), TER (Subramanian et al., 2007) and unigram F1. BLEU value is selected because it is used to evaluate the translation quality of the output produced by the models. It also measures the similarity between output translations and gold reference using n-gram overlap. On the other hand, perplexity is used to measure models' ability to generate coherent, contextually appropriate translations. It will evaluate the models' confusion when generating text. The remaining metrics are used to determine the accuracy of the model. Lastly, we will analyze the models' performance through qualitative analysis of the translated texts.

4. Result & Discussion

This section compares the performance of the proposed FConv-NN model to two other models, CNN and CNN-ADL. The CNN-ADL model (Israr et al., 2023) adds an additional attention-based dropout layer in CNN for MT. This additional layer helps the model to focus on parts of the input data that are useful and to avoid over-fitting.

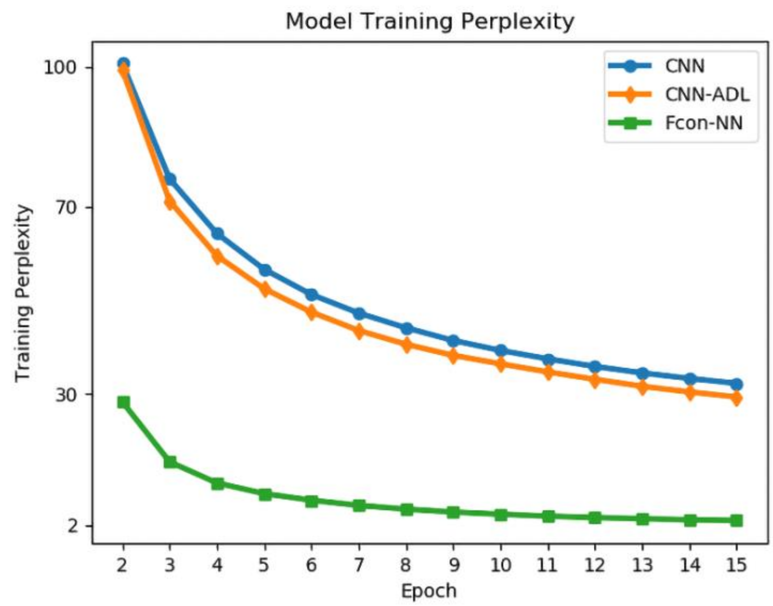
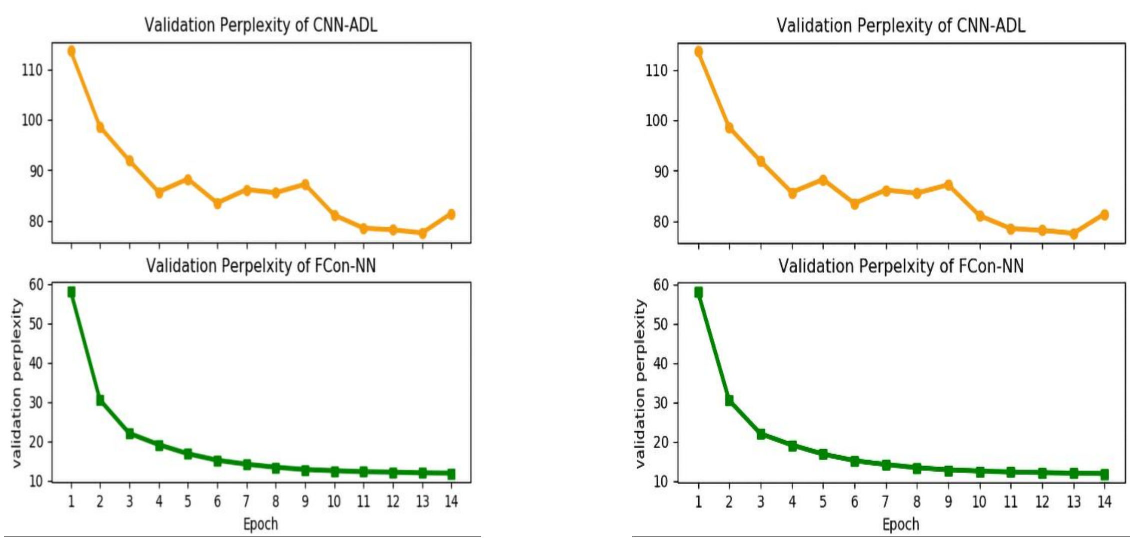


Figure 5. Training perplexity of CNN, CNN-ADL, and FConv-NN model.

In order to establish how well each model generalizes on training data, we analyzed learning behavior of all models. This analysis involves tracking step-wise perplexity during training and validation. It allows us to observe the model performing over time.



Validation Perplexity of CNN, FConv-NN model | Validation Perplexity of CNN-ADL, FConv-NN model

Figure 6. Validation perplexity of CNN, CNN-ADL, and FConv-NN.

Figure 5 illustrates the training perplexity at each step during training. Additionally, **Figure 6** depicts the validation perplexity achieved during validation. From **Figure 6**, it is also clear that the validation perplexity of CNN and CNN-ADL fluctuates in a certain range as compared to FConv-NN, which follows a smooth curve. In contrast to the FConv-NN model's curve, which achieved a stable state, the validation perplexity for the CNN and CNN-ADL models began rising as the number of training epochs increased.

Table 5. Model’s training and validation perplexity at final epoch.

Statistics	CNN	CNN-ADL	FCONV-NN
Training Perplexity	29.37	32.29	2.98 ↓
Validation Perplexity	81.44	83.53	11.68 ↓

As shown in **Table 5**, on the final training epoch, the FConv-NN model achieved a training perplexity of 2.98 and a validation perplexity of 11.87. In contrast, when evaluated on the same dataset, the CNN and CNN-ADL models achieved considerably higher validation perplexities of 32.98 and 29.37, respectively.

Table 6. Model’s statistics on test set at final epoch.

Result on Test set	FCONV-NN	CNN	CNN-ADL
Translation time in Min	10 ↓	205	-
BLEU Score	40.22↑	21.80	21.54
GLUE-Corpus	0.420 ↑	0.196	0.192
GLUE-sentence Average Score	0.445 ↑	0.287	0.279
ROUGE_L Score	0.669 ↑	0.474	0.466
METEOR	0.377 ↑	0.209	0.204
TER	0.411 ↓	0.826	0.859
WER	11.88 ↓	21.00	21.00
Precision:	0.742 ↑	0.512	0.510
Recall:	0.715 ↑	0.395	0.387
f1:	0.728 ↑	0.446	0.440
fMean:	0.719 ↑	0.409	0.402
Bleu_1:	69.15 ↑	37.40	36.70
Bleu_2:	56.61 ↑	30.10	29.40
Bleu_3:	47.52 ↑	25.40	24.67
Bleu_4:	40.32 ↑	21.80	21.54

In the context of machine translation, a lower perplexity score implies better predictive performance and an enhanced ability of the model to generalize when translating new sentences beyond the training data.

Table 6 summarizes various statistics and results of the proposed models on the test set. It shows the performance of the FConv-NN model, CNN model, and CNN-ADL model according to different evaluation metrics such as BLEU, Perplexity, METEOR, and TER. A comparison of the results to prior work on NMT for UR-EN translation from Israr et al. (2023) reveals that the proposed FConv-NN model achieved noticeable improvements.

The n-gram BLEU score shows that FConv-NN is superior to CNN and CNN-ADL in performance, where the FConv-NN gives a good score of 40.22. On the UR-EN dataset, the FConv-NN architectures benefit from a fully convoluting layer both in the encoder and decoder. For a single-layer decoder, a fully convolutional encoder outperforms the CNN accuracy by 18.4 BLEU points with a score of 40.22.

CNN-based models i.e. CNN, CNN-ADL, and FConv-NN can process the input in parallel, making them more efficient, especially for longer sequences. This parallel processing capability can lead to faster training and inference times for NMT models, making them more practical for real-time translation applications.

Table 7. Comparison of the model in terms of translation speed in minutes.

Reference	Model trained	Translation time in minutes
Israr et al. (2023)	Bi-RNN	40
	LSTM	43
	CNN	205
	GRU	39
	Transformer	48
Proposed model	**FConv-NN**	**10**

A comparison of runtimes, as shown **Table 7**, indicates that the FConv-NN model is more efficient. On a system with an NVIDIA Geforce GPU, it took the FConv-NN model only 10 minutes to generate translations for the entire test set. In contrast, the CNN baseline required 205 minutes, substantially more time. This shows that FConv-NN is less time-consuming and robust and takes less time to translate unseen data during testing. The CNN and CNN-ADL models, as opposed to FConv-NN, are less efficient, likely due to their convolutional filters taking longer to process each input sequence.

Certainly, FConv-NN outperforms both CNN and CNN-ADL in terms of success rate. It can generate more natural translation with a lower number of perturbations (TER) and higher semantic similarity with the gold reference for all sentences in the test set. The FConv-NN produces astoundingly precise translations.

Three representative sentences are selected from the test set to assess the translation quality of the FConv-NN model relative to the CNN and CNN-ADL models for the UR-EN translation task. The translation quality of these three example sentences were previously evaluated in the paper by Israr et al. (2023). **Table 8**, **Table 9**, and **Table 10** present the translations generated by each model for these example sentences. We have used the BLEURT-20 score to evaluate the translation quality as a success criterion (Wu and Lee, 2022). Comparing the results of the CNN, CNN-ADL, and FConv-NN models shows that using a fully convolutional deep neural network improves translation performance.

Table 8. Sentence 01 translation.

SENT:1	حکومت نے ملک میں لانگ مارچ روکنے کے لئے ہزاروں پولیس کارکنوں کو تعینات کیا ۔	**BLEURT-20 Score**
Gold Reference	The Government deployed thousands of Police workers to stop long march in the country.	
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.	0.462
CNN-ADL	The government set for thousands in to the country.	0.412
FConv-NN	The government deployed thousands of police workers to stop a cooperative March in the country.	0.655

The FConv-NN model generated a translation for the source sentence, “ حکومت نے ملک میں لانگ مارچ روکنے کے لئے ہزاروں پولیس کارکنوں کو تعینات کیا ۔ ” , “The government deployed thousands of police workers to stop a cooperative March in the country ”. A systematic evaluation of the translation for sentence No. 01, as presented in **Table 8**, indicates that the FConv-NN model achieved higher translation accuracy compared to the CNN and CNN-ADL models. Specifically, for sentence No. 01, the FConv-NN model achieved a BLEU score of 24.21, demonstrating superior translation quality when compared to the considerably lower BLEU scores of 0.00 and 0.05 obtained by the CNN and CNN-ADL models, respectively.

Furthermore, the translation produced by the FConv-NN model is compared to the translations generated by two widely used online translation systems: Google Translator and Microsoft Bing Translator. For the source sentence, No. 01, Google Translator has produced the target translation “The government deployed thousands of police workers to stop the long march in the country." with a BLEU score of 0.35. On the other hand, the Microsoft Bing Translator produced the target translation "The government deployed thousands of police personnel to stop the long march in the country." The BLEU score is 0.30.

From a quality perspective, the translation of sentence No. **01** produced by CNN is nonsensical and incoherent. It repeats the phrase "to prevent" multiple times without a clear subject or object. The meaning is unclear. The translation produced by CNN-ADL is grammatically incorrect. The verbs "set" and "for" do not logically connect to the other elements of the sentence. The meaning cannot be determined. Though the translation produced by the FConv-NN uses a vocabulary different from the original, this sentence is grammatically correct and conveys a clear meaning.

Table 9. Sentence No. 02 translation.

SENT:2	کیا آپ تھوڑا سا اونچی آواز میں بول سکتے ہیں ؟	**BLEURT-20**
Gold Reference	Can you speak a little louder?	**Score**
CNN	speak in you a bit more loudly?	0.490
CNN-ADL	Can you speak out loudly.	0.534
FConv-NN	Can you speak a little loud voice?	0.565

As shown in exemplary sentence No. **02** in **Table 9**, the n-gram BLEU scores for the translations generated by the CNN and CNN-ADL models are 0.00 and 0.005, respectively. In contrast, the FConv-NN model achieved a significantly higher n-gram BLEU score of 47.75 for the same sentence, demonstrating a notable improvement in translation quality.

When assessed using the BLEURT metric, the translation score of the FConv-NN model is 0.565, surpassing the translation scores of 0.534 and 0.490 achieved by the CNN and CNN-ADL models, respectively.

From a quality perspective, the translation produced by CNN is grammatically incorrect. "Speak in you" doesn't make syntactic sense. The translation produced by CNN-ADL is grammatically correct but "speak out" is an unusual phrase compared to "speak louder/more loudly". This sentence attempts to duplicate the original structure. It conveys the core meaning. However, it is not fully grammatically correct.

For sentence No. **03**, the translation in **Table 10** produced by the CNN, CNN-ADL, and FConv-NN models has n-gram BLEU scores of 0.00, 0.002, and 38.18, respectively. When evaluated with the BLEURT metric, the translation scores of FConv-NN, CNN, and CNN-ADL are 0.795 and 0.341, respectively. A higher BLEURT score for the target translation indicates a closer similarity to human assessment.

Table 10. Sentence No. 03 translation.

SENT:3	کلاسیکی معاشیات میں ، یہ خیال کیا جاتا ہے کہ معیشت بنیادی طور پر دولت کے علم کے بارے میں ہے .	**BLEURT-20**
Gold Reference	In classical economics, it is believed that the economy is primarily about the knowledge of wealth.	**Score**
CNN	In the classic economics of the wealth.	0.244
CNN-ADL	In the classic situation, it is about.	0.341
FConv-NN	In the classic economics it is believed that the economy is mainly about the knowledge of wealth.	0.795

Qualitative analysis shows that the target translation produced by CNN is grammatically incomplete and does not convey a clear meaning. The translation produced by CNN-ADL is also grammatically incomplete, and the intended meaning cannot be determined. However, the translation produced by

FConv-NN closely mirrors the structure and meaning of the original sentence. It is grammatically sound and clearly communicates the same conceptual point about classical economics.

Table 11. Comparison of the FConv-NN with state-of-the-art models.

Reference	Model trained	BLEU Score
Israr et al. (2023)	Bi-RNN	49.67
	LSTM	47.14
	CNN	21.80
	GRU	00.77
	SRU	28.61
	Transformer	01.27
	CNN-ADL	21.54
	GRU-ADL	44.80
	SRU-ADL	32.40
	Transformer-ADL	02.02
Proposed Model	**FConv-NN**	**40.22**

The comparison between the results of the FConv-NN model and the state-of-the-art RNN-based NMT model trained is illustrated in **Table 11**. The RNN-based models and transformer are trained on the identical Urdu-English dataset, as obtained by Israr et al. (2023).

To outline briefly, the findings showcased in this section highlight the effectiveness of the proposed fully convolutional encoder-decoder network (FConv-NN) for the translation of low-resource language pairs such as Urdu-English. Across all evaluation metrics on the test datasets, the FConv-NN model significantly outperformed baseline CNN and CNN-ADL models. In addition, example translations showed that FConv-NN is able to generate more fluent and coherent outputs compared to the baseline models.

4.1 Ablation Study

In this Ablation study section, we present the analysis of two different sentences and their translation to demonstrate how the proposed system performs better than the other baseline models. **Table 12** and **Table 13** outline two examples from UR→EN translation task. Each example contains output translation from FConv-NN and other RNN-based NMT (i.e. LSTM, GRU, Transformer, SRU and CNN) for the same input sentence. Words with **Bold and red** fonts are correct translations revised by our model. Words with *Italic and blue* fonts are mistranslated words from gold standard. Words in green with underline are correctly translated but at a different place in the reference sentence. The words in brown represent over-translation, while ( ) represent under translation.

Table 12 and **Table 13** outline a comprehensive analysis of the translation produced by different NMT models. From qualitative perspective, the given translation produced by FConv-NN closely resembles the gold reference in terms of structure and content. The translation effectively conveys the intended meaning. The comparison between the target translations and the gold reference sentence shows that the translation generated by the FConv-NN outperforms the state-of-the-art NMT model significantly. This underscores the potential strength of the FConv-NN model over the other baseline NMT model.

As demonstrated in the findings presented by Israr et al. (2023) the Bi-RNN, LSTM, and GRU have exhibited their efficacy in yielding commendable BLEU scores. The FConv-NN model has proved its efficiency by outperforming the aforementioned models in generating contextually correct translations. Moreover, it successfully predicted translations for words ("**خاص طور**", particularly) that were overlooked by the LSTM and GRU models. Notably, the FConv-NN model displayed superior performance with longer sentences compared to other RNN-based models.

Table 12. Comprehensive analysis of target translation generated by different NMT models-I.

Source Sentence	میں کھولا گیا تو ، یورپی یونین ان ممبر ممالک میں شہریوں کے 2011جب مزدوری منڈی کو مشرقی یورپ کے ممالک کے لئے تحفظ کے لئے اقدامات کرنے کا پابند ہوگا جو خاص طور پر مزدور منڈی کے کھلنے سے متاثر ہوں گے ۔
Gold Reference	When the labor market is opened up to the countries of Eastern Europe in 2011, the European Union will be obliged to take measures to protect citizens in those Member States that will be particularly affected by the opening up of the labor market.
CNN	When __ *regulating* __the labor market.
CNN-ADL	When the labor market *was* opened in 2011 __.
FConv-NN	When the labor market *was* opened in 2011 *for* the countries of *eastern* Europe, the European Union will be obliged to take measures to protect citizens in those Member States *which* will be **particularly** affected by the opening up of the labor market.
Bi-RNN	When the labor market *has been* opened in 2011 *for* the countries of *eastern* Europe, the European Union will be *bound* to take measures to protect citizens in those Member States that will be __ affected by the opening up of the labor market.
Bi-RNN ADL	When the labor market *were* opened *into* 2011 *for* the countries of Eastern Europe, the European Union will be obliged to take measures *in order* to protect citizens in those Member States that will __ be affected by the __ labor market in particular.
GRU-ADL	When the labor market *was* opened in 2011 for the countries of Eastern Europe, the European Union will be obliged to take measures to protect citizens in those Member States *which would* be particularly affected by the labor market opening.
GRU	*The first* measures of Member States *are adopted.*
LSTM	When the labor market __ opened in 2011 for the countries of Eastern Europe, the European Union *would* be obliged to take measures to protect citizens in *these* Member States *which* will be __ affected by the opening __ of the labor market.
LSTM-ADL	When labor market *was* opened *up* in 2011 for countries *in eastern* Europe, the European Union will be obliged to take measures to protect citizens in *these* Member States *which* will be particularly affected by the opening __ of the labor market.
Transformer	*Mr. President , I would like* to *thank you* for *your patience* __ .
SRU	When the labor market __ in Eastern Europe *was* opened, the European Union *would* be obliged to take measures for protection of citizens in those Member States which will be particularly badly affected by the opening up of the labor market, to take measures to protect citizens *who* will particularly *be* affected by the opening up of the labor market.

An important finding from the experiment is that the FConv-NN model faces challenges with less frequent and out-of-vocabulary (OOV) words. OOV words are commonly found in language pairs with limited resources, such as Urdu-English. However, the RNN-based NMT model demonstrates effectiveness in handling OOV terms compared to the CNN-based NMT model.

Table 13. Comprehensive analysis of target translation generated by different NMT models-II.

Source Sentence	ستمبر میں قازقستان کے ایک وفد کے دورے پر ، مجھے جیلوں میں انتہائی ظالمانہ سلوک کے حتمی شواہد ملے ۔
Gold Reference	On a delegation visit to Kazakhstan in September, I received conclusive evidence of the most brutal treatment in prisons.
CNN	On *the* visit *of* a delegation __ __*of extending* __ __ the most *cruel* treatment *of a* prisons.
CNN-ADL	On *the* visit *of* a delegation __ __*of extending* __ __the most *cruel* treatment __ *of a richer*.
FConv-NN	On *the* visit *of* a delegation __ in September in September, I *found the final* evidence of __ *extremely* brutal treatment in prisons.
Bi-RNN	On a visit to a delegation *of* Kazakhstan in September, I received the *final* evidence of the most *cruel* treatment in prisons.
Bi-RNN ADL	On visit to a delegation __in September, I received the *final* evidence of the most *cruel behavior* in prisons.
GRU-ADL	On a visit to Kazakhstan, a delegation in September, I have the final evidence of the most brutal behavior in prisons.
GRU	I __ __ __ *rarely stay* __ __ in the __ __ *session*.
LSTM	On *the* visit to a *Romanian* delegation in September, I received *the final* evidence of the most brutal *behavior* in *the* prisons.
LSTM-ADL	On *the* visit of a delegation of a delegation in September, I received *final* evidence of the most *cruel* treatment in prisons.
Transformer	I *would like* to *conclude by thanking* the *rapporteur*.
SRU	On a visit to a delegation visit to *the* Kazakhstan visit in September, I *ge*t the most *cruel* evidence of the prisons.

Table 13 presents the analysis of target translation produced by FConv-NN with gold reference. It is evident from the analysis that the model redundantly translated "in September", which makes the sentence less concise and slightly awkward. This issue exposes a significant weakness of the FConv-NN model. It suggests that when faced with less frequent terms like "Kazakhstan," the model struggles to provide accurate approximations, resulting in repetitive translations.

5. Conclusion

This study explored fully convolutional neural network (FConv-NN) models for low-resource Urdu-English language pair. The goal was to explore avenues for using CNN-based NMT for Urdu-English translation and to identify opportunities for addressing issues in model architecture. The present study designed a fully convolutional CNN model for Urdu-English MT. The empirical tests and comparisons with the rest of the models proved that the FConv-NN model improved the translation quality. This became evident by the low perplexity and a lower over-translation score of the translated text. The model also shows a considerable improvement in the BLEU score. Apart from this, the current study also investigated the role of fully convolutional layers in both the encoder and decoder of the model. As a result of this architectural change, the FConv-NN model performance was improved. It also enabled the model to perform better than the classic CNN for MT. The findings also revealed that adding fully convolutional layer speed up training and testing process. This is because the model processes the input sequence in parallel using convolutional layers with filters. As a result of this property, it becomes very suitable for real-time translation when speed is vital.

The present research also proves the effectiveness of using FConv-NN for improving translation quality in under-resourced languages like Urdu. By using this model, high-quality translation is obtained for Urdu language. This finding opens up future possibilities for improving MT systems for other under-resourced languages.

In addition, the FConv-NN model efficiently translates complex sentences in low-resource languages. This feature makes it one of the most feasible candidates for deployment in real-life applications on translation services. The accuracy and efficiency of the model also point toward its applicability in a wide range of practical uses. These include facilitating global communication, enabling international business transactions, enhancing accessibility for multilingual users, and supporting global tourism.

5.1 Limitation & Future Work

Fully Convolutional Neural Network models have turned out to be surprisingly effective in Urdu-English machine translation (MT). However, it is worth mentioning that there are some limitations associated with the proposed model.

While the FConv-NN model achieved performance over the CNN and CNN-ADL models, there remains room for improvement. When evaluated on the same dataset, RNN-based NMT models such as Bi-RNN with LSTM cells, LSTM, and GRU-ADL generated translations with higher BLEU scores, i.e. 49.67, 47.26, and 44.8, respectively. In contrast, the FConv-NN model achieved a BLEU score of 40.22, indicating the potential to improve the FConv-NN model further to increase the translation accuracy. Although our research is primarily centered on UR-EN MT task, future endeavors will concentrate on enhancing the performance of FConv-NN for MT even further. Moreover, there's potential to investigate FConv-NN models alongside other state-of-the-art NMT models by applying them to other morphologically rich, under-resourced Asian languages. This exploration can significantly contribute to bridging linguistic gaps and advancing language technology in diverse cultural contexts.

Conflict of Interest
The authors confirm that there is no conflict of interest to declare for this publication.

Acknowledgments
The authors would like to express sincere thanks to the School of Electrical Engineering and Computer Science (SEECS), National University of Science and Technology (NUST), and the Department of Information Engineering Technology (IET), National Skills University Islamabad for their outstanding assistance and for providing the resources that enabled us to conduct this research.

References

Ahmed, A., Huang, D., Arafat, S.Y., & Hameed, I. (2024). Enriching Urdu NER with BERT embedding, data augmentation, and hybrid encoder-CNN architecture. *ACM Transactions on Asian and Low-Resource Language Information Processing, 23*(4), 1-38. https://doi.org/10.1145/3648362.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *Computation and Language.* arXiv preprint arXiv:1409.0473.

Banerjee, S., & Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *Paper Presented at the Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. 65-72. https://aclanthology.org/w05-0909.

Beddiar, D.R., Jahan, M.S., & Oussalah, M. (2021). Data expansion using back translation and paraphrasing for hate speech detection. *Online Social Networks and Media*, *24*, 100153. https://doi.org/10.1016/j.osnem.2021.100153.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In: Montavon, G., Orr, G.B., Müller, K.R. (eds) *Neural Networks: Tricks of the Trade*. Springer, Berlin, Heidelberg, pp. 437-478. https://doi.org/10.1007/978-3-642-35289-8_26.

Bunrit, S., Kerdprasop, N., & Kerdprasop, K. (2023). Systematic configuration for hyperparameters optimization in transferring of CNN model to disaster events classification from UAV Images. *Journal of Image and Graphics, 11*(3), 263-270. https://doi.org/10.18178/joig.11.3.263-270.

Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014a). On the properties of neural machine translation: Encoder-decoder approaches. *Computation and Language.* arXiv preprint arXiv:1409.1259.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014b). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Computation and Language*. arXiv preprint arXiv:1406.1078.

Costa-Jussa, M.R., & Fonollosa, J.A.R. (2016). Character-based neural machine translation. *Computation and Language.* arXiv preprint arXiv:1603.00810.

Dowlagar, S., & Mamidi, R. (2021). Gated convolutional sequence to sequence based learning for english-hingilsh code-switched machine translation. In: Solorio, T., Chen, S., Black, A.W., Diab, M., Sunayana, S., Victor, S., Emre, Y., Anirudh, S. (eds) *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*. Association for computational linguistics, Online, pp. 26-30. https://doi.org/10.18653/v1/2021.calcs-1.4.

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y.N. (2017). Convolutional sequence to sequence learning. In: Precup, D., Teh, Y.W. (eds) *Proceedings of the 34th International Conference on Machine Learning* (pp. 1243-1252). Sydney, Australia.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Practical methodology. *Deep Learning*, 416-437. MIT Press, Cambridge MA.

Haddow, B., Bawden, R., Barone, A.V.M., Helcl, J., & Birch, A. (2022). Survey of low-resource machine translation. *Computational Linguistics*, *48*(3), 673-732. https://doi.org/10.1162/coli_a_00446.

Hu, R. (2023). Exploring English - Chinese translation skills based on big data technology. *Applied Mathematics and Nonlinear Sciences. 9*(1), 1-14. https://doi.org/10.2478/amns.2023.2.00196.

Hujon, A.V., Amitab, K., & Singh, T.D. (2023). Convolutional sequence to sequence learning for English - Khasi neural machine translation. In *2023 4th International Conference on Computing and Communication Systems* (pp. 1-4). IEEE. Shillong, India. https://doi.org/10.1109/i3cs58314.2023.10127426.

Israr, H., Khan, S.A., Tahir, M.A., Shahzad, M.K., Ahmad, M., & Zain, J.M. (2023). Neural machine translation models with attention-based dropout layer. *Computers, Materials & Continua, 75*(2), 2981-3009.

Jelinek, F., Mercer, R.L., Bahl, L.R., & Baker, J.K. (1977). Perplexity-a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America, 62*(S1), S63.

Johri, P., Khatri, S.K., Al-Taani, A.T., Sabharwal, M., Suvanov, S., & Kumar, A. (2021). Natural language processing: History, evolution, application, and future work. In: Abraham, A., Castillo, O., Virmani, D. (eds) *Proceedings of 3rd International Conference on Computing Informatics and Networks*. Springer, Singapore, pp. 365-375. ISBN: 978-981-15-9712-1(e), 978-981-15-9711-4(p). https://doi.org/10.1007/978-981-15-9712-1_31.

Kaiser, Ł., & Bengio, S. (2016). Can active memory replace attention? *Machine Learning*. arXiv:1610.08613v2.

Kaiser, L., Gomez, A.N., & Chollet, F. (2017). Depthwise separable convolutions for neural machine translation. *Computation and Language.* arXiv preprint arXiv:1706.03059.

Kalchbrenner, N., Espeholt, L., Simonyan, K., Oord, A.V.D., Graves, A., & Kavukcuoglu, K. (2016). Neural machine translation in linear time. *Computation and Language*. arXiv preprint arXiv:1610.10099.

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. *Computation and Language*. arXiv preprint arXiv:1404.2188.

Karpukhin, V., Levy, O., Eisenstein, J., & Ghazvininejad, M. (2019). Training on synthetic noise improves robustness to natural noise in machine translation. In: Wei, X., Alan, R., Tim, B., Afshin, R. (eds) *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*. Association for Computational Linguistics, Hong Kong, China, pp. 42-47. https://doi.org/10.18653/v1/D19-5506.

Krasadakis, P., Sakkopoulos, E., & Verykios, V.S. (2024). A survey on challenges and advances in natural language processing with a focus on legal informatics and low-resource languages. *Electronics*, *13*(3), 648.

Laskar, S.R., Khilji, A.F.U.R., Pakray, P., & Bandyopadhyay, S. (2022). Improved neural machine translation for low-resource English-Assamese pair. *Journal of Intelligent & Fuzzy Systems, 42*(5), 4727-4738.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436-444. https://doi.org/10.1038/nature14539.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278-2324. https://doi.org/10.1109/5.726791.

Luong, M.T., Pham, H., & Manning, C.D. (2015). Effective approaches to attention-based neural machine translation. *Computation and Language*. arXiv preprint arXiv:1508.04025.

Meng, F., Lu, Z., Wang, M., Li, H., Jiang, W., & Liu, Q. (2015). Encoding source language with convolutional neural network for machine translation. *Computation and Language.* arXiv preprint arXiv:1503.01838.

Mohamed, Y.A., Khanan, A., Bashir, M., Mohamed, A.H.H.M., Adiel, M.A.E., & Elsadig, M.A. (2024). The impact of artificial intelligence on language translation: A review. *IEEE Access*, *12*, 25553-25579.

Naeem, M., Siddique, A.B., Ali, R.H., Arshad, U., ul Abideen, Z., Khan, T.A., Shah, M.H., Ijaz, A.Z., &. Ali, N. (2023). Performance evaluation of popular deep neural networks for neural machine translation. In *2023 International Conference on Frontiers of Information Technology* (pp. 220-225). IEEE. Islamabad, Pakistan.

Pandey, A.K., & Roy, S.S. (2023). Natural language generation using sequential models: A survey. *Neural Processing Letters*, *55*(6), 7709-7742. https://doi.org/10.1007/s11063-023-11281-6.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311-318). Association for Computational Linguistics. Philadelphia, Pennsylvania, USA. https://doi.org/10.3115/1073083.1073135.

Sánchez-Cartagena, V.M., Pérez-Ortiz, J.A., & Sánchez-Martínez, F. (2024). Understanding the effects of word-level linguistic annotations in under-resourced neural machine translation. *Computation and Language.* https://arxiv.org/abs/2401.16078v1.

Subramanian, K., Stallard, D., Prasad, R., Saleem, S., & Natarajan, P. (2007). Semantic translation error rate for evaluating translation systems. In *2007 IEEE Workshop on Automatic Speech Recognition & Understanding* (pp. 390-395). IEEE. Kyoto, Japan. https://doi.org/10.1109/asru.2007.4430144.

Sutskever, I., Vinyals, O., & Le, Q.V. (2014). Sequence to sequence learning with neural networks. *Computation and Language*. https://arxiv.org/abs/1409.3215v3.

Tan, T.P., Lim, C.K., & Rahman, W.R.E.A. (2022). Sliding window and parallel LSTM with attention and CNN for sentence alignment on low-resource languages. *Pertanika Journal of Science & Technology, 30*(1), 97-121.

Tang, G., Müller, M., Rios, A., & Sennrich, R. (2018). Why self-attention? A targeted evaluation of neural machine translation architectures. *Computation and Language*. arXiv preprint arXiv:1808.08946.

Tiwari, G., Sharma, A., Sahotra, A., & Kapoor, R. (2020). English-Hindi neural machine translation-LSTM seq2seq and ConvS2S. In *2020 International Conference on Communication and Signal Processing* (pp. 871-875). IEEE. Chennai, India. https://doi.org/10.1109/iccsp48568.2020.9182117.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Computation and Language*. https://arxiv.org/abs/1706.03762.

Veliz, C.M., De Clercq, O., & Hoste, V. (2021). Is neural always better? SMT versus NMT for Dutch text normalization. *Expert Systems with Applications, 170*, 114500. https://doi.org/10.1016/j.eswa.2020.114500.

Vylomova, E., Cohn, T., He, X., & Haffari, G. (2016). Word representation models for morphologically rich languages in neural machine translation. *Neural and Evolutionary Computing.* arXiv preprint arXiv:1606.04217.

Wang, C., Jain, A., Chen, D., & Gu, J. (2019). Vizseq: A visual analysis toolkit for text generation tasks. *Computation and Language*. https://arxiv.org/abs/1909.05424v1.

Wu, S.H., & Lee, J.J. (2022). An automatic evaluation method for open-domain dialogue based on BLEURT. In *2022 IEEE 23rd International Conference on Information Reuse and Integration for Data Science* (pp. 83-89). IEEE. San Diego, CA, USA. https://doi.org/10.1109/iri54793.2022.00030.

Yang, S., Wang, Y., & Chu, X. (2020). A survey of deep learning techniques for neural machine translation. *Computation and Language*. https://arxiv.org/abs/2002.07526v1.