Information Sciences Letters
An International Journal

http://dx.doi.org/10.18576/isl/130501

Urdu-to-English Neural Machine Translation using Transformer with Subword Tokenization

Huma Israr, Novera Parvaz and Safdar Abbas Khan*

School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan.

Received: 1 Mar. 2024, Revised: 1 Aug. 2024, Accepted: 23 Aug. 2024
Published online: Sep. 2024

Abstract: Neural machine translation (NMT) model uses deep learning algorithms to translate text from one language to another. With continuous advancements in this field, numerous state-of-the-art techniques have been developed to make translations more accurate and faster. However, the development of Urdu-to-English (UR-EN) machine translation (MT) systems has remained limited compared to other language pairs. The complexity of Urdu language, characterized by its unique writing system and intricate morphology contributes to this limitation. Furthermore, the lack of large, standardized datasets and linguistic resources for Urdu makes it hard to create effective UR-EN translation models. This research introduces a specialized NMT model for translating Urdu text to English. It uses a transformer-based method with subword tokenization to improve the accuracy of previous Urdu-to-English translation models. This study achieved an impressive BLEU score of 45.58, showing that the transformer with subword tokenization performs well for UR-EN translation. The trained model outperformed the classical Transformer with word-level tokenization and the Transformer with attention-based dropout layer by +43.48 BLEU scores. This noteworthy achievement underscores the effectiveness of the proposed approach and demonstrates its potential for practical deployment in UR-EN translation tasks.

Keywords: Low Resource Language, Multi-Head Attention, Neural Machine Translation, Positional encoding, Sub-Word Tokenization, Transformer, Urdu, Word Embedding

1 Introduction

Machine translation (MT) is an automated process that translates bilingual text from one language to another using computer algorithms. Machine translation techniques have become increasingly proficient, particularly in translating widely spoken languages that have abundant training data. This progress has greatly improved translation accuracy and quality for language pairs such as English, German, French, and Chinese. There are sufficient linguistic resources available for these languages. According to Ethnologue [1], there exists 7,168 living languages spoken around the world. Many of these languages are considered low-resource in the field of natural language processing (NLP). A number of these languages lack important linguistic resources, such as large annotated datasets and pre-trained models. Due to the limited linguistic resources available for low-resource languages, it can be difficult to apply state-of-the-art NLP techniques and achieve high performance.

Urdu, despite being spoken by millions of people, falls into the category of low-resource languages. The lack of linguistic resources and tools available for Urdu presents difficulties. Developing robust and reliable Urdu NLP systems, such as machine translation (MT), sentiment analysis, and named entity recognition (NER), can be challenging without sufficient data and resources [2].

While facing resource limitations, there are ongoing efforts to generate resources and develop NLP tools specifically for Urdu. These initiatives involve creating linguistic corpora, sentiment lexicons, and MT systems. The goal is to bridge the resource gap and improve the accessibility and quality of NLP applications for Urdu speakers. Urdu poses unique challenges for natural language processing systems. Its grammar and writing system

* Corresponding author e-mail: hisrar.dph17seecs@seecs.edu.pk

include many complex variations. However, machine translation systems for Urdu show potential to address these obstacles and enable automated translation between Urdu and other languages. Urdu MT systems aim to bridge the language barrier by generating approximate translations that convey the general meaning of the original text [3].

MT has observed substantial developments in recent times, particularly in the field of translating widely spoken or universal languages like English, German, French, Chinese, and many others [4]. These languages often possess abundant training data, which allows MT models to learn and generalize patterns with great effectiveness [5]. As a result, MT systems have become increasingly proficient at producing accurate and high-quality translations in these language combinations. The availability of wide range of training resources is important for advancing the capabilities of MT. These resources are important for enhancing its performance in translating these widely used languages [6].

Classical MT encompasses various approaches for automatically translating text from one language to another. The prominent types of MT include rule-based machine translation (RBMT), which utilizes dictionaries and linguistic rules created by human linguists [7]. While effective in structured contexts, RBMT often struggles with the complexity and variability of natural language. To address these limitations, statistical machine translation (SMT) was developed. SMT uses statistical models and algorithms to learn translation patterns from parallel corpora [8]. SMT marked a significant improvement, allowing for more flexible and data-driven translations. Example-based machine translation (EBMT), focuses on reusing previously translated segments from a bilingual corpus [9]. Phrase-based machine translation (PBMT), a subset of SMT, looks for repeated phrases or sentences in parallel corpora. PBMT extracts common translation pairs or "phrases." [10] from source sentences. he advent of neural machine translation (NMT) represented a major leap forward. NMT employs neural networks with an encoder-decoder architecture to generate translation. Finally, hybrid machine translation combines multiple techniques to leverage their respective strengths and enhance translation quality. Researchers continue to work on improving MT systems using these diverse approaches.

NMT consists of ML methods that learn the translation directly from large bilingual text corpora. In particular, NMT systems are based on neural networks (NN) that are trained in a supervised fashion. During training, the network is shown sentence pairs with a source sentence in one language and the corresponding reference translation in the target language. The network learns to translate source sentences into the target language by matching its output to the reference translation [11]. Two prominent NN architectures used in NMT are recurrent neural networks (RNNs) and convolutional neural networks (CNNs) [12]. RNN-based models, such as the long short-term memory (LSTM) [13] and gated recurrent unit (GRU) [14], are widely employed in NMT. These models have seen substantial improvements in recent years, largely facilitated by the emergence of the attention mechanism [15] and the Transformer model [16]. While current NMT systems have reached close-to-human quality in translation [17], there is still room for improvement when it comes to translating low-resource languages. Many studies have successfully applied Transformers to various language pairs, demonstrating their flexibility in learning relationships between different language pairs. Transformers are the most widely adopted frameworks in NMT today. NMT using Transformer models remains an active area of development that holds great promise for continuing to push forward gains in translation quality [18].

While NMT has achieved significant advances for numerous language pairs, Urdu-English MT remains a comparatively understudied domain. A few research studies attempt to use NNET to translate Urdu-to-English and vice-versa. Andrabi et al. implemented LSTM for UR-EN translation task [19]. Khan et al. uses artificial neural network (ANN) for English-Urdu MT task [20]. In a subsequent study, khan et al. implemented LSTM and GRU to translate Urdu-to-English MT [21]. Rauf et al. presented a comparative study of English-to-Urdu using encoder-decoder attention-based NMT [2]. In a recent attempt, Naeem et al. experimented with English-to-Urdu MT task using RNN-based NMT i.e. LSTM and GRU [22]. From the literature review it is clear that there has been limited exploration in translating Urdu-to-English using RNNs. Furthermore, there is virtually no research on employing transformer for Urdu-to-English MT. This research aims to help address this gap through the following key contributions:

1. Developing an Urdu-to-English NMT model leveraging the Transformer architecture, which has proven effective for other language pairs but has seen limited application to Urdu-English translation.
2. Fine-tuning the baseline Transformer model on a large parallel Urdu-English text corpus using subword tokenization.
3. Evaluates the fine-tuned Transformer model while it is learning and exploring the structural distinctions between Urdu and English.
4. and thoroughly analyze the translation results generated by the trained model.

This study is divided into the following sections: **Section 2** reviews relevant prior studies to provide context for the contributions of this work. **Section 3** describes the Transformer architecture and how it is applied to

Table 1: Review of Literature

Framework	Ref	Languages	Dataset
Transformer	[23]	Urdu-English	Own Dataset
	[24]	Dutch (Nl)-German (De), Romanian (Ro)-Italian (It)	IWSLT 2017
	[25]	English-German	WMT 2014
	[26]	(English-Thai, Myanmar),(Thai-English, Myanmar), (Myanmar-English, Thai)	ASEAN-MT Parallel Corpus
	[27]	Hindi-Marathi	Own dataset
	[28]	AKAN-Twi to English	Own dataset
	[29]	Chinese-English	NIST'12 (Zh-En-Small) benchmarks
With BPE	[30]	English-German, Japanese-English, Sinhala-English Transformer	WMT'14, (newstest13, newstest14) Michel and Neubig-2018, FLoRes
	[31]	English–Irish	Dataset from the Directorate General for Translation (DGT)
	[32]	Hindi English	IIT Bombay English-Hindi Corpus
	[33]	Chinese-English	NIST12 (Zh-En), MT06 and MT08
	[34]	De-En, En-De, Fr-En, En-Fr	WikiText-103 language task
Charchter-level Transformer	[35]	De-En, Cs-En, FI-En, Ru-En	WMT15
	[36]	English-German and English-Czech; English-Turkish.	WMT14, WMT18
	[37]	Czech and Croatian, German, Hungarian, Slovak, and Spanish	MultiParaCrawl corpus
	[38]	English to Fr, Ro, Fi, Tr, He, Ar, Vi, MI	OPus 100 corpus

machine translation. **Section 4** examines the results generated by the implemented Transformer model using subword tokenization. Lastly, **Section 5** summarizes the paper and reflects on some challenges encountered when employing the Transformer and suggests opportunities for further exploration on this topic.

2 Related Work

This section reviews related work on applying Transformer to machine translation and recent developments in Transformer architectures. Table 1 and Table 2 summarize the key aspects of different techniques and methods from the literature.

2.1 Transformer Models

Transformers have become the leading neural machine translation (NMT) architecture in recent years. Since their introduction, Transformer models for NMT have consistently achieved top-tier results in MT and researchers continue advancing the methodology each day. Recently, considerable work has explored leveraging Transformers for various language pair translations.

Israr et al. conducted an extensive experiment comparing RNN, CNN, and Transformer-based models for Urdu-English machine translation. They trained six different NMT models including transformers for the same Urdu-English dataset. Through their trials translating Urdu to English, Israr et al. achieved the highest BLEU score of 47.0. The Transformer performed significantly worse than the other approaches, achieving notably lower BLEU scores and translation quality. Based on the findings the study determined that of all the models evaluated, the Transformer technique proved least effective for Urdu-English translation task [23] [39]. Lakew et al. experimented with Transformers and RNNs for multilingual NMT. They analyzed the performance and translation quality of Transformers for six language pairs through statistical and interpretative analysis of translations from bilingual, multilingual, and zero-shot systems. Their findings concluded that the Transformer approach delivered the highest performance across all techniques evaluated [24].

Ahmed et al. proposed the weighted Transformer for translating En-to-Du. The proposed model implements a modified version of multi-Head attention. Instead of using a regular multi-head attention, the model computes and combines multiple self-attention during training. The proposed model outperforms the conventional Transformer with multi-head attention. The model trains faster and converges earlier during training and achieves better performance than the original Transformer network [25].

San et al. explored Transformer and their variational models for translating low-resource languages i.e. Thai Myanmar and English language. The datasets available for these languages are very small. To handle the limited resources they considered enhancing the existing dataset using data augmentation techniques such as SwitchOut and Ciphertext (CipherDAug). On all datasets, the multi-source transformer with CipherDAug achieved the best BLEU [26]. In subsequent efforts to translate the low-resource languages using a transformer Dhanani et al. implemented the model for Marthi-Hindi translation task [27], agyei et al. for AKAN-English [28], Wang et al. for Chinese-English [29] translation task.

Table 2: Summary of Research findings

S.No.	Ref	Model Used	Research findings
1.	[23]	Transformer, CNN, LSTM, GRU	Low BLEU, Higher TER, Translation Quality
2.	[24]	Transformer, RNN	Improved BLEU score, lower TER, mTER, and lTER score
3.	[25]	Transformer	Improved BLEU score, faster Training
4.	[26]	Transformer, mBART	BLEU, TER, chrF
5.	[27]	Transformer	Improved BLEU, TER score
6.	[28]	Transformer	Improved BLEU
7.	[29]	Transformer	Improved BLEU Score, faster and smaller model
8.	[30]	Transformer	Improved BLEU, Translation Quality
9.	[31]	Transformer, RNN	Improved BLEU, TER, Runtime Complexity
10.	[32]	Transformer	Improved BELU RIBES
11.	[33]	Weights sharing Transformer	Improved BLEU
12.	[34]	Transformers with relative positional encoding	Improved perplexity, time complexity
13.	[36]	Transformer	Improved BLEU, METEOR
14.	[37]	Transformer	Improved BLEU CHRF COMET
15.	[38]	Transformer	Improved BLEU, Recall CHRF
16.	[40]	Transformer	Improved BELU score

2.2 Transformer with byte-pair-encoding (BPE)

To handle data sparsity issues that exist in morphologically rich languages and enhance the performance of NMT models, a byte-pair encoding (BPE) scheme is used [41]. It has become standard practice in neural machine translation (NMT) to build a vocabulary using byte-pair encoding (BPE). Generally, BPE operates on the character-level instead of the byte-level. Wang et al. explore the use of byte-level "subwords" to tokenize text into variable-length n-grams byte, rather than character-level subwords which represent text as sequences of character n-grams. This byte-level approach results in a much more compact vocabulary size compared to character-based models, without sacrificing performance. Experiments were conducted using Fairseq [42] to train Transformer models. The results of their experiments show that BBPE has comparable performance to BPE. While the vocabulary size is shorten to one-eighth the size of BPE. Additionally, in multilingual settings BBPE maximizes shared vocabulary across languages, resulting in better translation quality [30].

Lankford et al. employed a Transformer-based NMT model for the English-Irish translation task. They preprocessed the data using SentencePiece models with both Byte Pair Encoding (BPE) and unigram approaches. The research findings demonstrated that the Transformer performed significantly better at reducing errors in both accuracy and fluency when compared to an RNN-based model [31]. Gangar et al. developed an NMT system using the Transformer model to translate Hindi-to-English. Hindi is a low-resource language, posing challenges for NMT. The paper implements back-translation and experiments with word-level and subword-level tokenization using Byte Pair Encoding (BPE) in ten Transformer configurations [32].

The attention module in Transformers does not scale efficiently to long sequences due to its quadratic complexity. Many works try to approximate the attention calculation to reduce complexity. Pen et al. proposed a novel way to accelerate attention calculation for Transformers. This approach relies on computing the attention score using kernalized relative positional encoding [34].

2.3 Character-level Transformer-based NMT

A promising alternative approach to using BPE focuses on character-level translation. Employing character-level translation simplifies the processing pipelines in NMT. Banare et al. proposed a new character-level Transformer-based NMT model called "CharTransformer". In their experiments, they preprocessed the data using a source length reduction technique and trained a six-layer character-level Transformer. The researchers then compared translations produced by the CharTransformer to those from a CharRNN and a subword-level Transformer. For four language pairs De-En, Cs-En, Fi-En, and Ru-En. they found that the CharTransformer translations were more accurate than CharRNN and Transformer with sub-wording [35].

Character-level Transformer model typically requires deep architectures, which can be challenging and time-consuming to train. Libovick et al. implemented a 6-layer vanilla Transformer for character-level translation. Their proposed model shows significance in capturing the morphological phenomena and was found to be robust

Table 3: List of variables

S.No.	Variable	Description	S.No.	Variable	Description
1.	x	source sentence	5.	q	query for self-attention
2.	y	target sentence	6.	k	value for self-attention
3.	PE	positional encoding	7.	K	Key for Multi-Head Attention
4.	Q	Query for Multi-Head Attention	8.	V	Values for Multi-Head Attention

but the quality of translation produced was a little worse [36]. Bojar et al. explore the effectiveness of character-level Transformer for language pairs with different similarity levels and varying sizes of training data. The languages evaluated are Czech translated to and from Croatian, German, Hungarian, Slovak, and Spanish. The trained models are then evaluated using automatic MT metrics. Their study confirms that character-level NMT works best for closely related languages, while subword segmentation is better for distant language pairs [37]. Li et al. Compares character-based and subword-based transformer model under low-resource, cross-lingual, and domain adaptation settings. Experimental results show that character-based NMT is competitive with and sometimes outperforms subword-based NMT, especially for handling morphology, rare/unknown words, and transferring to new domains [38].

An analysis of existing literature uncovers that machine translation for numerous language pairs has been extensively explored using neural models like Transformer. However, the area of Urdu-English NMT remains understudied and leaves room for improvement. This research aims to help fill the gap by developing a neural machine translation model for Urdu-English using the Transformer architecture.

3 Methodology

This part of the paper consists of the details of tools, the implemented Transformer model, as shown in Figure 1, and the proposed subword tokenization technique for Urdu-English data prepossessing. To effectively describe the methodology and ensure clear comprehension, a list of variables are presented in Table 3, alongside a description of each of each varaible.

3.1 Tools

In this research we used OpenNMT [43] which is an open-source toolkit for NMT, providing a flexible and powerful framework for building and training NMT models. It supports various neural network architectures, including RNNs and Transformers, and offers tools for data pre-processing, model training, and evaluation.

3.2 The Transformer

The Transformer with attention mechanism is an NMT model that consists of an encoder, and decoder modules. The encoder takes source sentence $x = (x_1, x_2, ..., x_n)$ and generates a target sentence $y = (y_1, y_2, ..y_n)$ as shown in Figure 1. The encoder processes the input text and produces an encoded output. This encoded output serves as the input for the decoder. The decoder, in turn, generates a target sentence based on the encoding of the source sentence.

3.2.1 Word embedding and Positional encoding

Like RNNs, Transformers work on sentences all at once. Typical RNN and CNN-based model contains recurrence and convolution that help the model preserve the positional information (either relative or absolute position). The transformer computes the positional encoding, which helps the model to learn the position of words/tokens in a sentence. Now even though Transformer looks at the whole sentence together, it still understands which word comes first. Positional encoding in both the encoder and decoder of the Transformer is achieved through sine and cosine formulas as follows:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_model}) \quad (1)$$

$$PE_{(pos,2i)} = cos(pos/10000^{2_i/d_model}) \quad (2)$$

Here, *pos* refers to the position of the input in the sequence, i refers to the index of the dimension in the embedding vector, and d refers to the dimensionality of the model. Essentially, each dimension of the positional encoding is associated with a sinusoidal pattern. In Transformer model, the positional encoding is added to the embeddings through a simple addition operation.

3.2.2 Encoder

The encoder takes each word (in the form of embedding) in the input sentence along with positional encoding as input, processes it, and produces an intermediate representation of words in the input sentence. A typical encoder of a transformer consists of several stacked layers. Every stacked layer contains two components, a multi-head self-attention mechanism and a fully-connected feed-forward network (FFN). A residual connection is connection is employed around each component, followed by layer normalization. All components in a layer produce outputs of the same dimension.

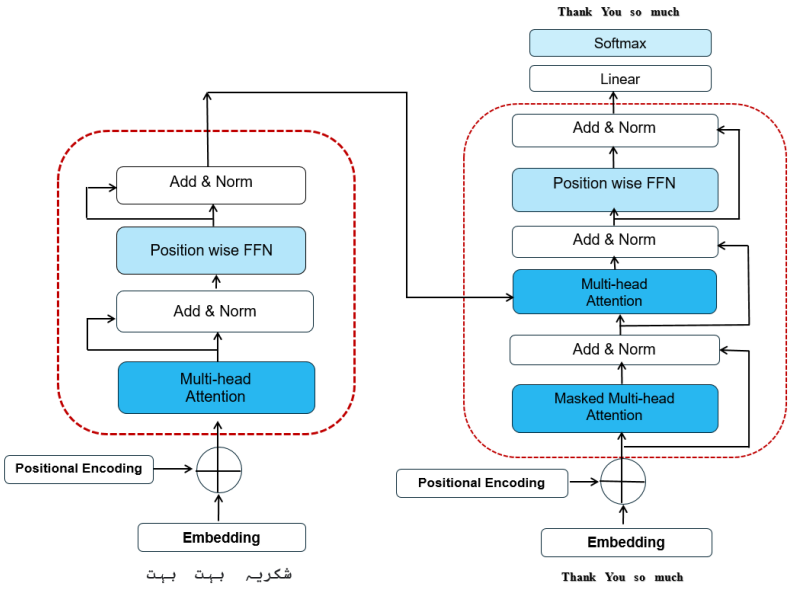


Figure 1: The Transformer for NMT

3.2.3 Self-Attention

Attention in NMT model is a powerful mechanism that enables the NN to selectively focus on relevant parts of a sequence when performing tasks such as language translation or image captioning. In transformer, the scaled dot product attention function computes the attention weight for a sequence of queries (q), keys (k), and values (v).

$$q.k = \sum_{k=1}^{d_k} q_i.k \quad (3)$$

It computes the weighted sum of the value vectors using the attention weights. The function returns the output and attention weights.

3.2.4 Multi-Head-Self-Attention

The idea behind multi-head attention as shown in Figure 2 is to perform several different attention operations in parallel, with each attention head focusing on a different aspect of the input sequence. The Multi-Head attention in a transformer is a mapping function. It maps a vector query Q and a set of key-value pairs $K-V$ to an output

$$attention(Q,V,K) = softmax(QK^T/d_k)^V \quad (4)$$

while d is the dimension of embedding.

Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions.

$$MultiHead(Q,K,v) = Conact(head_1,...,head_n)W^o \tag{5}$$

multi-head attention is to perform several different attention operations in parallel i.e. $head_1 \ldots head_n$.

$$head_1 = attentionend(QW_i^Q, KW_i^K, VW - i^V) \tag{6}$$

The inputs and outputs of both the multi-head attention layer are processed by Add & Norm layers which contain a residual structure and a normalization layer.

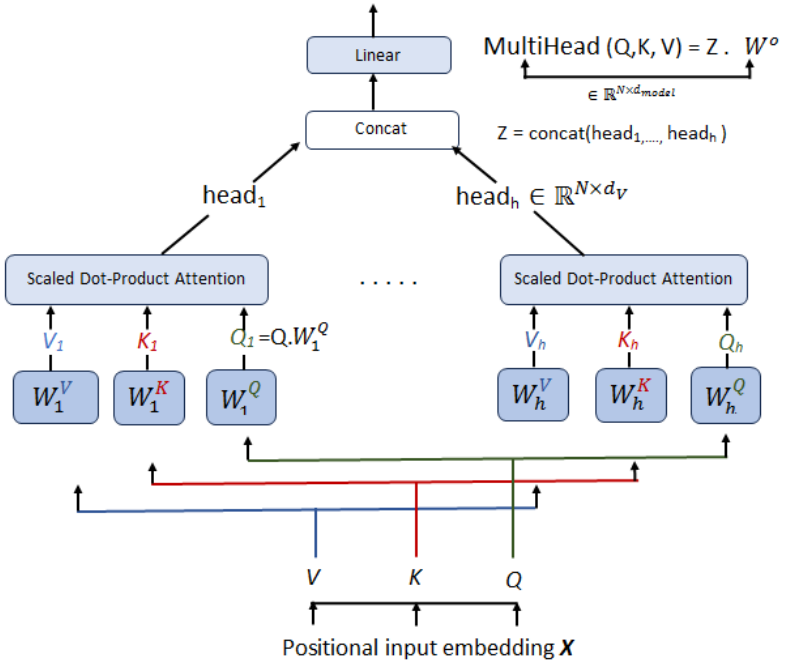


Figure 2: Multi-Head Self-Attention

3.2.5 Residual Connections

The Transformer adds an additional layer "Add & Norm" to the output's sub-layer. It consists of a bypass that is called a residual connection from the original input and the output from the previous sublayer.

3.2.6 Feed-forward network

Within both the encoder and decoder, every layer comprises a fully connected feed-forward network, which operates independently. The FFN in transformer encoder-decoder involves two linear transformations separated by a ReLU activation function.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{7}$$

The inputs and outputs of the Position-Wise Feed-Forward network(FFN) are processed by Add & Norm layers. The Add & Norm Layers mitigate the vanishing gradient problem and improve training stability.

3.2.7 Decoder

The output of the Encoder is a sequence of hidden states that contains information about the context of each input token. This sequence is then passed to the transformer decoder for further processing, where it is used to generate an output sequence. The decoder in a transformer is a separate module that generates the output sentence word by word while considering the representation vector that was created by the encoder and the decoder's last output.

The Transformer decoder contains three components that operate sequentially at each timestep. A Masked Multi-Head Self-Attention Mechanism, Multi-Head Self-Attention Mechanism, and a Fully-Connected Feed-Forward Network (FFN).

The decoder is different from the encoder. It has an extra step called masking. Masking hides future words so the decoder can't see the whole sentence. This is important for training. Without masking, the decoder would already know the answer. It would just copy the sentence. Masking makes the decoder guess words one at a time from left to right. This helps it learn instead of just copying. During inference, the masked self-attention mechanism is not used as the model is generating the output sequence token by token, and hence it does not have access to the future tokens.

3.2.8 Output

The decoder's output undergoes transformation via a linear layer, which aligns the high-dimensional vector space with the size of the output vocabulary. Subsequently, a softmax activation function is applied to yield the final probability distribution over the output vocabulary. The model then selects tokens from this distribution iteratively until an end-of-sequence token is generated.

3.2.9 Loss

The SparseCategoricalCrossentropy loss function is used to train the transformer. It measures the difference between the predicted target sequence and the true target sequence. The loss function is computed using the cross-entropy formula, which is given by

$$\sum_{i=1}^{n} log(p_i) \tag{8}$$

where y_i is the true label of the i-th class, and p_i is the predicted probability of the i-th class.

4 Experimental setup

In this section, the Urdu-English data set used, proposed data Pre-processing step i.e. Subword Tokenization, trained model parameters, and hyperparameter are discussed.

4.1 Data set

This study utilized a parallel Urdu-English corpus[1] developed by [23] containing 10,000 sentence pairs. The corpus contained sentences from common news, sports, movie dialogue and representatives sentences of everyday conversations. It comprised 90,000 unique sentences for training. The hold-out validation set comprise of 5,000 sentences and another 5,000 sentences for testing . The source Urdu and target English sentences were stored separately but aligned line-by-line such that each sentence in one language file had its translation in the corresponding line of the other file, as illustrated in Table 4 and Table 5.

4.2 Data Prepossessing

In NMT tasks, handling raw Urdu text directly is very challenging due to the complexity of the language. In traditional MT approach, raw text is often tokenized using simple techniques such as whitespace-based tokenization, where sentences are split into words based on spaces or punctuation marks. To better generalize, subword tokenization technique is used. This approach allows the model to handle rare words in morphologically

[1] https://github.com/Huma-Israr/Urdu-to-English-NMT-using-Transformer-with-Subword-Tokenization

Table 4: Examples of source sentences from Urdu-English train set

Source sentence in Urdu
ہر وقت ایماندار رہنا آسان نہیں ہے ۔
یہ وہی خاندان ہے جو بنیادی طور پر فیصلہ کرتا ہے کہ آیا لڑکیوں کو تعلیم دی جا سکتی ہے ۔
کاشتکاروں کو اعتماد کرنا چاہئے کہ وہ جو فیڈ استعمال کر رہے ہیں اس میں وہی چیز ہے جو لیبل پر بیان کی گئی ہے ۔
ہمارے آج تک کے تجربے نے ، انفرادی اور اجتماعی طور پر ، ہمیں اس کے لیے تیار نہیں کیا ہے کہ ہم کو کیا کرنا چاہیئے ، یا ہمیں کیا بننا چاہیئے ۔

Table 5: Examples of target sentences from Urdu-English train set

Target sentence English
“ It ’ s not easy to be honest all the time . ”
“ it is the family that decides , in the main , whether girls can be educated . ”
“ Farmers must be able to trust that the feed they are using contains what is stated on the label . ”
“ Nevertheless , fiscal policy should be maintained on a sustainable course , anchoring expectations of an ordered resolution of the crisis . ”

rich languages. It also handles unseen words effectively. Subword tokenization breaks down words into smaller subword units. SentencePiece is a popular subword tokenization algorithm. It works by repeatedly merging the most common character pairs in the training data until the vocabulary reaches a specific size. This process creates subword units, which can represent both individual characters and meaningful linguistic components. For example, the word “unhappy” might be split into subword tokens like [“un”,“_happy”], where“_” denotes that “happy” is a continuation of the previous token. This way, the model can recognize that “unhappy” is a single concept composed of two subword units. The basic method for subword tokenization is described in the following algorithm.

Algorithm 1 Subword Tokenization

Input: Corpus containing Source and target sentences ($C_{S,Twords}$)
Output: Tokenized subword of source and target sentences $C_{S,TSubwords}$

function CREATE_SUBWORDS($C_{S,T}$)
  Token ={} define an empty list
  for each word in $C_{S,T}$ **do**
    Token += Tokenize (word)
  end of for
  UniqueToken ←**Unique**(Token) List of unique subword tokens
  $C_{S,TSubwords}$ ←**Generate**(*UniqueToken*)
  return $C_{S,TSubwords}$
end function

Table 6: Source sentences from train set after subword tokenization

Source sentence after subword Tokenization
ہر _وقت _ایماندار _رھن ا _ آسان _نہ یں _ہے _۔_
_یہ _وھی _خاندان _ہے _جو _بنیادی _طور _پر _فیصلہ _ کرتا _ہے _ کہ _آیا _لڑکی وں _ کو _تعلیم _دی _جا _سکتی _ہے _۔
_ کاشتکار وں _ کو _اعتماد _ کرنا _چاہئے _ کہ _وہ _جو _فیڈ _استعمال _ کر _رہے _ھیں _اس _میں _وھی _چیز _ہے _جو _لیبل _پر _بیان _ کی _ گئی _ہے _۔
_بہر _حال _، _مالی _پالیسی _ کو _ایک _پائیدار _راست ے _پر _ر قرار _رکھنا _چاہئے _، _اور _بحران _کے _حل _ کی _توقع ات _ کو _ختم _ کرنا _ھوگا _۔

Table 6 shows example sentences from the source training set after subword tokenization. Table 7 display example sentences from the target training set after subword tokenization. Subword tokenization allows Transformer models to better generalize to rare or unknown words by splitting them into meaningful pieces.

Table 7: Target sentence from train set after subword tokenization

Target sentence after subword Tokenization
_ It ' s _ not _ easy _to _be _honest _all _the _time _.
_it _is _the _family _that _decide s _, _in _the _main _, _whe ther _girls _can _be _educat ed _.
_Farm ers _must _be _able _to _trust _that _the _feed _they _are _us ing _contain s _what _is _state d _on _the _label _.
_Never the less _, _fiscal _policy _should _be _maintain ed _on _a _sustainabl e _course _, _anchor ing _expectation s _of _an _order ed _resolution _of _the _crisis _.

This also helps address the vocabulary limitations of traditional word-level models and improves translation of low-resource, morphologically rich Urdu language.

4.3 Trained NMT Models

We train three different transformer model. 1: Transformer with word-level tokenization. 2: Transformer with Attention-based dropout Layer and word-level tokenization. 3. Transformer with subword tokenization. Careful configuration and training of the model is a pivotal step. correct parameters increases the model's ability to generalize. To facilitate effective learning from the data, the following parameters are used to train the model.

- **Encoder Type:** The transformer architecture is chosen as the encoder. It is effective at capturing long-term relationships in sequential data.
- **Decoder Type:** To maintain a consistent modeling approach throughout the system a decoder was is implemented using the transformer architecture.
- **Position Encoding:** positional encoding (PE) technique is applied during training. It help the model to understand the order of words in the input sequences.
- **Encoder Layers:** The encoder consists of six layers. It capture complex linguistic patterns in the input data.
- **Decoder Layers:** Similarly, the decoder comprises of six layers. This design facilitate the generation of accurate and coherent translations.
- **Multi Heads attention:** To capture diverse semantic relationships multi-headed attention mechanisms is used.
- **Hidden Size:** To capture and represent complex linguistic features, the hidden size is set to 512.
- **Word Vector Size:** The word vector size is set to 512.
- **Transformer Feed-Forward Size:** The size of the feed-forward network within the transformer is set to 2048. This allows the model to capture and process non-linear relationships between words effectively.
- **Dropout:** To prevent overfitting, a dropout rate of 0.1 is applied during training. It improved the model's generalization capabilities.

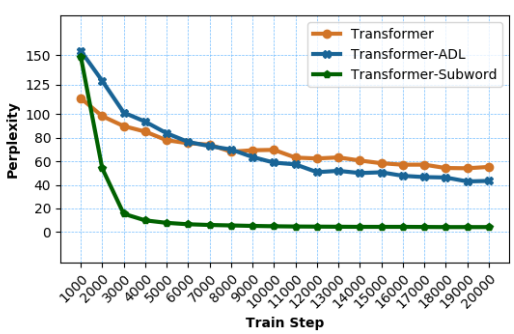
Additionally, Adam and adam_beta2 are used as optimizers. The value for adam_beta2 is set to 0.998. The initial learning rate was configured as 2. The Noam decay method is applied during training. By carefully selecting and configuring these parameters, the model is trained to achieve optimal translation performance. All these parameter are selected on the bases of prior research and empirical evidence. These are aligned with best practices in the field of machine translation.

4.4 Evaluation Methods

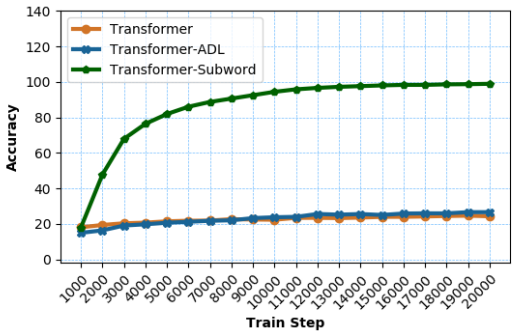
We used different automatic metrics to check how well the model worked. These matrices evaluate the quality of the translation produced by the model. These includes BLEU [44], perplexity (PPL) [45], accuracy, METEOR [46], and unigram F1. BLEU was chosen as the primary measure to assess the translation quality. PPL helped us evaluate how well the model generates coherent and contextually suitable translations. In this study, we used F1, precision, and recall to determine the overall accuracy of the model. In addition, we performed a subjective analysis to examine the quality of the translations generated by the models.

5 Result & Discussion

This section evaluates how well the "Transformer with subword tokenization" model works for UR-EN translation. The aim is to determine how well this model performs compared to two other models. It compares the proposed model to Transformer (TRF) with word-level tokenization and Transformer with attention-based dropout layer (TRF-ADL) [23]. The TRF-ADL uses an additional dropout layer on the top of the encoder. The proposed model is primarily assessed for the Urdu-English translation task.



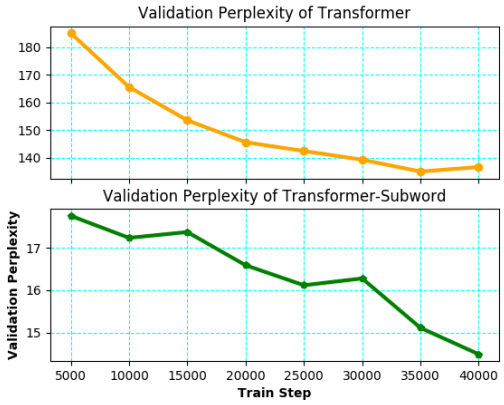
(a) Training Accuracy of TRF, TRF-ADL, and TRF-subword model



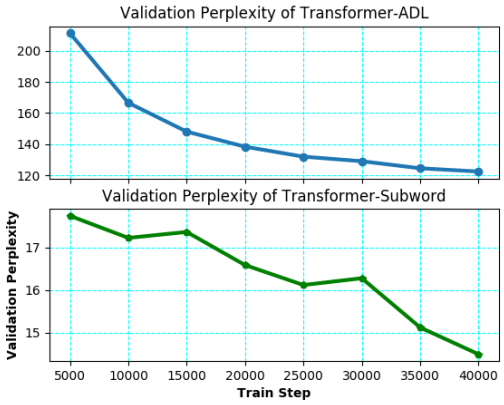
(b) Training Perplexity of TRF, TRF-ADL, and TRF-subword model

Figure 3: TRF Vs. TRF-ADL Vs. TRF-subword

Figure 3 (a) illustrates the perplexity attained during the training phase as the model optimized its parameters. To monitor how effectively the model is learning their hyperparameters and weights from the training examples, and fitting to the training data, training accuracy is recorded at each step. Figure 3 (b) shows the training accuracy attained at successive training steps.



(a) TRF Vs. TRF-subword model

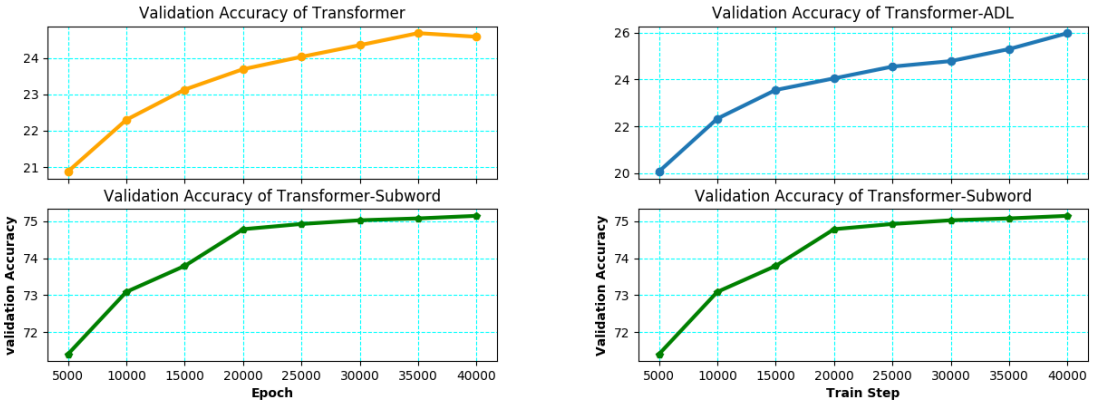


(b) TRF-ADL Vs. TRF-subword model

Figure 4: Validation Perplexity

Additionally, Figure 4 and Figure 5 depict the perplexity and accuracy achieved on the held-out validation set at successive stages of training. As training progressed through 40,000 steps, the TRF-subword model attained a significantly lower validation perplexity compared to the TRF and TRF-ADL models.

As shown in Figure 5, the perplexity of the TRF and TRF-ADL models started rising at this point in training. Moreover, The validation perplexity of TRF-subword was notably lower than the perplexity obtained by TRF and TRF-ADL, subsequently.



(a) TRF Vs. TRF-Subword model **(b)** TRF-ADL and TRF-subword Model

Figure 5: Validation Accuracy

By the final 40,000 training steps, the TRF-subword model achieved a much more stable validation accuracy while the accuracy of the TRF and TRF-ADL models started decreasing. As shown in Figure 5, The TRF-subword demonstrated a considerably higher validation accuracy than TRF and TRF-ADL throughout training.

Table 8: Model Perplexity and Accuracy on last training step

Sentence	Metric	Transformer	Transformer-ADL	Transformer-subwrod
Train Set	Accuracy	26.79	29.48	99.6 ↑
	Perplexity	41.88	31.52	3.97 ↓
Validation Set	Accuracy	24.58	24.78	75.14 ↑
	Perplexity	136.64	135.83	17.75 ↓

Table 8 presents the training and validation accuracy as well as the perplexity achieved at the final training step. On the development set, the transformer-subword model achieved the best accuracy of 75.14%, while TRF and TRF-ADL achieved 24.58% and 24.78% respectively. For the hold-out validation data set, the perplexity of the transformer-subword was 17.5, compared to 136.64 and 135.83 for TRF and TRF-ADL respectively.

Table 9 outlines the predication of proposed model's and the achieved scores on the test data. As shown in Table 9, the proposed transformer model with subword tokenization achieved better performance metrics on the test set compared to the baseline models. It demonstrated improved BLEU, GLEU, ROUGE_L, and METEOR scores as well as higher Precision and Recall, along with lower TER and WER scores. The statistics on the test set demonstrate that utilizing subword tokenization enhanced the transformer model's ability to generate more accurate predictions for unseen examples.

The proposed model shows impressive results. When compared to other NMT models trained on the same UR-EN dataset, this model performed much better. Table 10 summarizes the result of different NMT model on UR-EN data set. The BLEU scores for CNN, GRU, and SRU models are 21.80, 00.77, and 28.61, respectively. The proposed model also outperforms LSTM-ADL, CNN-ADL, GRU-ADL, and SRU-ADL models[2], which have BLEU scores of 44.69, 21.54, 44.8, and 32.4. The BLEU score for the Transformer-subword model is 45.5.

Translation quality evaluation was performed using three example sentences. The generated translations produced by the model and their respective ngram-BLEU score are shown in Table 11, Table 12, and Table 13. Examining these examples provides insight into the effectiveness of the proposed MT model.

[2] https://github.com/Huma-Israr/NMT-with-Attention-based-Dropout-Layer

Table 9: Performance statistics of trained NMT models on test set.

Result on Test-set	TRF-subword	TRF	TRF-ADL
BLEU Score	45.5 ↑	1.27	2.02
GLUE-Corpus	0.470 ↑	0.039	0.40
GLUE-sentence average Score	0.485 ↑	0.049	0.059
ROUGE_L Score	0.669 ↑	0.156	0.181
METEOR	0.700 ↑	0.048	0.060
TER	0.420 ↓	1.005	1.257
WER	10.00 ↓	24.00	31.00
Precision:	0.770 ↑	0.172	0.138
Recall:	0.798 ↑	0.107	0.141
f1:	0.784 ↑	0.132	0.139
fMean:	0.793 ↑	0.114	0.140
Bleu_1:	73.04 ↑	12.80	16.20
Bleu_2:	61.30 ↑	4.20	06.60
Bleu_3:	52.52 ↑	2.10	03.50
Bleu_4:	45.48 ↑	1.27	02.20

Table 10: Comparison of Transfomer-subword with other NMT models trained on same UR-EN dataset

Refrence	Model Trained	BLEU Score
Israr et al., [23]	CNN	21.80
	GRU	00.77
	SRU	28.61
	LSTM-ADL	44.69
	CNN-ADL	21.54
	GRU-ADL	44.80
	SRU-ADL	32.40
This Research	Transformer	01.27
	Transformer-ADL	02.02
Proposed model	Transformer-subwording	45.50

Table 11: Sentence 1 Translation

SENT:1	حکومت نے ملک میں لانگ مارچ روکنے کے لئے ہزاروں پولیس کارکنوں کو تعینات کیا ۔	BLEU
GOLD: 1	“ The Government deployed thousands of Police workers to stop long march in the country . ”	
Transformer-subword	“ Government deployed thousand of Police workers to stop killing in the country . ”	0.456
Transformer	“ I would like to thank you for your support . ”	0.00
Transformer-ADL	“ It is not enough to interfere in society . ”	0.00
Google Translator	“ The government deployed thousands of police workers to stop the long march in the country . ”	0.466
Bing Translator	“ The government deployed thousands of police personnel to stop the long march in the country . ”	0.426

For the example sentence No. 01 shown in the Table 11 translation produced by the TRF-subword, TRF and TRF-ADL model has n-gram BLEU scores 0.456, 0.00, and 0.00. The BLEU_1 score for the translation produced by TRF and TRF-ADL model is 0.121 and 0.171, respectively. The translation produced by the TRF is “I would like to thank you for your support .” The source sentence No. 01 has been incorrectly translated by TRF model. In comparison to the Gold reference, this target translation is completely unrelated and does not translate any part of the source sentence. The quality of TRF-ADL’s translation is also poor, as it fails to translate the source sentence correctly. The TRF-subwrod model produced the correct translation for much of the source sentence. Though, the model has changed the key action being stopped from “long march” to “killing”. The BLEU score assigned to this translation is 0.456. The higher BLEU score indicated that the translation produced by TF-subword is more similar to a human reference translation. In this case, our model’s output achieved a reasonably high BLEU score. We have also compared the translation result with Google Translator and Microsoft Bing Translator.

For source sentence No. 02, translation produced by TRF, TRF-ADL, and TRF-subwrod model is shown in the Table 12. TRF-subword, TRF and TRF-ADL model has achieved n-gram BLEU scores 0.467, 0.00, and

Table 12: Sentence 2 Translation

SENT: 2	کیا آپ تھوڑا سا اونچی آواز میں بول سکتے ہیں ؟	BLEU
GOLD :2	“ Can you speak a little louder ? ”	
Transformer-subword	“ Can you speak a little louder , please ? ”	0.467
Transformer	“ I am sorry . ”	0.00
Transformer-ADL	“ Why ? ”	0.00
Google Translator	“ Can you speak a little louder ? ”	1.00
Bing Translator	“ Can you speak a little louder ? ”	1.00

0.00. The BLEU_1 score for the same sentence is 0.667, 0.00 and 0.041, respectively. For the example, sentence No. 02 translation produced by TRF-subword is very similar to gold reference, with the addition of the word “please”. This is a minor augmentation that does not change the core meaning or accuracy of the translation. The term “please” adds politeness to the sentence. It is an important aspect, as Urdu often uses polite markers to make requests or ask questions.

For the same source sentence, the translation produced by TRF and TRF-ADL bears no resemblance to the gold reference. It is apparent that the outputs from the two models do not convey any aspect of the source sentence.

Table 13: Sentence 3 Translation

SENT: 2	کلاسیکی معاشیات میں ، یہ خیال کیا جاتا ہے کہ معیشت بنیادی طور پر دولت کے علم کے بارے میں ہے ۔	BLEU
GOLD :2	“ In classical economics , it is believed that the economy is primarily about the knowledge of wealth . ”	
Transformer-subword	“ In classical economics , it is believed that the economy is radically wealth knowledge of knowledge . ”	0.637
Transformer	“ I would like to congratulate the rapporteur . ”	0.00
Transformer-ADL	“ It is also important for us to see the principles . ”	0.00
Google Translator	“ In classical economics, it is believed that the economy is primarily about the knowledge of wealth . ”	0.800
Bing Translator	“ In classical economics, it is believed that economics is primarily about knowledge of wealth .”	0.363

For sentence No. 03, as shown in the Table 13 translation produced by the TRF-subword, TRF and TRF-ADL model has n-gram BLEU scores 0.637, 0.00, and 0.00. while the BLEU_1 Score is 0.83, 0.07, 0.144, respectively. For the source sentence No. 03, TRF-subword has rearranged some words in output translation but keeps the overall meaning. though “radically” is an inaccurate substitution for “primarily”. The concepts is still conveyed adequately. The translation produced by TRf and TRF-ADL bears no resemblance to the original meaning or concepts in the source sentence. While a few words like “is” and “principles” appear in the output translation produced by TRF-ADL, still it fails to convey the overall meaning.

To further check the applicability and benefits of employing subword tokenization two additional datasets, Arabic-English and Persian-English are selected. The writing systems of Arabic, Persian and Urdu share a common script [47]. These languages exhibit a right-to-left (RTL) writing style and share similar orthographic characteristics [48]. This Arabic-English and Persian-English dataset was used by Israr et al [23]. The Arabic-English dataset is from a Kaggle competition. It has 25,132 sentences in the training set. The validation set contains 5,000 Arabic-English sentence pairs, and the test set has 5,001.

Table 14: Comparison of Transfomer-subword with other NMT models trained on same on Pr-EN dataset

Refrence	Dataset	Model Trained	BLEU Score
Israr et al., [23]	Pr-EN	CNN	13.62
	Pr-EN	GRU	16.47
	Pr-EN	CNN-ADL	14.11
This Research	Pr-EN	Transformer	2.08
	Pr-EN	Transformer-ADL	3.62
Proposed model	Pr-EN	Transformer-subwording	17.42

The Persian-English dataset is sourced from Tatoeba. It consists of a small collection of open movie subtitles [49]. It includes 25,000 sentences in the training set. Both the validation and test sets have 5,000 Persian-English sentences each.

Table 15: Comparison of Transfomer-subword with other NMT models trained on same AR-EN dataset

Refrence	Dataset	Model Trained	BLEU Score
Israr et al., [23]	AR-EN	CNN	11.57
	AR-EN	GRU	13.95
	AR-EN	CNN-ADL	13.06
This Research	AR-EN	Transformer	3.43
	AR-EN	Transformer-ADL	5.71
Proposed model	AR-EN	Transformer-subwording	18.01

Table 14 and Table 15 summarizes the results of the proposed technique on the Pr-EN and AR-EN test set, respectively. A comparison of the results reveals that the proposed model achieved noticeable improvements. For 5,000 Persian sentences in test set translated into English by transformer with subword tokenization showed +15.34 BLEU points improvement. For 5,001 Arabic sentences translated into English by transformer with subword tokenization showed +14.58 BLEU points improvement. The n-gram BLEU score shows that the transformer with subword tokenization is superior to other two model in performance. Using subword tokenization contributed to the perplexity, accuracy, and BLEU score for Arabic and Persian translation to English. Table 14 and Table 15 shows the results of other NMT models trained on the same Pr-EN and AR-EN datasets. The Transomer-subword model performs even better than the CNN, CNN-ADL, and GRU models.

In summary, the analysis of the test set results shows that subword tokenization improves the transformer model's predictions. This technique breaks down words into smaller parts. It helped the model to perform better on new unseen data. Furthermore, subword tokenization also enhanced the model's ability to handle data with right-to-left (RTL) writing style effectively.

6 Conclusion & Future Work

The research explores the use of the transformer models for Urdu-to-English MT. The primary aim of the study was to look into different ways to use transformer for languages with complex morphology and writing style. For this purpose, The present study used subword tokenization. The empirical tests and comparisons with the rest of the models proved that the transformer with subword tokenization improved the translation quality. The transformer with subword tokenization achieves a high BLEU score of 45.84. This score represents a significant improvement over previous results for the same language pair. The study used the OpenNMT toolkit for experimentation, emphasizing the importance of accessible resources in advancing NMT research. The success of the research implies that transformers are effective in capturing complex language patterns and can improve MT for low-resource languages. This feature makes it one of the most feasible candidates for deployment in real-life applications on translation services. Apart from this, The research also highlights the significance of addressing language-specific challenges and suggests further work in training and fine-tuning language models.

While the Transformer model with subword tokenization achieved performance over other two models, there remains room for improvement. One major drawback of the model is that the translations it produces are less accurate than Bi-RNN and LSTM. The BLEU scores for these two models on the same UR-EN dataset are 49.67 and 47.14, respectively [23]. In this study, our primary research focus was on the UR-EN MT task. Future work will aim to further improve the performance of transformer for UR-EN MT task. Furthermore, identifying optimal hyper-parameters or discovering enhancements to the transformer architecture itself for UR-EN remains an important area for future work. Moreover, there's potential to investigate transformer models alongside other state-of-the-art NMT models by applying them to other morphologically rich, under-resourced languages.

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

References

[1] M Paul Lewis, Gary F Simons, and Charles D Fennig. Ethnologue: Languages of the world, 2015.

[2] Sadaf Abdul Rauf, Syeda Abida, Syeda Zahra, Dania Parvez, Javeria Bashir, et al. On the exploration of english to urdu machine translation. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 285–293, 2020.

[3] Ali Daud, Wahab Khan, and Dunren Che. Urdu language processing: a survey. *Artif. Intell. Rev.*, 47(3):279–311, 2017.

[4] Thierry Poibeau. *Machine translation*. MIT Press, 2017.

[5] John Hutchins. Machine translation: History and general principles. *The encyclopedia of languages and linguistics*, 5:2322–2332, 1994.

[6] Jonathan Slocum. A survey of machine translation: Its history, current status, and future prospects. *Comput. Linguistics*, 11(1):1–17, 1985.

[7] Paisarn Charoenpornsawat, Virach Sornlertlamvanich, and Thatsanee Charoenporn. Improving translation quality of rule-based machine translation. In *COLING-02: machine translation in Asia*, 2002.

[8] Adam Lopez. Statistical machine translation. *ACM Comput. Surv.*, 40(3):8:1–8:49, 2008.

[9] Harold L. Somers. Review article: Example-based machine translation. *Mach. Transl.*, 14(2):113–157, 1999.

[10] Richard Zens, Franz Josef Och, and Hermann Ney. Phrase-based statistical machine translation. In Matthias Jarke, Jana Koehler, and Gerhard Lakemeyer, editors, *KI 2002: Advances in Artificial Intelligence, 25th Annual German Conference on AI, KI 2002, Aachen, Germany, September 16-20, 2002, Proceedings*, volume 2479 of *Lecture Notes in Computer Science*, pages 18–32. Springer, 2002.

[11] Felix Stahlberg. Neural machine translation: A review. *J. Artif. Intell. Res.*, 69:343–418, 2020.

[12] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. *CoRR*, abs/1705.03122, 2017.

[13] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[15] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[17] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[18] Shiwen Ni, Min Yang, Ruifeng Xu, Chengming Li, and Xiping Hu. Layer-wise regularized dropout for neural language models. In Nicoletta Calzolari, Min-Yen Kan, Véronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 10208–10218. ELRA and ICCL, 2024.

[19] Syed Abdul Basit Andrabi, Abdul Wahid, et al. Machine translation system using deep learning for english to urdu. *Computational intelligence and neuroscience*, 2022, 2022.

[20] Shah Nawaz Khan and Imran Usman. Amodel for english to urdu and hindi machine translation system using translation rules and artificial neural network. *Int. Arab J. Inf. Technol.*, 16(1):125–131, 2019.

[21] Ahmed Khan and Aaliya Sarfaraz. Rnn-lstm-gru based language transformation. *Soft Computing*, 23(24):13007–13024, 2019.

[22] Muhammad Naeem, Abu Bakar Siddique, Raja Hashim Ali, Usama Arshad, Zain ul Abideen, Talha Ali Khan, Muhammad Huzaifa Shah, Ali Zeeshan Ijaz, and Nisar Ali. Performance evaluation of popular deep neural networks for neural machine translation. In *2023 International Conference on Frontiers of Information Technology (FIT)*, pages 220–225. IEEE, 2023.

[23] Huma Israr, Safdar Abbas Khan, Muhammad Ali Tahir, Muhammad Khuram Shahzad, Muneer Ahmad, and Jasni Mohamad Zain. Neural machine translation models with attention-based dropout layer. *Computers, Materials & Continua*, 75(2):2981–3009, 2023.

[24] Surafel Melaku Lakew, Mauro Cettolo, and Marcello Federico. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 641–652. Association for Computational Linguistics, 2018.

[25] Karim Ahmed, Nitish Shirish Keskar, and Richard Socher. Weighted transformer network for machine translation. *CoRR*, abs/1711.02132, 2017.
[26] Mya Ei San, Sasiporn Usanavasin, Ye Kyaw Thu, and Manabu Okumura. A study for enhancing low-resource thai-myanmar-english neural machine translation. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 23(4):54, 2024.
[27] Farhan Dhanani and Muhammad Rafi. Attention transformer model for translation of similar languages. In Loïc Barrault, Ondrej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Yvette Graham, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno-Yepes, Philipp Koehn, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, and Matteo Negri, editors, *Proceedings of the Fifth Conference on Machine Translation, WMT@EMNLP 2020, Online, November 19-20, 2020*, pages 387–392. Association for Computational Linguistics, 2020.
[28] Emmanuel Agyei, Xiaoling Zhang, Sophyani Banaamwini Yussif, and Bless Lord Y Agbley. Akan-english: Transformer for low resource translation. In *2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pages 256–259. IEEE, 2021.
[29] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep transformer models for machine translation. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1810–1822. Association for Computational Linguistics, 2019.
[30] Changhan Wang, Kyunghyun Cho, and Jiatao Gu. Neural machine translation with byte-level subwords. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9154–9160. AAAI Press, 2020.
[31] Séamus Lankford, Haithem Afli, and Andy Way. Human evaluation of english-irish transformer-based NMT. *Inf.*, 13(7):309, 2022.
[32] Kavit Gangar, Hardik Ruparel, and Shreyas Lele. Hindi to english: Transformer-based neural machine translation. *CoRR*, abs/2309.13222, 2023.
[33] Tong Xiao, Yinqiao Li, Jingbo Zhu, Zhengtao Yu, and Tongran Liu. Sharing attention weights for fast transformer. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5292–5298. ijcai.org, 2019.
[34] Shengjie Luo, Shanda Li, Tianle Cai, Di He, Dinglan Peng, Shuxin Zheng, Guolin Ke, Liwei Wang, and Tie-Yan Liu. Stable, fast and accurate: Kernelized attention with relative positional encoding. *CoRR*, abs/2106.12566, 2021.
[35] Nikolay Banar, Walter Daelemans, and Mike Kestemont. Character-level transformer-based neural machine translation. In *NLPIR 2020: 4th International Conference on Natural Language Processing and Information Retrieval, Seoul, Republic of Korea, December 18-20, 2020*, pages 149–156. ACM, 2020.
[36] Jindrich Libovický and Alexander Fraser. Towards reasonably-sized character-level transformer NMT by finetuning subword systems. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2572–2579. Association for Computational Linguistics, 2020.
[37] Josef Jon and Ondrej Bojar. Character-level NMT and language similarity. *CoRR*, abs/2308.04398, 2023.
[38] Jiahuan Li, Yutong Shen, Shujian Huang, Xinyu Dai, and Jiajun Chen. When is char better than subword: A systematic study of segmentation algorithms for neural machine translation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 543–549. Association for Computational Linguistics, 2021.
[39] Huma Israr, Muhammad Khuram Shahzad, and Shahid Anwar. Improved urdu-english neural machine translation with a fully convolutional neural network encoder. *International Journal of Mathematical, Engineering and Management Sciences*, 9(5):1067–1088, 2024.
[40] Jinhua Zhu, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. Incorporating BERT into neural machine translation. *CoRR*, abs/2002.06823, 2020.
[41] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016.
[42] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.
[43] Guillaume Klein, Yoon Kim, Yuntian Deng, Vincent Nguyen, Jean Senellart, and Alexander M. Rush. Opennmt: Neural machine translation toolkit, 2018.
[44] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL, 2002.
[45] Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63, 1977.

[46] Satanjeev Banerjee and Alon Lavie. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss, editors, *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics, 2005.
[47] Mahinnaz Mirdehghan. Persian, urdu, and pashto: A comparative orthographic analysis. *Writing Systems Research*, 2(1):9–23, 2010.
[48] Amna Mirza and Alexandra Gottardo. The role of context in learning to read languages that use different writing systems and scripts: Urdu and english. *Languages*, 8(1):86, 2023.
[49] Jörg Tiedemann. The tatoeba translation challenge–realistic data sets for low resource and multilingual mt. *arXiv preprint arXiv:2010.06354*, 2020.

Huma Israr earned her MSc in Computer Science from the University of Peshawar and her MS in Information Technology (IT) from the Institute of Management Sciences, Peshawar. Currently, she is a Lecturer in the Department of Information Engineering Technology at National Skills University, Islamabad, and pursuing her PhD at the National University of Sciences and Technology (NUST), Islamabad. Since 2017, she has been a research scholar at the School of Electrical Engineering and Computer Science, NUST. Her research focuses on Natural Language Processing, with interests in Artificial Intelligence, Machine Learning, Speech and Image processing, Machine translation, and Conversational AI.

Novera Pervaz earned her BS in Information Technology from the University of Agriculture, Faisalabad, and her MS in Computer Science from the National University of Sciences and Technology (NUST), Islamabad. Her research focuses on Natural Language programming, with interests in Artificial Intelligence, Machine Learning, Machine Translation, and Conversational AI. She has been actively engaged in exploring the intersections of these fields, particularly in the application of transformers and neural networks.

Dr. Safdar Abbas Khan is an assistant professor at NUST School of Electrical Engineering and Computer Science. He did his PhD from the University of Paris Est, Créteil, France and masters in Mathematics with a gold medal from Quaid-i-Azam University, Islamabad, Pakistan. His research interests include mathematical modeling of real world systems, machine learning, artificial intelligence and computer vision.