applied sciences

MDPI

Article

Framework for Handling Rare Word Problems in Neural Machine Translation System Using Multi-Word Expressions

Kamal Deep Garg [1], Shashi Shekhar [2], Ajit Kumar [3], Vishal Goyal [4], Bhisham Sharma [5], Rajeswari Chengoden [6,*] and Gautam Srivastava [7,8,9]

1 Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura 140401, Punjab, India
2 Department of Computer Engineering and Applications, GLA University, Mathura 281406, India
3 Department of Computer Science, Multani Mal Modi College, Patiala 147001, Punjab, India
4 Department of Computer Science, Punjabi University, Patiala 147001, Punjab, India
5 Chitkara University School of Engineering and Technology, Chitkara University, Baddi, Himachal Pradesh 174103, India
6 School of Information Technology and Engineering, Vellore Institute of Technology, Vellore 632014, India
7 Department of Mathematics and Computer Science, Brandon University, Brandon, MB R7A 6A9, Canada
8 Research Centre for Interneural Computing, China Medical University, Taichung 40402, Taiwan
9 Department of Computer Science and Math, Lebanese American University, Beirut 1102, Lebanon
* Correspondence: rajeswari.c@vit.ac.in

Abstract: Machine Translation (MT) systems are now being improved with the use of an ongoing methodology known as Neural Machine Translation (NMT). Natural language processing (NLP) researchers have shown that NMT systems are unable to deal with out-of-vocabulary (OOV) words and multi-word expressions (MWEs) in the text. OOV terms are those that are not currently included in the vocabulary that is used by the NMT system. MWEs are phrases that consist of a minimum of two terms but are treated as a single unit. MWEs have great importance in NLP, linguistic theory, and MT systems. In this article, OOV words and MWEs are handled for the Punjabi to English NMT system. A parallel corpus for Punjabi to English containing MWEs was developed and used to train the different models of NMT. Punjabi is a low-resource language as it lacks the availability of a large parallel corpus for building various NLP tools, and this is an attempt to improve the accuracy of Punjabi in the English NMT system by using named entities and MWEs in the corpus. The developed NMT models were assessed using human evaluation through adequacy, fluency and overall rating as well as automated assessment tools such as the bilingual evaluation study (BLEU) and translation error rate (TER) score. Results show that using word embedding (WE) and MWEs corpus increased the accuracy of translation for the Punjabi to English language pair. The best BLEU score obtained was 15.45 for the small test set, 43.32 for the medium test set, and 34.5 for the large test set, respectively. The best TER rate score obtained was 57.34% for the small test set, 37.29% for the medium test set, and 53.79% for the large test set, repectively.

Keywords: neural machine translation; multi-word expressions; out of vocabulary; word embedding; technological progress; communication technologies; technology transfer; deep learning

check for updates

Citation: Garg, K.D.; Shekhar, S.; Kumar, A.; Goyal, V.; Sharma, B.; Chengoden, R.; Srivastava, G. Framework for Handling Rare Word Problems in Neural Machine Translation System Using Multi-Word Expressions. *Appl. Sci.* **2022**, *12*, 11038. https://doi.org/10.3390/app122111038

Academic Editor: Pengjie Ren

Received: 14 September 2022
Accepted: 21 October 2022
Published: 31 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Spoken languages are a tool that is used by human beings to express their thoughts. Language plays an essential role in the faster growth of society. Language is utilized in society as a means for the interchange of ideas, as well as the sharing of information and experiences. The entire world is full of different communities, and each community has its independent language. Each language has its syntax and semantics. There is a great diversity of languages that makes communication between different communities difficult. Therefore, to enable communication among the communities, translation is required.

Text is converted from one language to another through the process of translation. Warren Weaver coined the term "machine translation" in his 1949 Memorandum on Translation, however, René Descartes first used the term "universal language" in 1629, where one symbol may be used by numerous languages [1]. Despite decades of progress in this field, the most promising translation work only started in the early 1990s as a result of revolutionary work in artificial intelligence and computational linguistics. Third-generation machine translation systems or corpus-based architectures such as statistical and example-based techniques were developed through research in the 1990s. The example-based MT system uses combinations of pre-translated data examples from its database [2]. Since 1990, a number of research groups at UMIST, the University of Brussels, Grenoble University, and the Science University of Malaysia have been conducting experiments with "dialogue-based MT" systems. In these systems, the text that is going to be translated is composed or written through a collaborative process involving both a human and a machine.

A brand-new end-to-end encoder-decoder framework for machine translation was put forth in 2013 [3]. Their work might be seen as the inception of neural machine translation (NMT), a technique that uses deep learning neural networks to map between different natural languages. A linked system of nodes that is partially based on the human brain is called a neural network. These nodes are part of an information system that processes incoming data to generate output. A sequence-to-sequence neural network (Seq2Seq) is a type of neural network that analyses a source-language sentence and generates a target-language sentence in response [4,5]. Identical to how the human brain functions, neural translation systems are always hunting for the proper patterns and making independent decisions [6].

A deep learning-based neural network (NN) has two or more hidden layers in the network [7,8]. NMT employs the softmax process on the output layer of a recurrent neural network. The computational complexity of the softmax function is exponentially large, due to which the NMT system has a limited vocabulary. In the NMT framework, this triggers the issue of out-of-vocabulary (OOV) words.

The definition of multi-word expressions (MWEs) is "lexical items that: (a) can be broken down into multiple lexemes; and (b) show lexical, syntactic, semantic, pragmatic and/or statistical idiomaticity" [9]. NMT can fail to learn and remember and replicate the MWEs since, in a high-dimensional vector, it represents the entire sentence. The objectives of this paper were to examine how to deal with OOV words and MWEs in the NMT system. To achieve this objective, a baseline NMT system was trained. To handle the OOV words, byte pair encoding (BPE) and word embedding (WE) were used while training the models in the NMT.

The following is a list of the research questions that this article will address:

1. How can we create a parallel corpus for the development of the MT system?
2. What are the various techniques that may be used to remove the noisy sentences in a parallel corpus?
3. What are the various pre-processing processes that must be completed before the dataset can be used to train an NMT system?
4. How should OOV terms be handled inside the NMT system?
5. How should one assess the predictions that are produced by the NMT system?

The following is an outline of the most significant contributions made by the paper:

1. Comparison of various existing MT systems based on the technique used for development.
2. Development of Punjabi–English parallel corpus containing named entities as well as MWEs.
3. Development of a pre-processing module that includes tokenization, true casing, and the replacement of contractions from the parallel corpus.
4. Development of four different NMT models to train the Punjabi to English MT system.
5. The OOV words were handled by using pre-trained word vectors given by fastText and adding MWEs to the parallel corpus.
6. Comparison and validation of the models using human and automated evaluation.

1.1. Motivation for Work

The motivation for this work is that MT helps people exchange their views by reducing the communication gaps due to linguistic divergence. The primary focus of translation was the Punjabi–English language pair because these languages are used for communication by many people. Globally, there are 124 million Punjabi speakers and 369.7 million English speakers [10]. Based on the number of speakers, Punjabi ranks tenth, and English ranks third in the world. Moreover, English is an international language, whereas Punjabi is one of the official languages of India. Therefore, translation between this language pair is the most important for India to relate to the world and vice-versa.

1.2. Problem Statement

The NMT system is unable to handle OOV words and MWEs in translation. Up until now, no such system has existed for those deals with MWEs in the Punjabi-to-English MT system. MWEs and a named entities dataset were used in conjunction with the parallel Punjabi–English corpus in order to resolve these concerns. The various NMT models were trained using the combined dataset. This method enhances the MT system's overall accuracy while translating from Punjabi to English.

1.3. Organization of Paper

The rest of this paper is organized are as follows. Section 2 outlines the contribution of researchers in the area of NMT and how to handle OOV words and MWEs in the NMT system is discussed. Section 3 discusses the architecture of the sequence-to-sequence NMT model, byte pair encoding (BPE), word embedding (WE), and MWEs of Punjabi. The corpus preparation and pre-processing of the corpus are discussed in Section 4. Section 5 discusses the different proposed NMT models and evaluations by using human and automated tools. Section 6 of the paper provides the conclusions and further work.

2. Related Work

These days, neural network-based MT is an ongoing trend. Various papers related to NMT and handling OOV words and MWEs in NMT are discussed in this section. In Section 2.1, papers related to handling attention, the OOV problem, and BPE in NMT are discussed, whereas Section 2.2 discusses the research related to handling MWEs in NMT.

2.1. Based on Handling Attention, OOV Problem, and BPE in NMT

Syed Abdul Basit Andrabi and Abdul Wahid (2022) developed an NMT system for English-to-Urdu pairs [11]. The parallel corpus of 30,923 was used to develop the model and the BLEU score claimed was 45.83.

Xiaoda Zhao and Xiaoyan Jin (2022) developed an English-to-Chinese attention-based NMT system [12]. A total of 220,000 parallel corpora of Chinese–English was extracted from the 2019 International Oral English and Translation Competition. The best BLEU score claimed was 22.70 with LSTM and grammar.

Levi Corallo et al. (2022) developed a German-to-English-NMT system using gated recurrent units (GRUs) [13]. The dataset containing 400,000 German–English parallel sentences was downloaded from WMT21 (Workshop on Machine Translation). Two models were trained using GRUs with 128 and 512 units. Model 1 attained a validation accuracy of 0.653 and Model 2 attained 0.649.

Sahinur Rahman Laskar et al. (2021) developed the Assamese to Bengali bidirectional NMT system [14]. A parallel corpus of approximately 150,000 Assamese–Bengali sentences were used to develop two models. The BLEU score claimed 7.20 for Assamese to Bengali and 10.10 for the Bengali to Assamese NMT systems.

Sahinur Rahman Laskar (2021) developed a Tamil-to-Telugu bidirectional MT system using a neural network [15]. The parallel corpus of 43,143 sentences was downloaded from the WMT21. Word embedding was used with a transformer to train both models. Both models claimed a BLEU score of 4.05.

Minni Jain et al. (2020) developed a neural machine translation system for Tamil to English [16]. A dataset comprising 236,427 parallel English–Tamil sentences were used to develop the two NMT models: one by using recurrent neural network (RNN) and the second by using the transformer architecture. The BLEU score claimed was 14.9 and 16.1 using RNN and transformer architecture.

Himanshu Choudhary et al. (2020) developed an NMT system for Eng-lish-to-Malayalam and English-to-Tamil [17]. The pre-trained BPE and MultiBPE encodings were used to overcome the OOV problem for both systems. The BLEU score that was claimed for the English to Tamil MT system was 24.34, while the BLEU score that was claimed for the English to Malayalam MT system was 9.78.

Fandong Meng et al. (2020) developed the Chinese-to-English NMT system [18]. The transformer and DTMT architecture based on deep transition were used to develop two models. The newest of 2019 was used to train the model. The BLEU score claimed was 38.93 and 38.66 with the transformer and DTMT architecture.

Vikrant Goyal et al. (2019) developed an attention-based NMT system for Gujarati-to-English pairs [19]. The Hindi–English parallel corpus was combined with a Gujarati–English parallel corpus to improve the baseline model and was used in training. In this way, a multi-source translation system was developed. The model achieved a BLEU score of 9.8 with this multilingual model architecture.

Charu Verma et al. (2019) developed an attention-based NMT system for Hindi-to-English language pairs [20]. The authors developed two models: one was the baseline model, and the other was the baseline model with the attention mechanism. The model was tested with 500 sentences and it was concluded that the BLEU score improved with the attention model compared to the BLEU score of the baseline model of the NMT.

Yi Mon Shwe Sin et al. (2019) developed three attention-based models: word-to-word level, character-to-word level, and syllable-to-word level [21]. The attention-based model was trained using the ALT (Asian Language Treebank) parallel corpus of 18,965 sentences and the UCSY (University of Computer Studies, Yangon) parallel corpus of 208,638 sentences. The BLEU scores claimed for all the models were 21.88, 20.71, and 26.50, respectively.

Amarnath Pathak et al. (2018) developed the SMT and NMT systems for English to Mizo [22]. Both the SMT and NMT systems were evaluated manually and by using automated tools. The authors concluded that for the low-resource language, the SMT model performed better than the NMT model.

2.2. Based on Handling MWEs in NMT

Lifeng Han et al. (2020) created a multi-lingual MWE parallel corpora for German-to-English and Chinese-to-English language pairs and used them for the development of the NMT system [23]. To extract the MWEs from the German–English parallel corpus, the following steps were followed by the researchers: (i) tagging of German and English corpus; (ii) converting tagged data into the XML format; (iii) design of the MWE pattern for German–English; (iv) extraction of monolingual MWEs with the MWE toolkit; (v) generating translation probabilities using GIZA++ and Moses; and (vi) using MPAligner to align bilingual MWEs. By using this approach, 3,159,226 and 143,042 bilingual MWEs parallel corpora were prepared for the German–English and Chinese–English language pairs. This MWEs corpus was used with the base corpus to train the NMT model. The authors claimed that MWE helped to produce a better translation, but the BLEU scores did not improve significantly.

M. Rikters et al. (2019) handled the MWEs in the NMT system for the English–Czech and English–Latvian language pairs [24]. The authors extracted parallel 400,000 MWEs for English to Czech and 60,000 for English to Latvian language pairs from a parallel corpus. The authors also extracted the parallel sentences containing MWEs from the parallel corpus. Both MWE sentences and MWE phrases were incorporated into the parallel

corpus. The author claimed that there was a significant improvement over the BLEU score by incorporating the MWE corpus in the baseline model.

Table 1 shows a review of different MT systems developed by researchers.

Table 1. Review of different machine translation systems.

MT System	Year	SMT	LSTM	MWEs	Transformer
[11]	2022	✘	✔	✘	✘
[12]	2022	✘	✔	✘	✘
✘ [13]	2022	✘	✔	✘	✘
[14]	2021	✘	✔	✘	✘
[15]	2021	✘	✔	✘	✘
[16]	2020	✘	✔	✘	✔
[17]	2020	✘	✔	✘	✘
[18]	2020	✘	✔	✘	✔
[19]	2019	✘	✔	✘	✘
[20]	2019	✘	✔	✘	✘
[21]	2019	✘	✔	✘	✘
[22]	2018	✔	✔	✘	✘
[23]	2020	✘	✔	✔	✘
[24]	2019	✘	✔	✔	✘

✔ means the technique is used in the paper, ✘ means the technique was not used in the paper.

3. The Architecture of the NMT System

The NMT system employs an architecture known as sequence-to-sequence. This architecture decodes one pattern into another pattern. The input sequence $\langle S_1, S_2, S_3, S_4 \rangle$ is words of the source language, which is then converted into another sequence $\langle T_1, T_2, T_3 \rangle$ using this architecture. A recurrent neural network (RNN) model can easily handle this sequential data [25,26]. The sequence of vectors $\langle S_1, S_2, S_3, \ldots\ldots\ldots S_n \rangle$ acts as the input for the RNN and this sequence is processed one by one. Figure 1 shows the standard RNN.

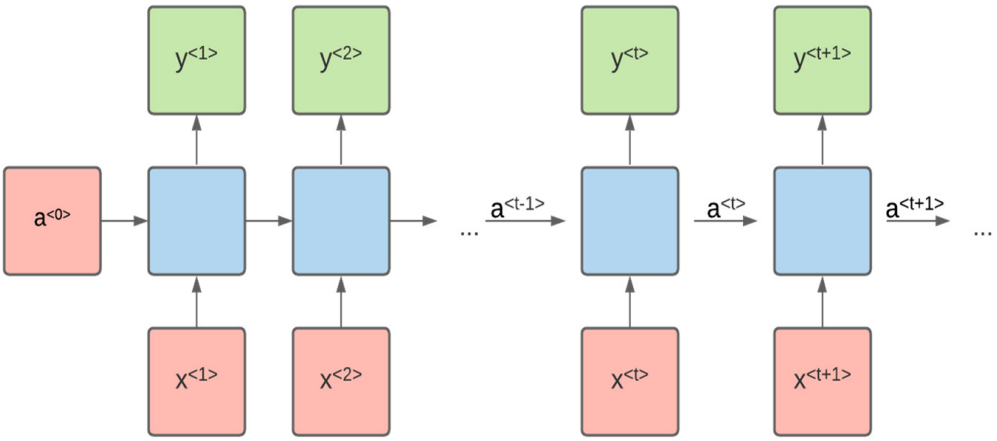


Figure 1. Recurrent softmax.

In the above figure, at time step t, $x^{<t>}$ is the input. For example, $x^{<1>}$ is the one-hot vector corresponding to the sentence's first word. For each time phase t, the activation function $a^{<t>}$ and output $y^{<t>}$ can be described as follows:

$$a^{\langle t \rangle} = g_1\left(W_{aa}a^{\langle t-1 \rangle} + W_{ax}x^{\langle t \rangle} + b_a\right) \quad (1)$$

$$y^{\langle t \rangle} = g_2\left(W_{ya}a^{\langle t \rangle} + b_y\right) \quad (2)$$

In both the above equations, W_{aa}, W_{ax}, Way, b_y are coefficients that are temporarily shared and g_1, g_2 are the activation functions. The activation function used for g_1 is tanh or relu, and the activation function used for g_2 is a softmax function for the task of MT.

RNN causes the vanishing gradient problem and exploding gradient problem in the model. The computation time is slow in RNN. It cannot consider the future input for the current state. These limitations can be handled by the long short-term memory (LSTM) model. To improve the accuracy of the system's predictions at a particular timestamp, the NMT model must consider the sequence information from both the earlier and later points in time. Bidirectional long short-term memory (BiLSTM) can be used for this purpose. The BiLSTM model that takes input from the previous and latter sequences is shown in Figure 2.

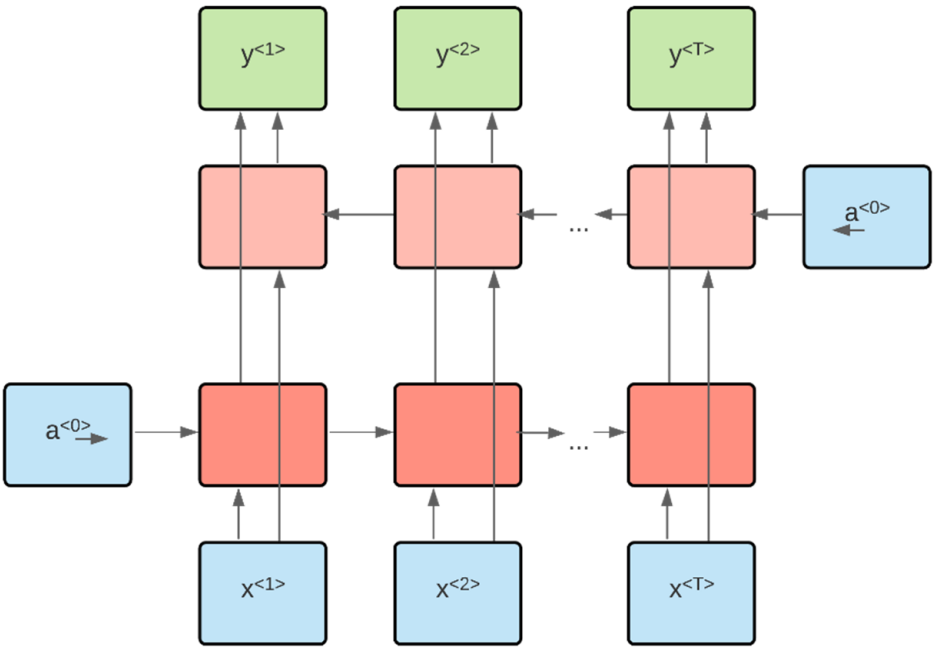


Figure 2. Bidirectional LSTM model.

3.1. Byte Pair Encoding

The NMT system has a limited vocabulary issue. To overcome this problem, researchers have used character embeddings and the subword algorithm. The subword is in between the word and the character. There are different subword algorithms: byte pair encoding (BPE), WordPiece, and SentencePiece. From these three, the BPE was selected to build a subword dictionary. BPE is a simple technique for compressing data. This substitutes a single unused byte for the most common bytes in a sequence. The algorithm to develop a subword using BPE is given below (see Algorithm 1).

Algorithm 1. BPE ()

Input: large training corpus, vocabulary size
Output: training corpus containing subword units

1. Create a large training repository.
2. Define the vocabulary size of the subword
3. Divide the word into a character series and add the suffix "</w>" to the end of the word with the word frequency. For example, if "fast" is 4 in frequency, then it is recreated as "f a s t </w>":4.
4. Create a new subword based on the occurrence's high frequency.
5. Repeat step 4 until you achieve the size of the subword vocabulary as defined in step 2, or the next highest frequency pair is 1.

The independent encodings for the Punjabi and English corpus with 35 K words were learned with the use of the BPE technique. After learning, it was applied to different sets of Punjabi and English corpus.

3.2. Word Embedding

Word embedding is a technique for learning various characteristics in which words that have the same meaning are given the same representation in a vector space that is continuous. Word embedding may be learned in a variety of different ways. Word2vec, Glove, and fastText are the three approaches that are used the most often to learn how to embed. Facebook's Artificial Intelligence Research (FAIR) team developed a neural network-based framework called fastText to categorize text based on word representations [27]. It is written in C++ and supports multiprocessing during training. The fastest supports are the CBOW and Skip Gram models. Facebook also provides a pre-trained word vector for various languages that are trained using fastText [28]. Punjabi and English pre-trained word vectors were downloaded from the fastText website and used in this research to overcome the OOV problem of NMT.

3.3. Multi-Word Expressions

MWEs play an important role in the NLP and MT. MWEs consist of two or more words but are treated as a single word [29]. Each word of the MWE has a specific meaning and it is different from the collective meaning of MWEs. Different types of MWEs exist in the Punjabi language.

3.3.1. Replicated Words

The Punjabi language has replicated words that can be treated as MWEs. The replicated word may contain a particle or hyphen sign in between. Table 2 displays the replicated word in Punjabi.

Table 2. Replicated words in Punjabi.

Punjabi Replicated Word	Translation in English
ਰੋਜ਼ ਰੋਜ਼ (“Rōz rōz”)	Every day
ਹੋਲੀ ਹੋਲੀ (“Hōlī hōlī”)	Quite slowly
ਪਾਣੀ ਹੀ ਪਾਣੀ (“Pāṇī hī pāṇī”)	Water all over

3.3.2. Waala Morpheme Construct

The term “waala” acts as a morpheme construct to create MWEs in Punjabi. “Waala” has many forms such as “waalaa”, “waalii”, “waale”, or “waalaen”. Table 3 shows the waala morpheme construct in Punjabi.

Table 3. The waala morpheme constructed in Punjabi.

Waala Morpheme Construct	Translation in English
ਕੰਮ ਵਾਲੀ (“Kam wāli”)	Maid
ਦੁੱਧ ਵਾਲਾ (“Dudha vālā”)	Milkman
ਪਾਣੀ ਵਾਲੀ ਬਾਲਟੀ (“Pāṇī vālī bālaṭī”)	Water Bucket

3.3.3. Combination of Word with Synonym, Antonym, Hyponym, or Number

MWEs are also constructed in Punjabi by combining a word with its synonym, antonym, hyponym, or number. Table 4 shows the combination of Punjabi words with their synonym, antonym, hyponym, or number.

Table 4. Combination of Punjabi words with their synonym, antonym, hyponym, or number.

Punjabi Word	Translation in English	Category
ਦਾਲ ਰੋਟੀ (“Dāl rōṭī”)	Food	Word combination with a synonym
ਦਿਨ ਰਾਤ (Din rāt)	Day and Night	Word combination with an antonym
ਪਾਣੀ ਵਾਨੀ (“Pāṇī vānī”)	Water	Word combination with hyponym
ਦਿਨੋ ਦਿਨ (“Dinō din”)	Day by Day	Word combination with a number

4. Corpus

Parallel corpora of 259,623 sentences were developed to train the NMT model. The MWEs were extracted from this parallel corpus by Kapil Dev Goyal [9,30]. Kapil Dev Goyal also collected the named entities dataset for the Punjabi–English language pair during his research work. The name entities dataset was also used in the development of the NMT model. Table 5 shows the number of Punjabi–English parallel sentences used to train the model.

Table 5. The Punjabi–English parallel corpus.

	Count of Parallel Sentences
Punjabi–English parallel corpus	259,623
Name entities dataset	558,129
MWEs extracted from a parallel corpus	89,123

4.1. Division of Dataset into Different Sets

To train and test the different NMT models, the dataset was split into different sets. The training set contained 85% of the corpus, the validation set had 5% of the corpus, and the testing set contained 10% of the parallel corpus. The division of the parallel corpus into various sets is displayed in Table 6.

Table 6. Different sets of the parallel corpus.

Waala Morpheme Construct	Translation in English
Training set	85%
Validation set	5%
Test set	10%

4.2. Pre-Processing of Corpus

The first stage in the production of the NMT method is pre-processing. The pre-processing method involves a variety of phases.

- Tokenization of the corpus

Tokenization splits a sentence on a word-by-word basis. To segment English sentences into word levels, the Moses tokenization script was used [31]. A contemporary tokenizer for Punjabi was developed using Python to break down sentences into phrases.

- Cleaning of the long sentences from the corpus

The training is often influenced by the length of the sentence. Parallel sentences with lengths of more than 50 characters were cleaned using the Moses cleaning script [32].

- Lowercasing the English corpus

Both lower-case and upper-case letters are used in English sentences. The true casing would aid in the proposed system’s increased precision. This project used the Moses casing script.

- Replacement of contractions in English corpus

A short term used in the writing of words or syllables is contraction. Cannot, we are, and other widely used contractions are only a few examples. Contractions were tested across the entire corpus. To substitute the contractions with the proper words, a Python script was written.

- Byte pair encoding to the dataset

BPE was used in the training, validation, and testing files to reduce the size of the vocabulary. Table 7 compares the scale of Punjabi and English vocabulary before and after using the 35 K merge operation to apply BPE.

Table 7. The Size of Vocabulary with the use of BPE.

Vocabulary	Number of Words in the Vocabulary
Size of English vocabulary before BPE	165,233
Size of English vocabulary after BPE	45,231
Size of Punjabi vocabulary before BPE	189,231
Size of Punjabi vocabulary after BPE	56,123

5. Experiments

There are many toolkits available to train the NMT models including Nematus [33], OpenNMT [34], Neural Monkey [35], CytonMT [36], etc. Out of all these toolkits, the OpenNMT toolkit was chosen to create the NMT models. Training the NMT model also requires a high hardware configuration. Therefore, a graphical processing unit (GPU) with 4 GB memory was used to train the different NMT models.

5.1. NMT Model Details

The OpenNMT toolkit was used to build the different NMT templates. In all models, some training parameters are fixed. All models were conditioned in batches of 64 for a total of 30 training cycles. In all versions, for the encoder, BiLSTM was used, and LSTM was used as a decoder. The beam size was set to seven during the decoding process of NMT [33]. The optimization function was the stochastic gradient descent (SGD). The attention was also used to improve the accuracy of the MT system [37,38].

The baseline NMT model 1 was trained by using the parallel corpus of 259,623 parallel sentences. Table 8 shows the configurations of different NMT models. NMT model 2 was trained by using the baseline model and WE to handle the OOV words of the Punjabi language. NMT model 3 was trained by including the MWE corpus in the baseline model. NMT model 4 contained the MWE corpus and name entities dataset corpus to the baseline model, and WE to tackle the OOV and MWEs of the source text. The overall architecture of NMT model 4 is shown in Figure 3.

Table 8. Configuration of different NMT models.

NMT Model	Configuration
Baseline model (NMT model 1)	BiLSTM as encoder LSTM as decoder BPE to reduce vocabulary size
Baseline model + word embedding (NMT model 2)	BiLSTM as encoder LSTM as decoder BPE to reduce vocabulary size WE to handle OOV words
Baseline model + MWE corpus (NMT model 3)	BiLSTM as encoder LSTM as decoder BPE to reduce vocabulary size MWEs are used in the training set

Table 8. *Cont.*

NMT Model	Configuration
Baseline model + word embedding+ MWE corpus+ name entities dataset corpus (NMT model 4)	BiLSTM as encoder LSTM as decoder BPE to reduce vocabulary size WE to handle OOV words MWEs and name entities dataset is used in the training set

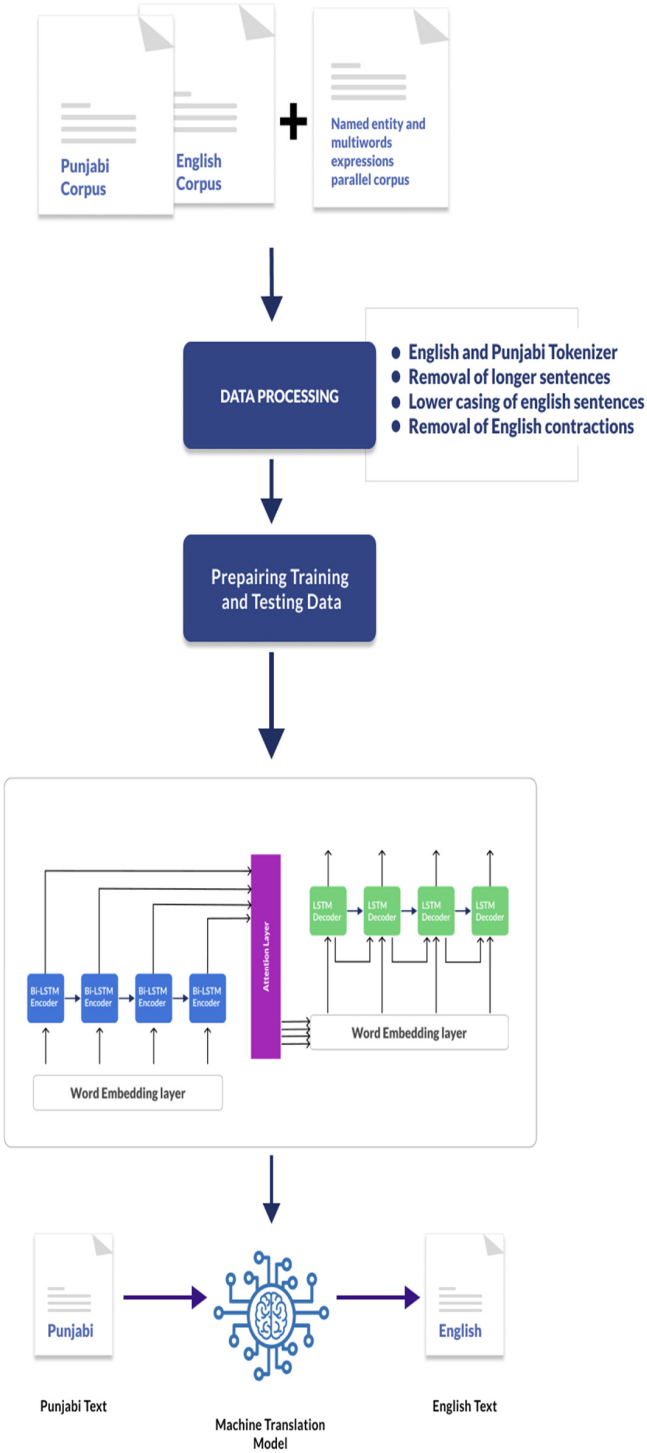


Figure 3. NMT model 4 architecture.

5.2. Evaluation of the NMT System

The performance of the MT system can be evaluated by using two methods: human assessment and automated assessment.

5.2.1. Human Assessment of MT System

Human assessment was undertaken by using linguistic experts. The linguistic expert knows both the source and target language. There are three parameters to assess the prediction of the MT system: adequacy, fluency, and overall rating. Both parameters are scored on a scale of 1 to 5, with 1 indicating mediocre performance and 5 indicating excellent performance. Figure 4 shows the scale of the adequacy, fluency, and overall rating of translation.

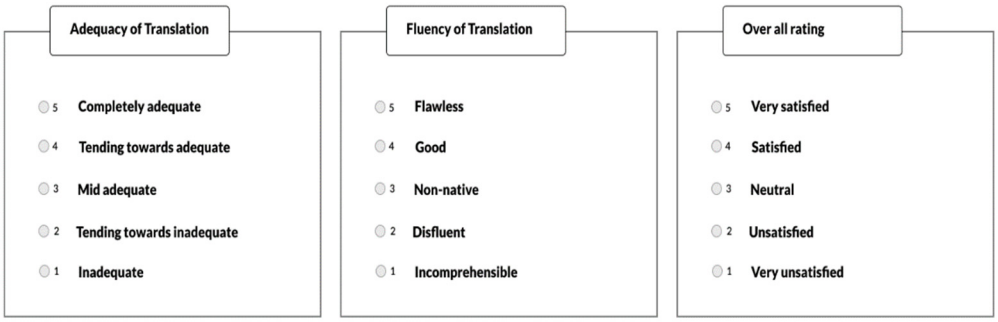


Figure 4. Human assessment scale of adequacy, fluency and overall rating.

5.2.2. Automated Assessment of MT System

When opposed to the human assessment of MT systems, the automated assessment of MT systems is more cost-effective. There are multiple tools available to automate the evaluation of MT systems such as BLEU, NIST, TER, METEOR, etc. In this research, two tools were used: BLEU and TER.

The most often used metric for the automated assessment of the MT method is the Bilingual Measurement Understudy (BLEU) [39]. The BLEU score was computed using the mean of the precision for the unigram, bigram, trigram, and 4-g, and a length penalty was also applied if the MT output sentence had a shorter length than the best matching reference translation.

$$BLEU = BP\ .exp \sum_{n=1}^{N} \frac{1}{n} log p_n \tag{3}$$

The translation error rate (TER) is an MT error metric that computes the amount of post-editing used to convert the prediction obtained from the model to the given reference translation.

$$TER = \frac{number\ of\ edits}{average\ number\ of\ referecne\ words} \tag{4}$$

The test set data was split into three sets based on the number of words in the source sentence to validate the NMT models. The small test set contained all sentences having a maximum token of five. The medium test set contained all sentences with tokens between six and fifteen. The large test set contained all sentences with tokens more than or equal to sixteen.

5.3. Human Evaluation Score of NMT Models

One linguistic expert was hired to evaluate all four NMT models. The individual score of each sentence was averaged to obtain the final score of the parameter. The human evaluation score of different models is shown in Tables 9–12.

Table 9. Human assessment of NMT model 1.

Test Set	Adequacy	Fluency	Overall Rating
Small test set	1.3	2.1	1.5
Medium test set	2.1	1.4	2.0
Large test set	1.9	1.7	1.6

Table 10. Human assessment of NMT model 2.

Test Set	Adequacy	Fluency	Overall Rating
Small test set	2.6	3.5	2.8
Medium test set	2.5	3.6	2.4
Large test set	2.4	3.1	2.5

Table 11. Human assessment of NMT model 3.

Test Set	Adequacy	Fluency	Overall Rating
Small test set	3.1	3.0	3.1
Medium test set	3.3	3.1	3.2
Large test set	3.2	3.3	3.1

Table 12. Human assessment of NMT model 4.

Test Set	Adequacy	Fluency	Overall Rating
Small test set	3.4	3.6	3.5
Medium test set	3.5	3.9	3.4
Large test set	3.2	3.8	3.3

5.4. Automated Evaluation Score of NMT Models

The BLEU and TER scores were used to evaluate all four NMT models. Table 13 shows the BLEU performance of different NMT models. The precision of the NMT model is proportional to the BLEU score. The TER scores of all four NMT models are shown in Table 14. The lower the TER%, the higher the accuracy of the NMT model.

Table 13. BLEU score of the four NMT models.

Test Set	BLEU Score (NMT Model 1)	BLEU Score (NMT Model 2)	BLEU Score (NMT Model 3)	BLEU Score (NMT Model 4)
Small test set	13.12	13.39	14.11	15.45
Medium test set	38.01	37.23	39.01	43.23
Large test set	25.12	25.26	32.5	34.5

Table 14. TER score of the four NMT models.

Test Set	TER Score (NMT Model 1)	TER Score (NMT Model 2)	BLEU Score (NMT Model 3)	BLEU Score (NMT Model 4)
Small test set	60.31%	58.12%	59.31%	57.34%
Medium test set	43.01%	40.56%	40.12%	37.29%
Large test set	58.21%	57.12%	56.45%	53.79%

5.5. Analysis of NMT Models

In this subsection, many distinct NMT models are assessed depending on the numerous parameters that are available.

5.5.1. Analysis Based on Training Time

The amount of time required to train NMT models is contingent upon a variety of hyperparameters, including the number of hidden layers, the number of neurons present in each layer, the kind of encoder and decoder, the size of the training corpus, and many more. In order to train all four models, the number of neurons in each hidden layer was increased to 500, and the number of hidden layers was increased to four. The size of the training corpus varied amongst models, which increased the amount of time needed for training as the size of the corpus grew. Figure 5 shows the training time of each NMT model. NMT model 4 took the maximum time to train the model.

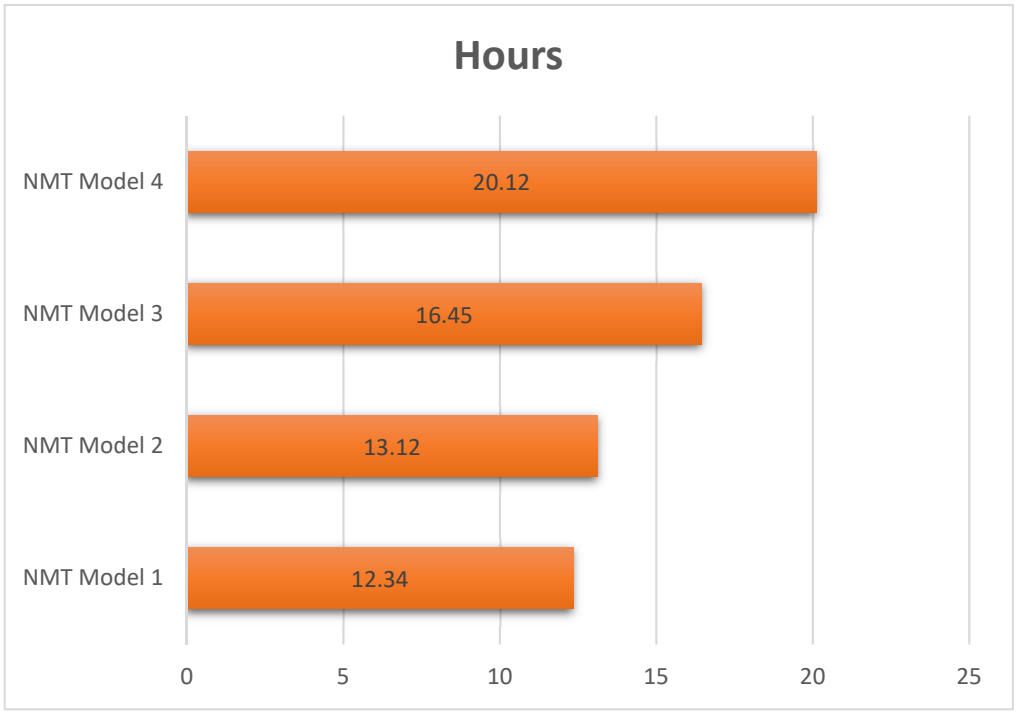


Figure 5. The training time of the NMT models.

5.5.2. Analysis Based on the Test Set

The test set was segmented into three separate sets—small, medium, and large—so that the accuracy of the various NMT models may be evaluated [40,41]. Figures 6–8 show the BLEU and TER scores of the small, medium, and large test sets. By including the WE, MWEs, and named entities dataset, there was an increase in the BLEU score. The best BLEU score was obtained for NMT model 4. For the small test sample, the NMT model 4 received a 15.45 BLEU score, 43.32 for the medium test sample, and 34.5 for the large test sample. The NMT model 4 had the lowest TER score, so it is giving a better result than all four models, as also shown in Figures 6–8.

5.5.3. Analysis Based on Length of Sentence

The accuracy of the NMT model depends on the length of the sentence. To analyze this, ten sentences were randomly picked from a test set with different lengths. The BLEU score was calculated for these sentences individually for all NMT models. Figure 9 shows the BLEU score of the sentences. It is clear from the graph that as the sentence length increased after a certain value, the BLEU score started decreasing.

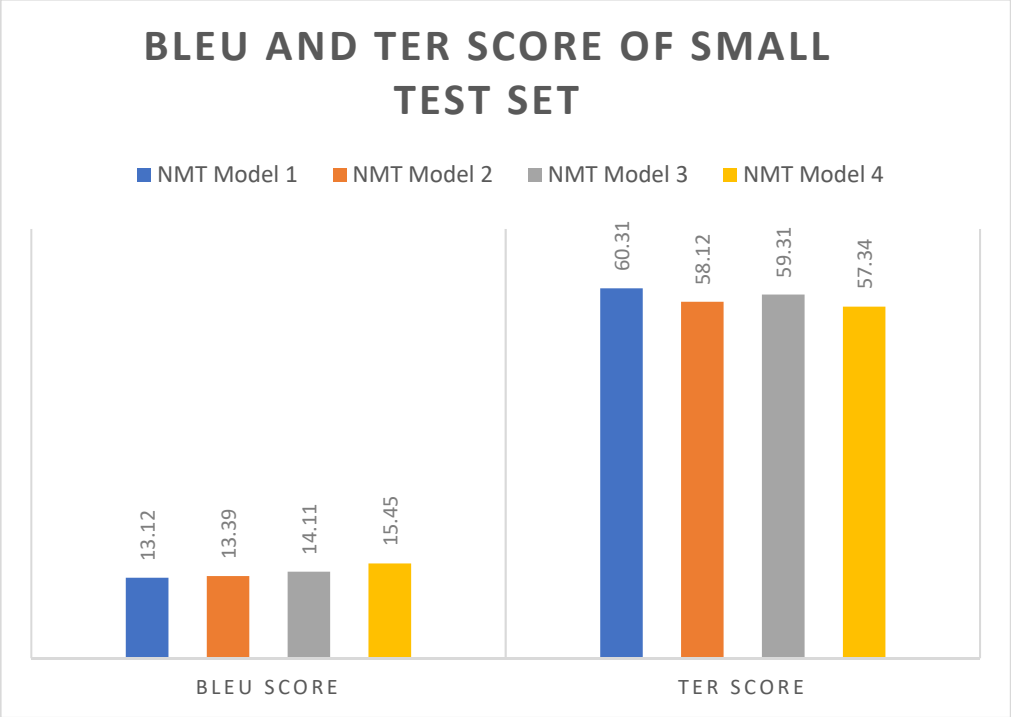


Figure 6. Scores of BLEU and TER for the small test set.

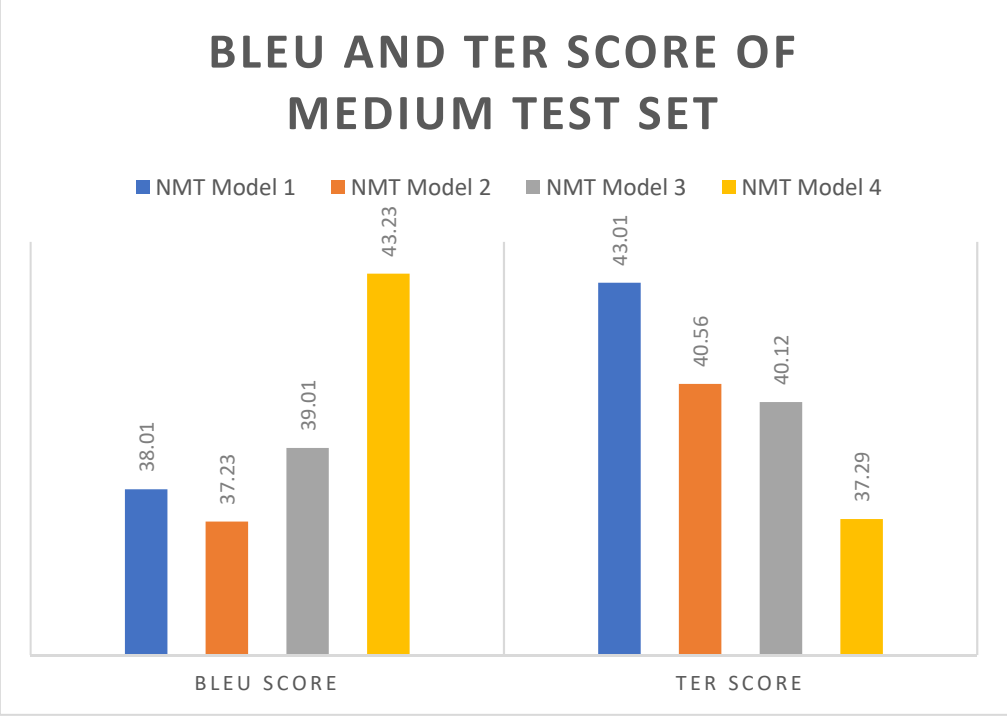


Figure 7. Scores of BLEU and TER for the medium test set.

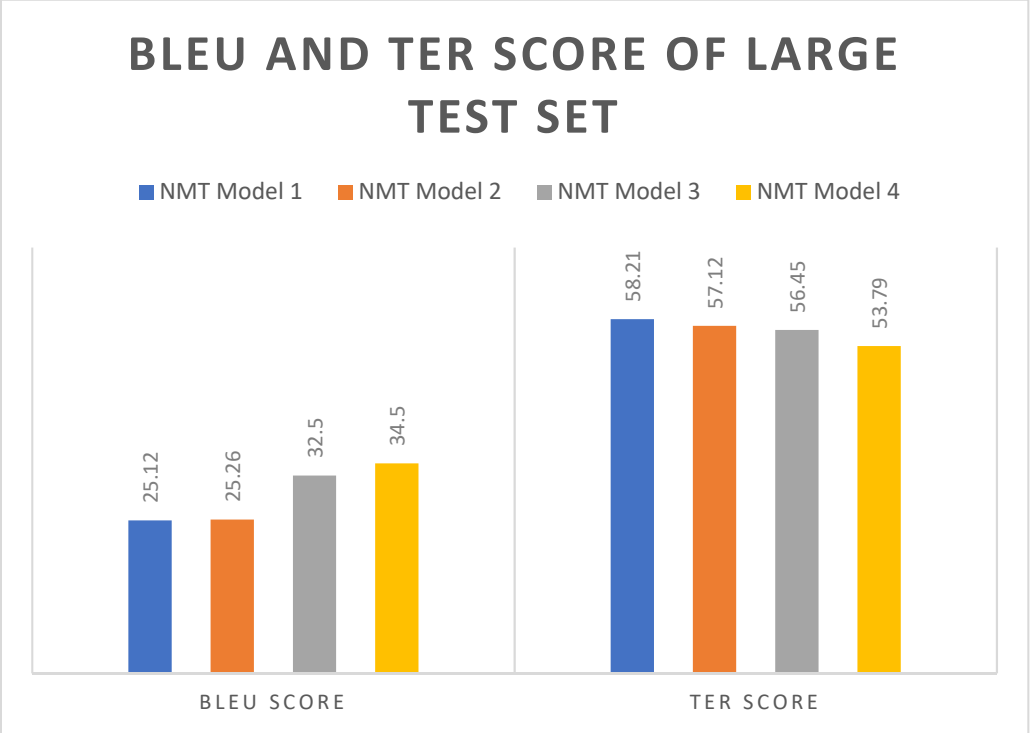


Figure 8. Scores of BLEU and TER for the large test set.

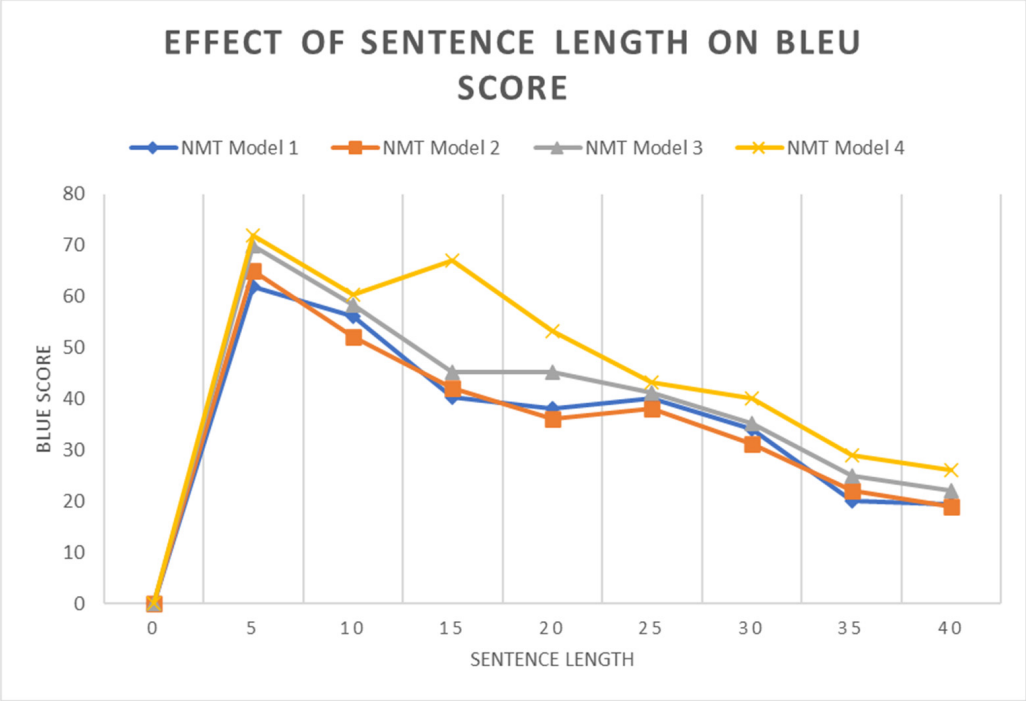


Figure 9. BLEU score of different sentences.

5.5.4. Predictions on the Sentence Level

To check the performance of NMT models, they were tested on different sentences. Now, how each model translates a sentence is shown by predictions obtained from the model.

Input text: ਸਾਡੇ ਸਰੀਰ ਵਿਚ ਸੈਂਕੜੇਂ ਅਜਿਹੇ ਜੀਵ ਰਹਿੰਦੇ ਹਨ ਜਿਹਨਾ ਨੂੰ ਮਾਈਕ੍ਰੋਸਕੋਪ ਨਾਲ ਹੀ ਦੇਖਿਆ ਜਾ ਸਕਦਾ ਹ।
Sāḍē sarīra vica saiṅkaṛēṁ ajihē jīva rahidē hana jihanā nū mā'īkrōsakōpa nāla hī dēkhi'ā jā sakadā hai.
Reference translation: hundreds of such small creatures live in our body that we can see only through a microscope.
NMT Model 1: in our $ body misconceptions remain such creatures, which can be seen with microos only.
NMT Model 2: there are such creatures in our body and cannot be seen with microos.
NMT Model 3: in our body there are many such creatures in our body that can be seen with sherbet.
NMT Model 4: in our body there are hundreds such creatures in our body that can be seen with microscope only.

In the input text, there was the word ਮਾਈਕ੍ਰੋਸਕੋਪ (mā'īkrōsakōpa). It is a named entity, and it was correctly translated by only NMT model 4. All other models translated it incorrectly. NMT model 1 showed a '$' sign in the output that was wrong. From all predictions given by different models, model 4 provided a more accurate translation for the input Punjabi text.

6. Conclusions

MT is a highlighted topic of NLP. There are various challenges in developing an accurate MT system. Out of the various challenges in the MT system are OOV words and handling MWEs. To overcome this, the NMT system was developed for the Punjabi to English system. To handle MWEs, the MWE corpora were prepared and used in the training set. Pre-trained WE for Punjabi and English was also used to handle OOV words of the target language. Different models were trained and evaluated by using human as well as automated evaluation. By using the WE, named entities, and MWE corpus, the accuracy of the Punjabi to English NMT model was improved. The limitation of the system is that it did not perform well for large sentences. Another limitation of the system is that Punjabi is a morphologically rich language and to date, the vocabulary is non-standardized, and Punjabi has various dialects that cannot be handled by this system. For example, Malawi is a dialect of Punjabi, but the system was trained on generalized Punjabi.

Future Work

In future, BERT-based different models will be trained to handle MWEs. In addition to this, a large parallel corpus of Punjabi–English will be developed and models with different hyperparameters will be trained to check the effect of various parameters on the accuracy of the NMT system. In addition, work will also be carried out to develop a multilingual NMT system with the Punjabi, English, Hindi, and Dogri languages.

Author Contributions: Data Curation, K.D.G., A.K., V.G., and R.C.; Formal Analysis, K.D.G., A.K., V.G., S.S., and B.S.; Funding Acquisition, G.S.; Investigation, S.S.; Methodology, K.D.G., S.S., R.C., and B.S.; Project Administration, K.D.G., A.K., V.G., and G.S.; Resources, B.S. and G.S.; Software, K.D.G., A.K., V.G., and S.S.; Validation, B.S.; Visualization, S.S. and G.S.; Writing—Original Draft, K.D.G., A.K., V.G., B.S., R.C., and S.S.; Writing—Review and Editing, S.S., R.C., and G.S. All authors have read and agreed to the published version of the manuscript.

Funding: Vellore Institute of Technology, Vellore.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hutchins, W.J. *Machine Translation: A Brief History*; Elsevier Science Ltd.: Amsterdam, The Netherlands, 1995. [CrossRef]
2. Somers, H. Review Article: Example-Based Machine Translation. *Mach. Transl.* **1999**, *14*, 113–157. [CrossRef]

3. Kalchbrenner, N.; Blunsom, P. Recurrent Continuous Translation Models. EMNLP 2013–2013 Conference on Empirical Methods in Natural Language Processing. *Proc. Conf.* **2013**, *1*, 1700–1709.
4. Sharma, A.; Yadav, D.P.; Garg, H.; Kumar, M.; Sharma, B.; Koundal, D. Bone Cancer Detection Using Feature Extraction Based Machine Learning Model. *Comput. Math. Methods Med.* **2021**, *2021*, 7433186. [CrossRef] [PubMed]
5. Lahoura, V.; Singh, H.; Aggarwal, A.; Sharma, B.; Mohammed, M.A.; Damaševičius, R.; Kadry, S.; Cengiz, K. Cloud Computing-Based Framework for Breast Cancer Diagnosis Using Extreme Learning Machine. *Diagnostics* **2021**, *11*, 241. [CrossRef] [PubMed]
6. Pradhan, R.; Sharma, D.K. An Ensemble Deep Learning Classifier for Sentiment Analysis on Code-Mix Hindi–English Data. *Soft Comput.* **2022**, 1–18. [CrossRef] [PubMed]
7. Yadav, D.P.; Sharma, A.; Athithan, S.; Bhola, A.; Sharma, B.; Dhaou, I.B. Hybrid SFNet Model for Bone Fracture Detection and Classification Using ML/DL. *Sensors* **2022**, *22*, 5823. [CrossRef] [PubMed]
8. Bhalla, K.; Koundal, D.; Sharma, B.; Hu, Y.C.; Zaguia, A. A Fuzzy Convolutional Neural Network for Enhancing Multi-Focus Image Fusion. *J. Vis. Commun. Image Represent.* **2022**, *84*, 103485. [CrossRef]
9. Goyal, K.D.; Goyal, V. Identification and Extraction of Multiword Expression from Indian Language: Review. *Int. J. Eng. Sci.* **2018**, *27*, 182–193.
10. Most Widely Spoken Languages in the World. Available online: https://www.infoplease.com/world/social-statistics/most-widely-spoken-languages-world (accessed on 12 June 2020).
11. Andrabi, S.A.B.; Wahid, A. Machine Translation System Using Deep Learning for English to Urdu. *Comput. Intell. Neurosci. vol.* **2022**, *2022*, 7873012. [CrossRef]
12. Zhao, X.; Jin, X. A Comparative Study of Text Genres in English-Chinese Translation Effects Based on Deep Learning LSTM. *Comput. Math. Methods Med.* **2022**, *2022*, 7068406. [CrossRef]
13. Corallo, L.; Li, G.; Reagan, K.; Saxena, A.; Varde, A.S.; Wilde, B. *A Framework for German-English Machine Translation with GRU RNN*; CEUR Workshop Proc: Aachen, Germany, 2022; p. 3135.
14. Laskar, S.R.; Pakray, P.; Bandyopadhyay, S. Neural Machine Translation: Assamese–Bengali. In *Smart Innovation, Systems and Technologies*; Springer: Singapore, 2021; Volume 206, pp. 571–579. [CrossRef]
15. Laskar, S.R.; Paul, B.; Adhikary, P.K.; Pakray, P.; Bandyopadhyay, S. Neural Ma-chine Translation for Tamil-Telugu Pair. In Proceedings of the Sixth Conference on Machine Translation (WMT), Online Event, 10–11 November 2021; pp. 284–287.
16. Jain, M.; Punia, R.; Hooda, I. Neural Machine Translation for Tamil to English. *J. Stat. Manag. Syst.* **2020**, *23*, 1251–1264. [CrossRef]
17. Choudhary, H.; Rao, S.; Rohilla, R. Neural Machine Translation for Low-Resourced Indian Languages. In Proceedings of the LREC 2020-12th International Conference on Language Resources and Evaluation, Marseille, France, 11–16 May 2020; pp. 3610–3615.
18. Meng, F.; Yan, J.; Liu, Y.; Gao, Y.; Zeng, X.; Zeng, Q.; Li, P.; Chen, M.; Zhou, J.; Liu, S.; et al. WeChat Neural Machine Translation Systems for WMT20. *arXiv* **2020**, arXiv:2010.00247.
19. Goyal, V.; Sharma, D.M. The IIIT-H Gujarati-English Machine Translation System for WMT19. In Proceedings of the Fourth Conference on Machine Translation (WMT), Florence, Italy, 1–2 August 2019; Volume 2, pp. 191–195. [CrossRef]
20. Verma, C.; Singh, A.; Seal, S.; Singh, V.; Mathur, I. Hindi-English Neural Machine Translation Using Attention Model. *Int. J. Sci. Technol. Res.* **2019**, *8*, 2710–2714.
21. Shwe Sin, Y.M.; Soe, K.M. Attention-Based Syllable Level Neural Machine Translation System for Myanmar to English Language Pair. *Int. J. Nat. Lang. Comput.* **2019**, *8*, 1–11. [CrossRef]
22. Pathak, A.; Pakray, P.; Bentham, J. English–Mizo Machine Translation Using Neural and Statistical Approaches. *Neural Comput. Appl.* **2018**, *31*, 7615–7631. [CrossRef]
23. Han, L.; Jones, G.J.F.; Smeaton, A.F. MultiMWE: Building a Multi-Lingual Multi-Word Expression (MWE) Parallel Corpora. In Proceedings of the LREC 2020-12th International Conference on Language Resources and Evaluation, Marseille, France, 11–16 May 2020; pp. 2970–2979.
24. Rikters, M.; Bojar, O. Paying Attention to Multi-Word Expressions in Neural Machine Translation. *arXiv* **2019**, arXiv:1710.06313.
25. Garg, S.; Sharma, D.K. Linguistic Features Based Framework for Automatic Fake News Detection. *Comput. Ind. Eng.* **2022**, *172*, 108432. [CrossRef]
26. Pradhan, R.; Sharma, D.K. A Framework for Topic Evolution and Tracking Their Sentiments with Time. *Int. J. Fuzzy Syst. Appl. (IJFSA)* **2022**, *11*, 1–19. [CrossRef]
27. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of Tricks for Efficient Text Classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017-Proceedings of Conference, Valencia, Spain, 3–7 April 2016; Volume 2, pp. 427–431. [CrossRef]
28. Grave, E.; Bojanowski, P.; Gupta, P.; Joulin, A.; Mikolov, T. Learning Word Vectors for 157 Languages. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018; pp. 1–5.
29. Goyal, K.D.; Goyal, V. Extraction of Replicated Punjabi Multiword Expressions. *Int. J. Eng. Sci.* **2020**, *33*, 33–45.
30. Goyal, K.D.; Goyal, V. Extraction of Named Entities from Punjabi-English Parallel Corpora. *J. Xi'an Univ. Archit. Technol.* **2020**, *12*, 639–648.
31. English Tokenizer. Available online: https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl (accessed on 25 June 2020).
32. Moses Clean Corpus Script. Available online: https://github.com/moses-smt/mosesdecoder/blob/master/scripts/training/clean-corpus-n.perl (accessed on 25 June 2020).

33. Kolonay, R.M.; Yang, H.T.Y.; Sennrich, R.; Firat, O.; Cho, K.; Birch, A.; Haddow, B.; Hitschler, J.; Junczys-Dowmunt, M.; Läubli, S.; et al. Nematus: A Toolkit for Neural Machine Translation. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Spain, 3–7 April 2017; Volume 35, pp. 65–68.
34. Klein, G.; Kim, Y.; Deng, Y.; Senellart, J.; Rush, A.M.; Crego, J.; Senellart, J.; Rush, A.M. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In Proceedings of the ACL 2017-55th Annual Meeting of the Association for Computational Linguistics, Proceedings of System Demonstrations, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 67–72. [CrossRef]
35. Helcl, J.; Libovický, J.; Kocmi, T.; Musil, T.; Cífka, O.; Variš, D.; Bojar, O. Neural Monkey: The Current State and Beyond. In Proceedings of the AMTA 2018-13th Conference of the Association for Machine Translation in the Americas, Boston, MA, USA, 17–21 March 2018; Volume 1, pp. 168–176.
36. Wang, X.; Utiyama, M.; Sumita, E. CytonMT: An Efficient Neural Machine Translation Open-Source Toolkit Implemented in C++. In Proceedings of the EMNLP 2018-Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Brussels, Belgium, 31 October–4 November 2018; pp. 133–138. [CrossRef]
37. Jia, Y. Attention Mechanism in Machine Translation. *J. Phys. Conf. Ser.* **2019**, *1314*, 012186. [CrossRef]
38. Gambhir, M.; Gupta, V. Deep Learning-Based Extractive Text Summarization with Word-Level Attention Mechanism. *Multimed. Tools Appl.* **2022**, *81*, 20829–20852. [CrossRef]
39. Zhang, Y.; Vogel, S.; Waibel, A. Interpreting BLEU/NIST Scores: How Much Improvement Do We Need to Have a Better System? In Proceedings of the 4th International Conference on Language Resources and Evaluation, Lisbon, Portugal, 24–30 May 2004; pp. 2051–2054.
40. Prabadevi, B.; Deepa, N.; Ganesan, K.; Srivastava, G. A decision model for ranking Asian Higher Education Institutes using an NLP-based text analysis approach. *ACM Trans. Asian Low-Resour. Lang. Inf. Processing* **2021**. [CrossRef]
41. Ashokkumar, P.; Shankar, S.G.; Srivastava, G.; Maddikunta, P.K.; Gadekallu, T.R. A two-stage text feature selection algorithm for improving text classification. *ACM Trans. Asian Low-Resour. Lang. Inf. Processing.* **2021**, *20*, 1–9.