



Towards achieving a delicate blending between rule-based translator and neural machine translator

Md. Adnanul Islam¹ · Md. Saidul Hoque Anik² · A. B. M. Alim Al Islam³

Received: 30 July 2020 / Accepted: 1 March 2021 / Published online: 29 March 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

Popular translators such as Google, Bing, etc., perform quite well when translating among the popular languages such as English, French, etc.; however, they make elementary mistakes when translating the low-resource languages such as Bengali, Arabic, etc. Google uses Neural Machine Translation (NMT) approach to build its multilingual translation system. Prior to NMT, Google used Statistical Machine Translation (SMT) approach. However, these approaches solely depend on the availability of a large parallel corpus of the translating language pairs. As a result, a good number of widely spoken languages such as Bengali, remain little explored in the research arena of artificial intelligence. Hence, the goal of this study is to explore improvised translation from Bengali to English. To do so, we study both the rule-based translator and the corpus-based machine translators (NMT and SMT) in isolation, and in combination with different approaches of blending between them. More specifically, first, we adopt popular corpus-based machine translators (NMT and SMT) and a rule-based machine translator for Bengali to English translation. Next, we integrate the rule-based translator with each of the corpus-based machine translators separately using different approaches. Besides, we perform rigorous experimentation over different datasets to report the best performance score for Bengali to English translation till today by revealing a comparison among the different approaches in terms of translation performance. Finally, we discuss how our different blending approaches can be re-used for other low-resource languages.

Keywords Neural machine translation (NMT) · Rule-based translator · Statistical machine translation (SMT) · Blending · Integration · Natural language processing

1 Introduction

Every human language has a vocabulary consisting of thousands of words, which are primarily built up from several dozens of speech sounds. More remarkable point here to be noted is that every normal child basically learns

the whole system (mother tongue) just from hearing others using it. However, apart from the mother tongue, other languages are generally learnt in a more systematic process. Besides, in all languages, there are many words that may have multiple meanings and also some sentences may use different grammatical structures to express the same meaning [30]. This challenge, in turn, makes it immensely difficult to perform translation between a pair of languages, which poses a major challenge in the sector of artificial intelligence. Moreover, the task of machine translation experiences the top level of difficulty when the pair of languages contain a source language that is less explored in terms of having substantially large parallel corpus [6]. Bengali represents an example of such a low-resource source language. Therefore, it remains a great challenge to do the right semantic analysis to properly recognize any sentence of such a language.

Natural languages such as English, Spanish, and even Hindi are rapidly progressing in machine translation using

✉ Md. Adnanul Islam
islamadnan2265@gmail.com; adnanul@cse.mist.ac.bd

Md. Saidul Hoque Anik
anik@cse.uui.ac.bd

A. B. M. Alim Al Islam
alim_razi@cse.buet.ac.bd

¹ Bangladesh University of Engineering and Technology/
Military Institute of Science and Technology, Dhaka,
Bangladesh

² United International University, Dhaka, Bangladesh

³ Bangladesh University of Engineering and Technology,
Dhaka, Bangladesh

artificial intelligence. While progress has been made in language translation software and allied technologies, the primary language of the ubiquitous and all-influential World Wide Web remains to be English.¹ Millions of immigrants who travel the world from non-English-speaking countries every year, face the necessity of learning English to communicate in the language, since it is very important to enter and ultimately succeed in mainstream English speaking countries. The success gets comprehended when the learning covers all forms of reading, writing, speaking, and listening that eventually realize the process of translation encompassing a diversified set of applications. However, similar to many other non-English-speaking countries, a major group of Bengali speaking people from Bangladesh and India lacks proficiency in English [33]. This crisis is getting boosted over the period of time, as there is no well-developed translator till now for Bengali to English translation. Moreover, popular translators perform well for languages that have text corpora of millions of sentences; however, they perform poorly for low-resource languages such as Bengali. Therefore, the importance of an efficient artificially intelligent translation system for low-resource languages such as Bengali to English is noteworthy.

To this context, in this article, we study machine translation for a low-resource language (e.g., Bengali to English) through exploring rule-based translation and neural (or statistical) machine translation—both in isolation and in combination, through applying different blending approaches. Besides, we discuss the implication of our blending approaches for several low-resource languages by providing concrete examples. Based on our work, our main contributions in this article are as follows:

- We integrate rule-based translator with existing NMT (and SMT) using different possible approaches. To do so, first, we implement the classical NMT, and adopt an existing Bengali to English rule-based translator. Next, we blend rule-based translator and NMT in three different ways to investigate the best-possible blending approach. Afterwards, similar to NMT, we implement SMT, and blend rule-based translator with it to verify our best-possible blending approach.
- Additionally, designing a parallel corpus containing Bengali–English sentence pairs for training NMT or SMT is one of the toughest challenges that we face, since Bengali is an extremely low-resource language. Hence, we develop three Bengali–English parallel corpora having reasonable sizes, which can enhance future research opportunities in this arena.

- Finally, we perform the performance evaluation for rule-based translator, classical NMT, and their integrated solutions using three standard metrics—BLEU, METEOR, and TER. We present the results for rule-based translator and NMT both in isolation and in combination. We also perform comparative analysis of the results among all the proposed approaches both statistically and graphically. Besides, we show performance scores for SMT and its integrated solutions with rule-based translator as an extension of our experimental results.

2 Background and related work

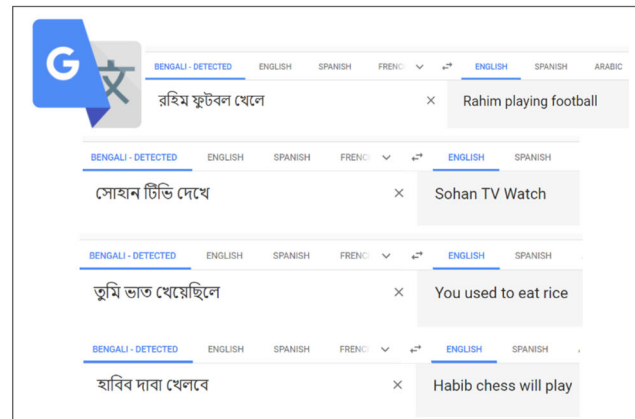
Bengali, being among the top ten languages worldwide,² lags behind in some crucial areas of research in natural language processing (NLP) such as parts-of-speech (POS) tagging [36], machine translation, text categorization and contextualization [11], syntax and semantic checking, speech to text conversion [12], etc. Most noteworthy previous studies specifically in machine translation include Example-based Machine Translation (EBMT) [33], phrase-based machine translation [9, 15], and use of syntactic chunks as translation units [17]. However, these studies lack in processing Bengali words semantically. Besides, although significant research work can be found on English to Bengali translation [4, 7, 32, 40], very few work have been performed on translating from Bengali to English [29–31]. Popular translators such as Google, Bing, Yahoo Babel Fish, etc., often perform very poorly when they translate from Bengali to other languages. Google translator, the most popular one among them, uses neural machine translation (NMT) approach with RNN at present [19, 41].

NMT (Bahdanau et al. [3]) has emerged as the most propitious machine translation technology recently, exhibiting superior performance on different public benchmarks. It is an end-to-end learning approach for automated translation, with the potential to overcome many of the weaknesses of conventional translation systems. In spite of the recent success of NMT in standard benchmarks, the lack of large parallel corpora poses a major practical problem for many language pairs such as Bengali–English [20]. This is why, NMT performs reasonably well when it translates among the most popular languages, however, it often makes elementary mistakes while translating languages that are less known to the system such as Bengali as shown in Fig. 1. Focusing on rule-based translation for such low-resource languages might be a solution.

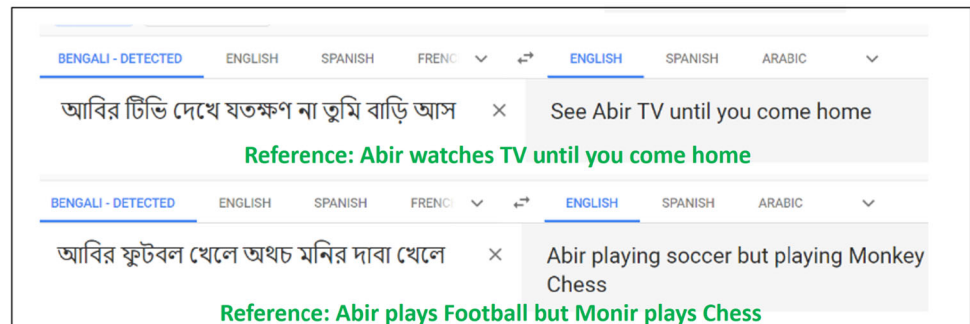
¹ <https://friedpapers.com/essay/importance-of-learning-english-essay>.

² <https://www.ethnologue.com/guides/ethnologue200>.

Fig. 1 Faulty translations of Google Translator (correct translations are appended as reference)



(a) Simple sentences



(b) Complex sentence and compound sentence

Moreover, blending NMT (and SMT) with such rule-based translator is yet an aspect to be investigated till now.

2.1 Corpus-based machine translation

Wu et al. [41] presented GNMT, Google's Neural Machine Translation system, with the objectives of reducing computational cost both in training and in translation inference, and increasing parallelism and robustness in translation. However, this approach solely relies on availability of significantly large parallel corpus and makes elementary mistakes while translating low-resource languages [16]. Sennrich et al. [37] proposed an approach for translating low-resource languages by pairing monolingual training data with an automatic back-translation to treat it as additional parallel training data. Besides, Gu et al. [10] proposed a new universal machine translation approach focusing on extremely low-resource languages. Furthermore, Artetxe et al. [2] removed the need of parallel data and proposed a novel method to train an NMT system with the objectives of relying on monolingual corpora only, and profiting from small parallel corpora. However, this promising approach still falls much behind the performance level of classical NMT. Saha et al. [35] reported an EBMT system with the objective of translating news

headlines from English to Bengali. However, this work was a specialized methodology only for newspaper headlines.

Gangadharaiyah et al. [9] converted CNF to normal parse trees using bilingual dictionary with the objective of generating templates for aligning and extracting phrase-pairs for clustering. Kim et al. [17] used syntactic chunks as translation units with the objective of properly dealing with systematic translation for insertion or deletion of words between two distant languages. However, these approaches also rely on availability of significantly large parallel corpus.

2.2 Rule-based and hybrid machine translation

Additionally, there exist several research studies focusing on Bengali language processing. For example, Bal et al. [4] proposed a solution based on parse tree, Naskar et al. [25] handled prepositions, Dasgupta et al. [7] proposed another approach based on parse tree, etc. However, these techniques consider English-to-Bengali context only, not focusing on Bengali-to-English. Rahman et al. [30] explored statistical approach for Bengali-to-English translation. Besides, Rahman et al. [31] explored a basic rule-based approach for the same. However, these techniques either depend on a large corpus or omit some basic

grammatical features. In addition to that, [1, 26] proposed hybrid translation techniques, which do not offer substantial improvement over other existing techniques.

None of the existing studies focuses on integration of rule-based translator with any corpus-based machine translator (NMT or SMT) for translation between any language pair. Moreover, an effectively large parallel corpus for Bengali-to-English machine translation is yet to be available. In this article, we focus on integration of rule-based translator with corpus-based machine translators (NMT and SMT) specifically for Bengali to English translation.

3 Architecture of NMT

Earlier, phrase-based translation systems accomplished translation tasks by splitting source sentences into several phrases, and then performing phrase-by-phrase translation. However, these approaches fail to realise the semantics of the whole (source) sentence before generating translation, and thus, fall short of accuracy and fluency in translation. With the advent of NMT, such limitations are significantly curtailed. NMT is basically an end-to-end learning approach for automated translation, which has the potential to generate cogent translations by realising long-range dependencies (e.g., subject-verb agreements, gender agreements, semantics, etc.) in source sentences [21]. NMT models generally differ in terms of their architectures. In this article, we employ the encoder–decoder architecture to generate neural machine translations. In this architecture (Fig. 2), an encoder converts the source sentence into a sequence of numbers that represents meaning of the sentence (i.e., ‘thought’ vector), and a decoder performs a translation from that vector. Usually, both the encoder and the decoder uses Recurrent Neural Network (RNN). However, RNN models vary in terms of several aspects such as number of layers (single or multi), directionality (unidirectional or bidirectional), and category (vanilla RNN, long short-term memory (LSTM), or gated recurrent unit (GRU)). We consider multi-layered (2 layers) bi-directional RNN (LSTM architecture) for both encoder and decoder. Figure 3 shows the architecture of our employed such NMT model for performing neural machine translations.

We discuss different components of our adopted NMT architecture in relevant details below.

3.1 Embedding layer

The NMT system is trained with a suitable parallel corpus as the system must fetch the corresponding word embeddings using source and target embeddings learned from

training. To accomplish so, first, the NMT model selects a vocabulary of size V for both source and target languages by taking only the most frequent V words into consideration. The remaining words are transformed into an “unknown” ($\langle \text{unk} \rangle$) token having identical embedding. Our NMT model utilizes ‘Word2Vec’ embedding technique with Skip Gram model [23].

3.2 Encoder

NMT models can use one or more LSTM layers to implement the encoder model. It outputs a fixed-sized vector that represents the internal portrayal (semantics) of the input sequence. The number of memory cells in each layer defines the length of this vector. Here, we use dynamic RNN to allow variable length sequential data processing. The encoder RNN states are initialized to zero vectors.

3.3 Decoder

The decoder converts the learned semantics of a source sequence into a target sequence. Similar to encoder, the decoder model can be implemented using single or multiple LSTM layers. Since the decoder needs to acquire information about the input sequence from the encoder, it is simply initialized to the final hidden state of the encoder. Hence, as shown in Fig. 3, the hidden state at the last input word ‘student’ is transferred to the decoder side. In our model, we use beam search decoder with $beamwidth = 10$ for translation.

3.4 Projection layer and loss

The Projection layer is a dense matrix to turn the top hidden states to logit^3 vectors of dimension equal to the vocabulary size. Next, ‘Loss’ is calculated between predicted translation and reference translation so that it can be propagated backwards for updating the weights in both encoder and decoder. Our employed NMT model uses cross entropy loss function for calculating ‘Loss’.

3.5 Gradient optimization

After calculating ‘Loss’, its derivative (i.e., gradient) needs to be calculated for updating the weights (controlled by a learning rate) during backward propagation through the

³ Logits generally refers to the unnormalized final scores of a machine learning model. In our model, logits refers to the scores of words in vocabulary to appear in the translation. We apply softmax to it to get a probability distribution over the classes.

Fig. 2 Neural machine translation (*English* → *Bengali*) using encoder–decoder architecture

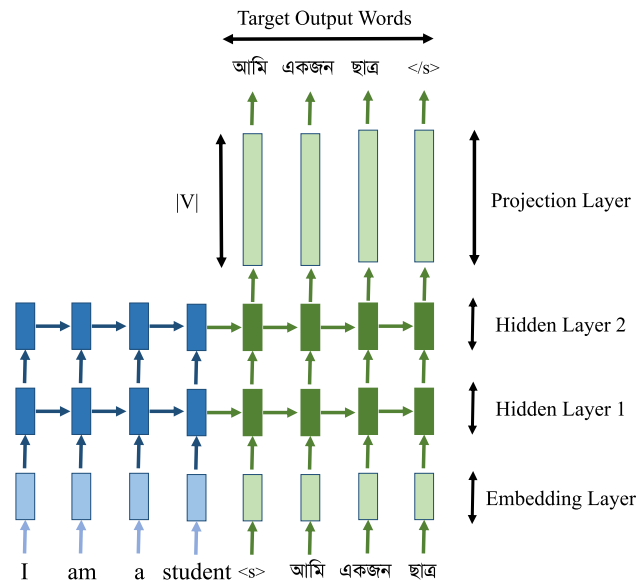
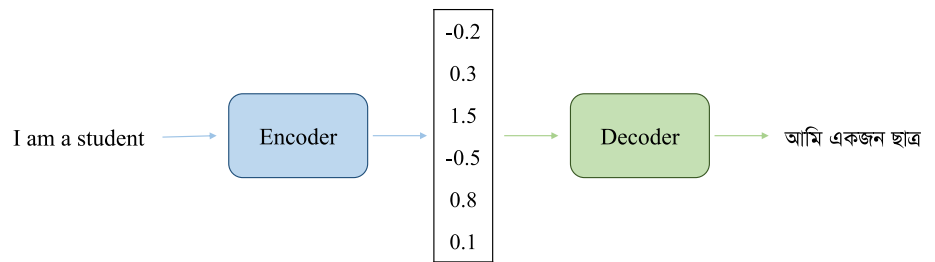


Fig. 3 Neural machine translation example of a deep recurrent architecture [21]

neural network. However, to avoid ‘exploding gradients’,⁴ we perform the gradient clipping by the global norm.

Next, the NMT model chooses an optimizer to control the attributes of its neural network (e.g., learning rate and weights) for diminishing the overall losses [34]. We choose the stochastic gradient descent (SGD) [34] optimizer (with a decreasing learning rate schedule) over the widely used Adam optimizer [18] based on the benchmarks achieved by [21].⁵

3.6 Inference with attention: generating translations

Once the NMT model has finished its training, it can generate translations of the source sentences that are not seen yet. This step of generating translations is termed as ‘Inference’. Inference differs from training since during

⁴ ‘Exploding gradients’ refers to the phenomenon when the updates to weights are significantly large leading to a numeric overflow or underflow during training.

⁵ Although Adam optimizer can generate plausible results for unknown neural network architectures, SGD with learning rate scheduling generally offers superior results.

inference, the NMT model has access to the input sentence only. During inference, firstly, NMT encodes the input sentence using the encoder (similar to training phase). Next, it initiates the beam search decoding process upon receiving a special starting symbol (*<s>*) as shown in Fig. 3. Then, at each decoder time step, NMT computes the attention to generate the RNN’s output as logit vector.

Attention mechanism basically operates as an interface between encoder and decoder in order to transfer relevant information from the encoder hidden states to the decoder. Computation of attention involves several successive steps such as attention weights (a_{ts}) derivation, context vector (c_t) calculation, and final attention vector (a_t) computation, which is then input to the next time step. We incorporate Luong’s attention mechanism⁶ [22] for computing a_{ts} , c_t , and a_t using the following three equations, respectively.

$$a_{ts} = \frac{\exp(\text{score}(h_t, \bar{h}_s))}{\sum_{s=1}^S \exp(\text{score}(h_t, \bar{h}_s))} \tag{1}$$

$$c_t = \sum_s a_{ts} \bar{h}_s \tag{2}$$

$$a_t = f(c_t, h_t) = \tanh(W_c [c_t; h_t]) \tag{3}$$

Here at time step t , ‘score’ function [22] is used to compare the decoder hidden state h_t with the encoder hidden states \bar{h}_s , which is normalized to produced the weights (a_{ts}). Afterwards, the computed attention vector a_t is utilized for obtaining the logit vector and loss using Softmax.

Using the logit vector, our NMT model applies beam search decoding technique with $width = 10$ (i.e., selecting the top 10 words based on the logit values in a depth first search fashion) at each decoder time step. The process terminates when the decoder outputs the special ending marker (*</s>*) as shown in Fig. 3. We limit the translation lengths by decoding up to twice the source sentence lengths.

Although the NMT models thrive to achieve the accuracy level of human translators in high-resource language settings, they always suffer from several major weaknesses.

⁶ We choose Luong’s attention over Bahdanau’s attention because of the flexibility of using the former one in different settings. The key difference between these two attention mechanisms lies in their alignment scores calculation.

Three such inherent weaknesses of NMT are: (1) inefficacy in handling atypical or unknown words, (2) torpid training and inference speed, and (3) sometimes, inability to translate every word in the input sentence [6, 19]. Thus, to enhance the translation performance of NMT (particularly in low-resource settings), we investigate the integration of rule-based approach with NMT in the subsequent sections.

4 Proposed methodology

Our work initially focuses on adopting a rule-based translator for Bengali to English translation. To our best knowledge, building a reasonably working rule-based *Bengali* → *English* translator has been studied only in [13] and [14]. Next, our target is to explore and implement the classical NMT⁷ [41] as discussed in the previous section. To do so, we collect and build datasets (Bengali–English parallel corpora) of different sizes from different sources. Subsequently, after implementing both rule-based translator and classical NMT in isolation, we integrate these two translators using different approaches to investigate the best possible translation performance. We present our proposed mechanisms and algorithms in details next.

4.1 Blending rule-based translator with corpus-based translators

Scope of rule-based translator expands as we keep adding more rules. However, it is near-to-impossible to implement unlimited and ever changing grammatical rules for any language. Besides, it is hard to deal with rule interactions in big systems, grammatical ambiguities,⁸ and idiomatic expressions.⁹ As a result, the potential of corpus-based machine translation (NMT and SMT) comes to light. Recently, NMT has emerged as the most popular machine translation system. However, both NMT and SMT have their own major limitation in terms of generating accurate translations for low-resource languages as shown in Fig. 1 earlier. Thus, both rule-based translator and corpus-based translators exhibit their advantages and limitations compared to other. This finding leads to our investigation on blending between rule-based translator and corpus-based translators.

To do so, we implement the classical NMT in our system from an open source resource⁷. Then, we integrate a Bengali to English rule-based translator [13, 14] with the classical NMT to investigate whether such an integration

can achieve a better performance in translation. To be precise, we explore the blending in three different ways:

- NMT followed by rule-based translation,
- Rule-based translation followed by NMT, and
- Either NMT or rule-based translation

Figure 4 illustrates how we can implement the possible blending approaches in our system. Note that we also implement similar approaches using SMT in place of NMT. Besides, we present our blending techniques in Algorithm 1, Algorithm 2 and 3. We discuss each of these three techniques next.

4.1.1 NMT followed by rule-based translation (NMT+RB)

Classical NMT initially requires training with parallel corpus (sentence pairs of source language and target language). In our case, we develop and adopt parallel corpus of different sizes containing Bengali–English sentence pairs for training the NMT. After training, we feed the intended input sentences to the NMT and generate the output translated sentences using the classical NMT approach. After getting the NMT generated translated sentence, our blending approach applies grammatical rules on the translated sentence to further modify the sentence to improve its translation performance (Fig. 4a). In our experimentation, we consider a deep multi-layer recurrent neural network (RNN), which is bidirectional and uses LSTM as a recurrent unit.

Algorithm 1 shows the skeleton of our blending approaches. Here, using the token tagging (parts-of-speech) information from the rule-based translator [13], our blending system substitutes some of the words or phrases in the NMT generated translated sentence with the translated words or phrases obtained from the rule-based translator. More specifically, rule-based translator just further ameliorates the skeleton of the translated sentence that NMT has already built as shown in Algorithm 2.

Algorithm 2 considers NMT generated translation and rule-based translation as ‘sentence1’ and ‘sentence2’, respectively, for ‘NMT followed by rule-based’ blending approach. Here, if our blending system finds any pair of unmatched words (tokens) having the same parts-of-speech (PoS_tag) between these two sentences, then our system replaces the NMT word with the corresponding rule-based word. This is how our system checks each word in the NMT generated translation with each word in the rule-based translation for replacement.

⁷ <https://github.com/tensorflow/nmt>.

⁸ <https://www.thoughtco.com/syntactic-ambiguity-grammar-1692179>.

⁹ <https://www.ef.com/wwen/english-resources/english-idioms>.

Algorithm 1 Blending between rule-based translator and corpus-based translators (NMT or SMT)

```

procedure GETBLENDINGOUTPUT
  Input : Source sentence (Bengali)
  Output : Target sentence (English) after blending
  NMT_Output  $\leftarrow$  Output generated by NMT (or SMT)
  RB_Output  $\leftarrow$  Output generated by rule-based translator
  Word  $\leftarrow$  An object with two attributes - token (word) and PoS_tag (Parts of Speech)
  PoS_tagger(sentence)  $\leftarrow$  ArrayList of Word objects with PoS_tag for each word in the sentence
  src_len  $\leftarrow$  Length of source sentence
  NMT_words  $\leftarrow$  ArrayList of Word objects (words with PoS tagging) in NMT
  RB_words  $\leftarrow$  ArrayList of Word objects (words with PoS tagging) in rule-based translation
  Translation_NMT+RB  $\leftarrow$  ArrayList of Word objects after blending (NMT followed by rule-based)
  Translation_RB+NMT  $\leftarrow$  ArrayList of Word objects after blending (RB followed by rule-based)
  Translation_NMTorRB  $\leftarrow$  ArrayList of Word objects after blending (Either NMT or rule-based)
  NMT_words := PoS_tagger(NMT_Output)
  RB_words := PoS_tagger(RB_Output)
  Translation_NMT + RB := PerformBlending(NMT_words, RB_words)
  Translation_RB + NMT := PerformBlending(RB_words, NMT_words)
  Translation_NMTorRB := PerformBlending_NMTorRB(NMT_Output,
  RB_Output, src_len)
  displayTranslation_NMT+RB
  displayTranslation_RB+NMT
  displayTranslation_NMTorRB
  
```

Figure 5 shows an example of how this blending technique works. Here, apart from generating translation of the source sentence by NMT, we also generate its rule-based translation. Next, our blending system matches translations from both the translators word by word using the parts-of-speech tagging information of the rule-based translator.

In the figure, the input Bengali sentence is pronounced as “*Oisheo tar kajti shesh kortechilo*”. Its reference translation is “*Oishee also was finishing her work*”.¹⁰ Here, NMT translates Bengali name “Oishee” to “Ishii”, where “Ishii” is tagged as noun. However, “Oishee” gets the same PoS_tag in rule-based translation. Therefore, first, this blending technique replaces “Ishii” by “Oishee” in the final translation. Afterwards, similarly, it also replaces “had”, “finish”, and “his” by “was”, “finishing”, and “her”, respectively, keeping the words in other positions intact. Here, all of these substitutions contribute in improving translation performance.

This technique proves itself to be the best blending technique (will be evaluated next) because of the fact that it realizes skeleton of translation from NMT and token-based attributes (person, number, tense, etc.) from rule-based translation. These two different forms of realizations best fit to strengths of the two different translation approaches.

4.1.2 Rule-based translation followed by NMT (RB+NMT)

Our next blending technique implements a reverse sequence of the previous blending technique. We modify the rule-based translated sentence by NMT in this blending technique as shown in Fig. 4b. Similar to the earlier case, Algorithm 2 also illustrates this blending technique. This time, our system considers rule-based translation as ‘sentence1’ and NMT generated translation as ‘sentence2’ in Algorithm 2.

¹⁰ We consider this source sentence to illustrate each of our blending approaches.

Algorithm 2 Blending module for NMT followed by rule-based approach and rule-based followed by NMT approach

```

procedure PERFORMBLENDING
  Input : sentence1  $\leftarrow$  ArrayList of Word objects in the sentence on which blending will be performed
  Input : sentence2  $\leftarrow$  ArrayList of Word objects in the sentence with which sentence1 will be blended
  Output : Translated sentence after performing blending
  sent1_len  $\leftarrow$  Length of sentence1
  sent2_len  $\leftarrow$  Length of sentence2
  sent1_word  $\leftarrow$  A Word object in sentence1
  sent2_word  $\leftarrow$  A Word object in sentence2
  blended_Translation  $\leftarrow$  Output sentence after performing blending
  blended_Translation := NULL
  for i := 0 to i < sent1_len do
    sent1_word := sentence1.get(i)
    token1 := sent1_word.token
    PoS1 := sent1_word.PoS_tag
    for j := 0 to j < sent2_len do
      sent2_word := sentence2.get(j)
      token2 := sent2_word.token
      PoS2 := sent2_word.PoS_tag
      if token1 = token2 then
        continue
      else
        if PoS1 = PoS2 then
          sentence1.set(i, sent2_word)
          sentence2.remove(j)
          break
      blended_Translation := blended_Translation + “ ” + sentence1.get(i).token
  return blended_Translation
  
```

Major limitation of this technique originates from the fact that NMT can generate completely wrong words during translation since NMT always predicts the next word in sequence based on its training data. On the other hand, rule-based translator at least cannot pick wrong words since it only searches the vocabulary for any particular word translation and pick the translated word if found. Therefore, if the blending system further modifies the rule-based translated sentence by NMT then it can happen that translation performance degrades in many cases. Only luck with this approach is when rule-based translator cannot recognize the source sentence due to lack of appropriate ruleset, which may leave some space for NMT to contribute plausibly.

Figure 6 presents an example on how this technique performs translation for the same source sentence (in Fig. 5). Here, initially, the two unmatched words - “Oishee” in rule-based translation and “Ishii” in NMT, hold the same PoS_tag (noun). Therefore, first, this blending approach replaces “Oishee” by “Ishii”. Afterwards, it also replaces “was”, “finishing”, and “her” by

“had”, “finish”, and “his”, respectively, as shown in Fig. 6. Unfortunately, in this example, all of these replaced words are incorrect; thus, degrading the translation performance.

4.1.3 Either NMT or rule-based translator (NMT or RB)

This blending technique is much simpler compared to the earlier ones. It basically performs choosing one between two translations generated by rule-based translator and NMT separately as shown in Fig. 4c. However, this blending system needs to make the choice based on some criteria so that it chooses the better one.

We find that the rule-based translator performs better for smaller sentences (not more than 6 words). We present a quantitative analysis to reflect this statement later in Sect. 7.1.4. Therefore, this blending approach chooses rule-based translation if the source sentence is smaller in length (less than 7). Otherwise, it chooses NMT generated translation as the output translation. We present this blending approach in Algorithm 3.

Algorithm 3 Blending module for Either NMT or rule-based approach

```

procedure PERFORMBLENDING_NMTORRB
  Input : sentence1 ← Corpus-based Translation
  Input : sentence2 ← Rule-based Translation
  Input : src_len ← Length of source sentence
  Output : Translated sentence after blending
    blended_Translation ← Output sentence
    blended_Translation := NULL
    if src_len ≤ 6 then
      blended_Translation := sentence2
    else
      blended_Translation := sentence1
    return blended_Translation

```

Figure 7 shows a working example of this blending technique. In the figure, we identify the source sentence (in Fig. 5) as a small sentence with only five words. Since our blending system considers sentences consisting of less than 7 words as small sentences, the system selects translation generated by the rule-based translator as the final translation and ignores NMT this time. Besides, note that we can update the selection criteria (sentence type) in this blending system according to the scope of the rule-based translator. The more we add rules, the more types of sentences (having different lengths) we can translate using rule-based translator. Therefore, selection criteria can be made much more flexible and tricky in this system depending on performance analysis after incorporating more rules.

5 Experimental settings

We perform rigorous performance evaluation of our different approaches on the basis of different types of metrics. We need to employ considerable resources for our experimentation, as such experimentation are resource-hungry and time consuming in general. To perform our experimentation with NMT, we use Python language, PyCharm IDE, Tensorflow library, and Linux (64-bit) operating system. Here, we consider an encoder–decoder model with a deep multi-layer RNN, which uses LSTM as a recurrent unit.

To start the experimentation, we design datasets for training NMT and testing its performance. We use the following hyper-parameters in our system for training NMT with our designed datasets—(1) 2-layer LSTMs of 512-dim hidden units with bidirectional encoder, (2) 12k–100k training steps, (3) 20% dropout rate, (4) Luong’s attention (scale=True), (5) embedding dimension 512, and (6) SGD optimizer with initial learning rate 1.00.¹¹

¹¹ Learning rate is halved at regular interval depending on the number of training steps. For example, for 12k training steps, we start halving it every 1k step after passing first 8k training steps.

Besides, we use sigmoid (σ) and hyperbolic tangent (Tanh) as the activation functions [8] in the LSTM cells,¹² and apply softmax in the output layer. We choose these hyper-parameters and activation functions based on the benchmarks achieved for English–Vietnamese and German–English translations in [21]. Besides, after experimenting with multifarious parameter settings, we find the best results for *Bengali* → *English* neural machine translation using this setup.

Afterwards, to experiment with SMT, we use Moses toolkit (with GIZA, SRILM, and IRSTLM)¹³ written in C, C++, and Perl. Besides, we use JAVA language, Netbeans IDE, Sqlite database, and Opennlp tools for both implementing the rule-based translation system and performing blending between rule-based and corpus-based translators.

6 Datasets and evaluation metrics

Designing and developing datasets has been one of the most challenging and time intensive tasks in our experimentation. For training the NMT reasonably, we require a large parallel corpus containing both source language and target language. In our case, NMT requires such a corpus of Bengali–English sentence pairs. However, we find very few sources available for constructing a reasonable sized dataset containing Bengali–English sentence pairs. Hence, one of the major contributions in this study is that we build three novel parallel corpora containing Bengali–English sentence pairs, which will enhance future research opportunity for Bengali language.

¹² Tanh (-1 to 1 probability distribution) is used for the cell state activation, and the sigmoid (0 to 1 probability distribution) is used for the node output. Note that we choose Tanh over ReLU because ReLU can have very large outputs (i.e., exploding nature) for LSTM.

¹³ <https://www.dlsi.ua.es/~mlf/fosmt-moses.html>.

Fig. 4 Different blending techniques between rule-based translation and NMT

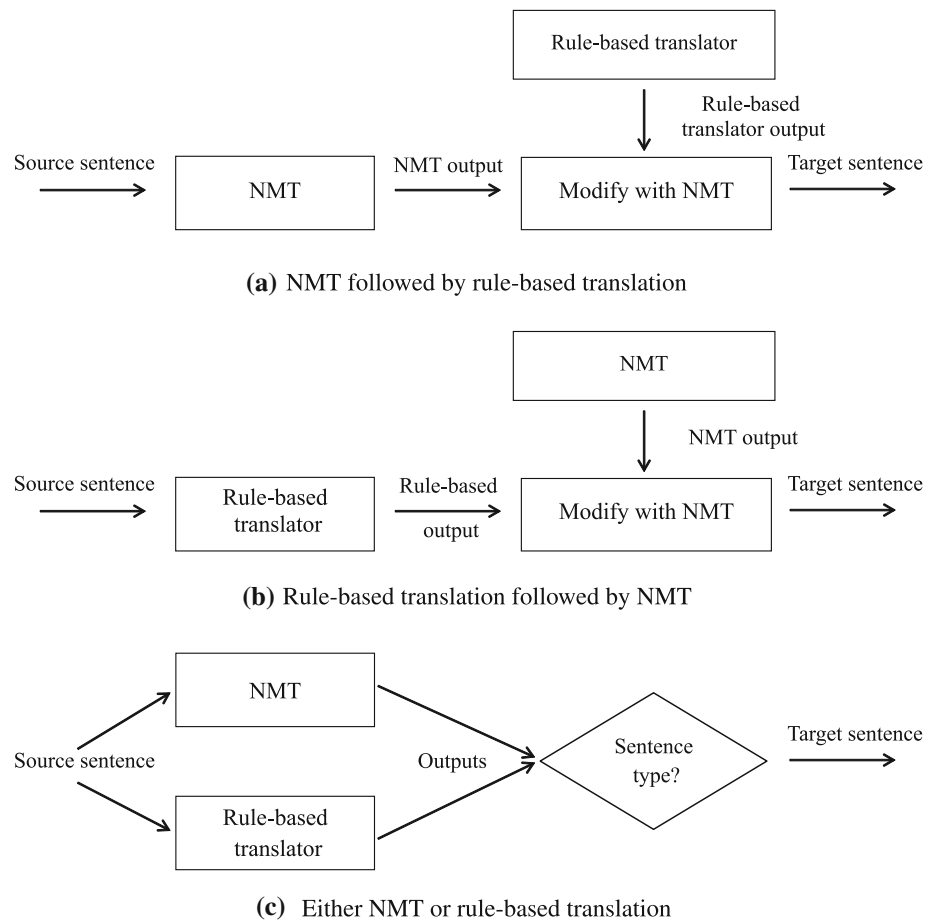
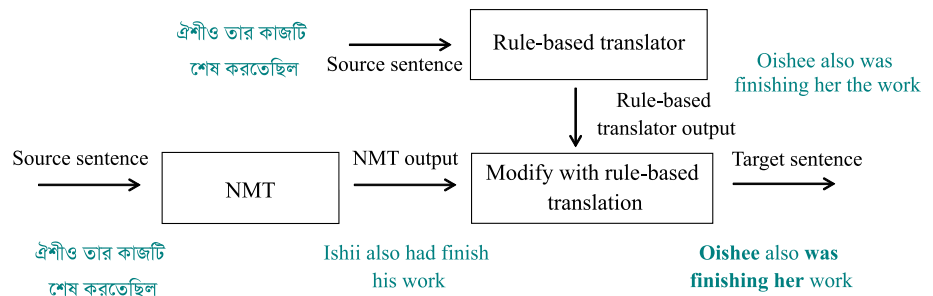


Fig. 5 An example to demonstrate NMT followed by rule-based translation



6.1 Demography of datasets

We develop our dataset of Bengali–English parallel corpus from well-established contents such as Al-Quran,¹⁴ newspapers,¹⁵ movie subtitles,¹⁶ and university websites.¹⁷ Besides, we translate different example-based individual Bengali sentences into English and accumulate them in the

¹⁴ <https://www.kaggle.com/zusmani/the-holy-quran>.

¹⁵ <https://www.prothomalo.com>.

¹⁶ <https://www.subscene.com>.

¹⁷ <https://www.sust.edu/d/cse/research>.

dataset. In fact, we create the corpus mostly at our own by translating different Bengali sentences to English one by one. Figure 8a illustrates a demography of our full dataset.

Initially, we experiment with only literature-based source (Al-Quran) of our full dataset since its size is large enough to be considered as a separate dataset when compared to the size our full dataset. Afterwards, we also experiment with our full dataset with an intent to generate results from a fairly diversified dataset. Therefore, our full dataset also includes another dataset (custom dataset) as its subset (apart from the literature-based dataset). However,

Fig. 6 An example to demonstrate rule-based translation followed by NMT

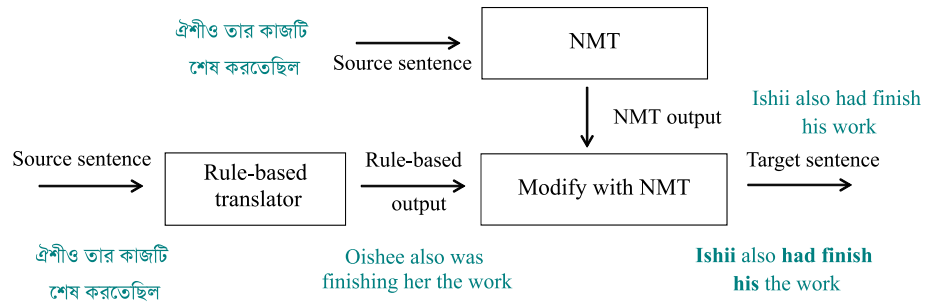
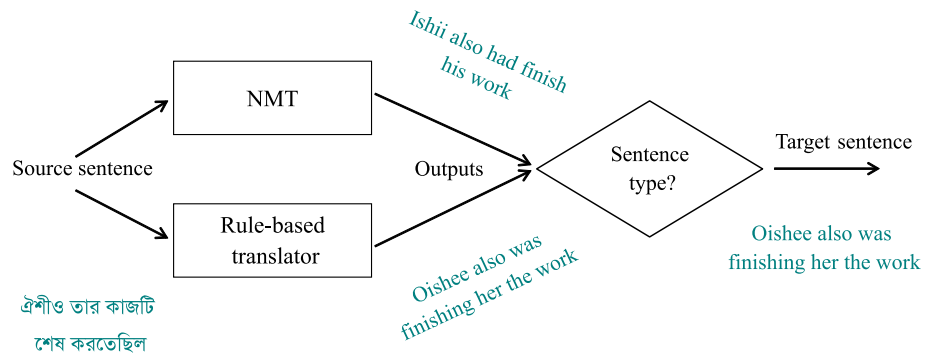


Fig. 7 An example of choosing either NMT or rule-based translation



we do not consider using this custom dataset independently in our experimentation since its size is too small to train an NMT system reasonably. We present a demography of our custom dataset (a subset of full dataset) in Fig. 8b. Besides, both Bengali and English sentences in our full dataset vary in size or length. Figure 9 reflects percentages (%) of different types of sentences in our full dataset in terms of different sizes or lengths.

There is another dataset containing more than 1 million Bengali–English parallel sentences, which is primarily collected from a website.¹⁸ Similar to our full dataset, we show the percentages (%) of different types of sentences in this dataset in terms of different sizes or lengths in Fig. 10. However, the sentences in this dataset contain numerous unknown characters and words (even from other languages such as Arabic, Chinese, German, etc.), which needs to be cleaned first before using in experimentation. Therefore, we carefully remove such unknown characters from this dataset. Besides, there are English sentences that are not proper translations of corresponding Bengali sentences in this dataset. Therefore, this dataset requires further rigorous manual checking and translation alignment for each sentence pair, which we leave for now as an immediate future work.

Table 1 shows summary of the different datasets. For experimentation, each dataset is split into train (around 80%), development and test data by choosing sentences randomly, where train and test data are mutually exclusive.

¹⁸ <http://opus.nlpl.eu/GlobalVoices.php>.

Also, development and test data are independent of each other. Besides, all the data (sentences) are tokenized and segmented into subword symbols using byte-pair encoding (BPE) [38] with 32,000 operations.

6.2 Representativeness in our datasets

We analyze the representativeness in our dataset using Zipf’s law [28]. Zipf’s law pertains to frequency distribution of words in a language (or a dataset of the language, which is large enough to be a representative of the language). To demonstrate that Zipf’s law holds in our dataset, we compute $freq(r)$ that involves computing frequency and ranking of each word. Then, we compute $r \times freq(r)$ to check whether $r \times freq(r)$ becomes approximately a constant in all cases. The simplest way to show that Zipf’s law holds in a dataset is to plot the computed values and check whether the slope is proportionately downward. Here, instead of plotting $freq(r)$ versus rank, it is better to plot $\log(r)$ in the X axis and $\log(freq(r))$ in the Y axis. Accordingly, we plot the computed values for both Bengali corpus and English corpus separately in two different graphs.

We present the graphs for our first dataset (literature-based dataset) in Fig. 11a, b, respectively. Figure 11a shows that our Bengali corpus exhibits a bit deviation from Zipf’s law; however, our English corpus perfectly follows Zipf’s law. Similarly, we present the graphs for our second dataset (full dataset) in Fig. 12a, b, respectively. Here,

Fig. 8 Demography of our datasets

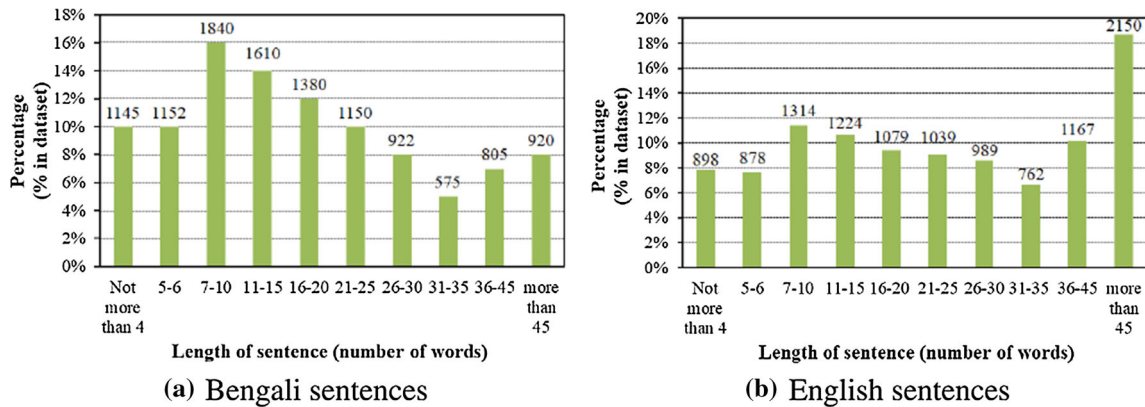
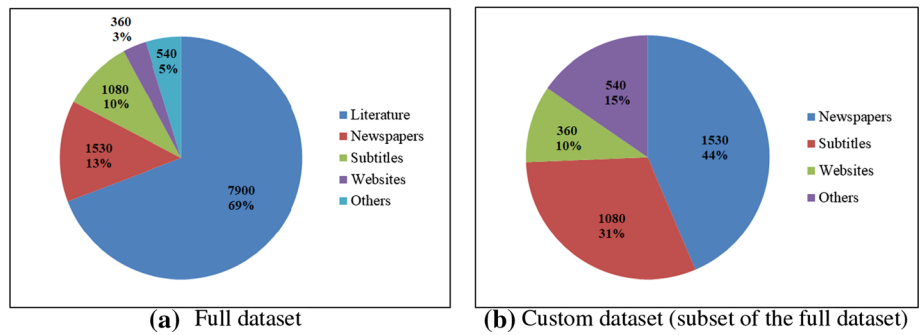


Fig. 9 Percentages of sizes of sentences in the full dataset

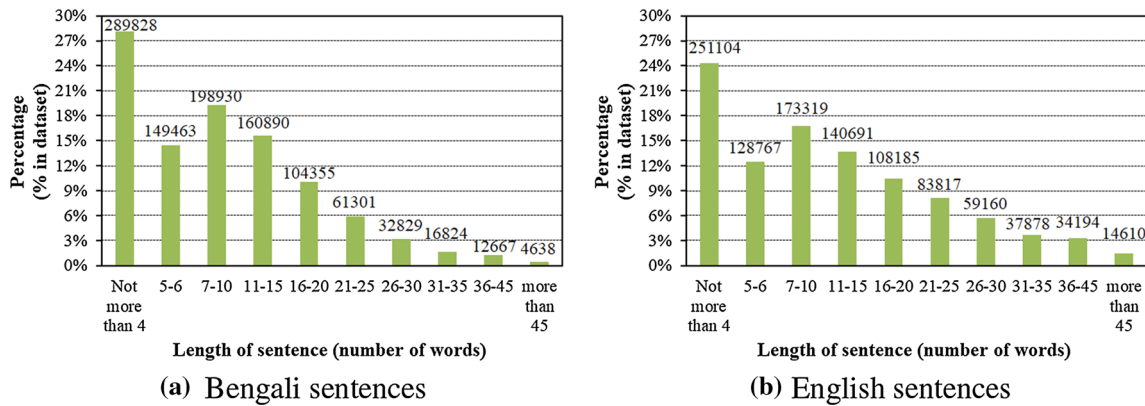


Fig. 10 Percentages of sizes of sentences in the GlobalVoices dataset

Table 1 Summary of the different datasets

Dataset	Sentences	Sources
Literature-based	8000	Al-Quran
Custom	3500	Newspapers, subtitles, websites, etc.
Full (combined)	11,500	Literature-based and custom
GlobalVoices ¹⁸	10,31,724	Website

Fig. 12a shows that our Bengali corpus of full dataset exhibits lesser deviation from Zipf’s law than Bengali

corpus of literature-based dataset due to combining literature-based dataset with custom dataset.

6.3 Performance evaluation metrics

For the purpose of performance evaluation of our system, we adopt three different metrics that are widely used for evaluating performance of machine translation—BLEU [27], METEOR [5], and TER [39]. Based on the above-mentioned performance metrics, we evaluate performances of our proposed blending approaches through rigorous experimentation. Next, we present results and findings of

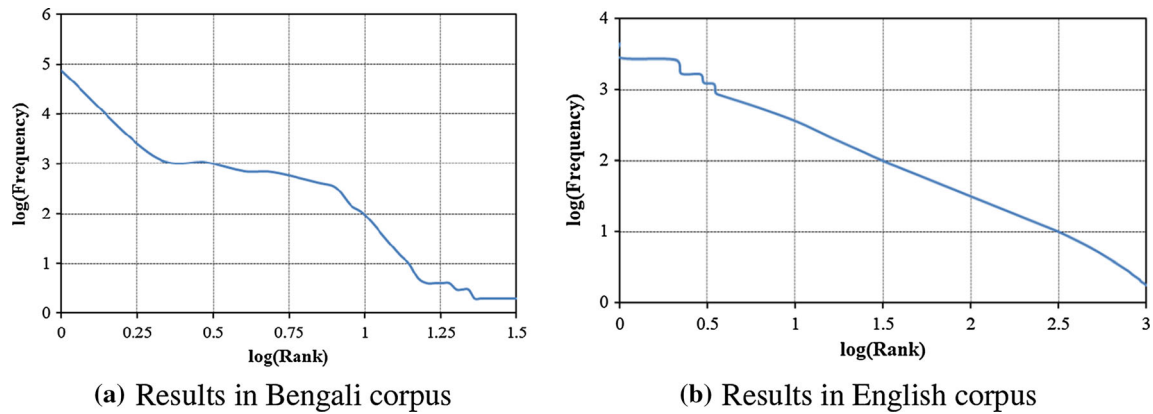


Fig. 11 Representativeness in our literature-based dataset according to Zipf's law

the evaluation using different datasets. Here, we report case-sensitive BLEU scores in all cases.

7 Experimental results from our different blending approaches

After implementation of both rule-based translator and classical NMT, we blend between these two approaches using three different techniques as discussed earlier (Sect. 4.1). We analyze performances of each of these approaches with three standard metrics, namely BLEU, METEOR, and TER as presented earlier. We consider three different datasets with different sizes for analyzing the performances. Hence, as already mentioned, we adopt a literature-based dataset (from Al-Quran) and create another dataset from different sources except any literature. The latter dataset, i.e., our custom dataset, is relatively smaller in size (around 3,500 parallel sentence pairs), which is too small to train an NMT system reasonably. Therefore, we combine this dataset with our literature-based dataset to form another dataset (full dataset) for experimentation.

Finally, we also show results in a high-resource context using our GlobalVoices dataset.

7.1 Results using literature-based dataset

First, we present results (scores of performance metrics) obtained from translation over our literature-based dataset in Table 2. We choose around 12,000 training steps for this dataset, where average length of test sentences is 10 words. Table 2 shows a comparison among all the approaches (in isolation and in combination) using the standard performance metrics. Here, the higher the METEOR score and the BLEU score, and the lower the TER score; the better the performance is. From Table 2, we notice that ‘NMT followed by rule-based’ (NMT+rule-based) blending technique exhibits significant improvement over the classical NMT. More specifically, it emerges as the best blending technique that gets reflected in the performance scores using each of the three metrics. Therefore, we can understand that our blending approaches can significantly improve performance of NMT generated translations. To

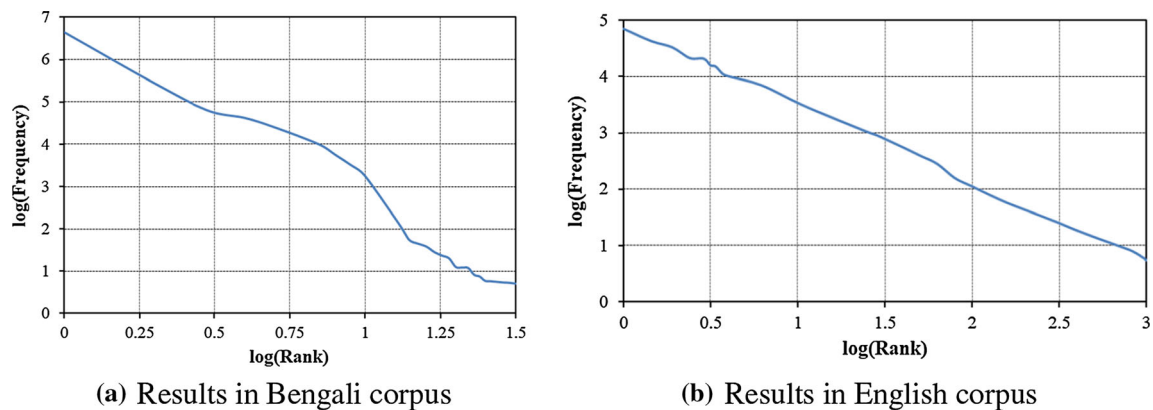


Fig. 12 Representativeness in our full dataset according to Zipf's law

Table 2 Comparison among different translation approaches

Score	NMT	Rule-based (RB)	NMT+RB	RB+NMT	NMT or RB
BLEU ↑	8.56	1.28	11.43	0.84	8.80
METEOR ↑	12.34	13.50	20.31	10.62	12.43
TER ↓	93.73	93.90	85.09	96.62	93.50

be precise, the best way of blending appears to be applying grammatical rules after translating by NMT.

The main reason behind this finding is that NMT discerns the basic skeleton of translation first. However, NMT often tends to output sequence of incorrect words based on prediction. This is where the strength of a rule-based translator lies as it either translates the words (or phrases) from vocabulary or cannot translate at all. In short, rule-based translator generally does not output any incorrect word-by-word translation as it does not predict. As a result, as discussed in Sect. 4.1.1, modifying NMT with rule-based translation results in replacement of incorrect words (output by NMT) with correct words (output by rule-based translator) in most of the cases, which leads to improved performance scores as presented in Table 2.

Another blending technique, ‘either NMT or rule-based’ (NMT or rule-based), also shows slight improvement over the classical NMT. We actually anticipate that since we carefully choose the best between the translations from rule-based translator and NMT as per the types (lengths) of sentences in this technique. However, performance scores decline in ‘rule-based followed by NMT’ (rule-based+NMT) blending approach, which points out the inability of NMT to further improve translations done by the rule-based translator, as this technique tends to replace the correct words with the incorrect ones (i.e., opposite to ‘NMT+rule-based’ approach). Moreover, Table 2 reflects that performance of both NMT and rule-based translator in isolation is worse than two of our blending approaches (‘NMT+rule-based’ and ‘NMT or rule-based’), which justifies the significance of exploring these blending techniques between these two translators. Specifically, note that the performance of *Bengali* → *English* rule-based translator is quite poor (BLEU = 1.28), as its implemented rules are not sufficient to plausibly recognize and translate the test sentences in our corpus, which also limits the improvement achieved by our ‘NMT+rule-based’ blending technique significantly.

Table 3 reflects a closer look at BLEU scores of all the approaches as per consideration of different n-grams ($n = 1, 2, 3,$ and 4). Ideally, BLEU score is considered for n-gram model where $n = 4$. In all cases, including $n = 4$, the blending of ‘NMT followed by rule-based’ outperforms all other alternatives. Besides, n-gram scores decrease from

$n = 1$ to $n = 4$, as we look forward to matching larger chunk of contiguous words (i.e., from one word matching to a sequence of four words matching).

Next, we present comparisons over classical NMT and other approaches graphically to portray the individual performance scores for several test sentences (chosen randomly). We show comparisons using METEOR and TER scores where light red lines indicate NMT score and deep blue lines indicate each one of the other approaches one by one. Here, we adopt NMT as the benchmark (baseline) approach in all the graphs, as it is commonly adopted by the widely-used Google translator. Note that we do not show any comparison in terms of BLEU score at sentence level as BLEU is generally calculated over the entire test corpus, and its characteristics are identical to METEOR’s.

7.1.1 Comparison between NMT and only rule-based approach

Firstly, Figs. 13 and 14 show a comparison between NMT and only rule-based approach (deep blue lines) in isolation in terms of METEOR and TER scores, respectively. Here, in Fig. 13, we see that the red lines (i.e, NMT) exceed the blue lines (i.e., rule-based) in most of the cases for METEOR scores, and vice-versa in Fig. 14 for TER scores. Thus, these two figures reflect that the overall performance of only rule-based approach is worse than NMT in isolation for literature-based dataset.

7.1.2 Comparison between NMT and ‘NMT followed by rule-based’ approach

Next, we show the performance of one of our blending techniques, ‘NMT followed by rule-based’ (NMT+rule-based), in terms of METEOR and TER scores in Figs. 15 and 16 respectively. Here, deep blue lines indicate the scores obtained using ‘NMT followed by rule-based’ blending approach. In Fig. 15, we see that the blue lines (i.e, NMT+rule-based) exceed the red lines (i.e., NMT) in most of the cases for METEOR scores, and vice-versa in Fig. 16 for TER scores. Thus, we can see significant improvement over classical NMT in these figures.

Table 3 Comparison as per BLEU scores

n-gram	NMT	Rule-based (RB)	NMT+RB	RB+NMT	NMT or RB
1-g	31.46	31.74	46.07	25.59	31.35
2-g	17.18	10.70	25.19	7.56	17.37
3-g	11.42	3.70	16.04	2.60	11.66
4-g	8.56	1.28	11.43	0.84	8.80

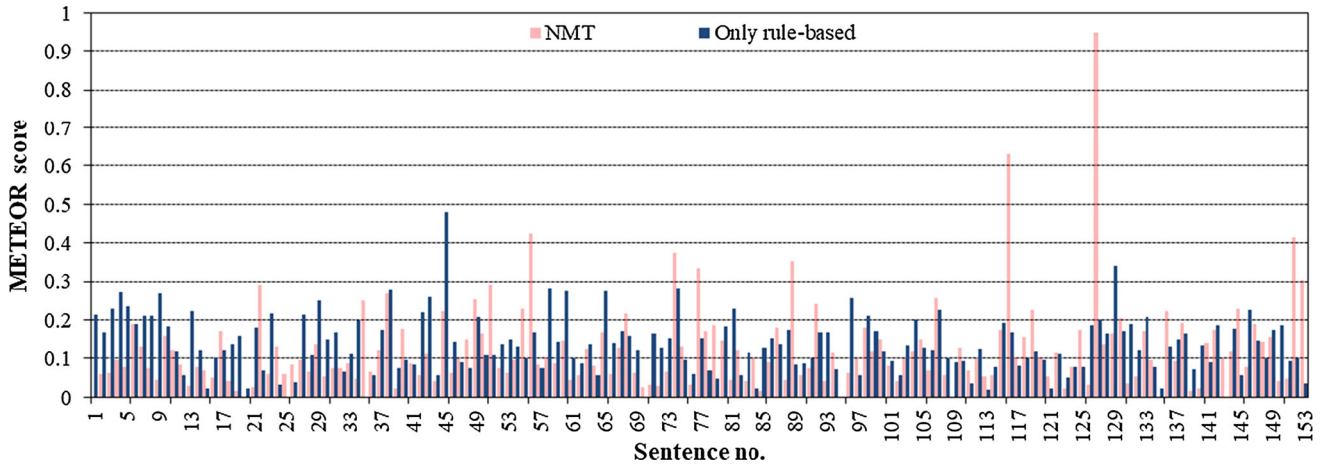


Fig. 13 NMT versus only rule-based METEOR score

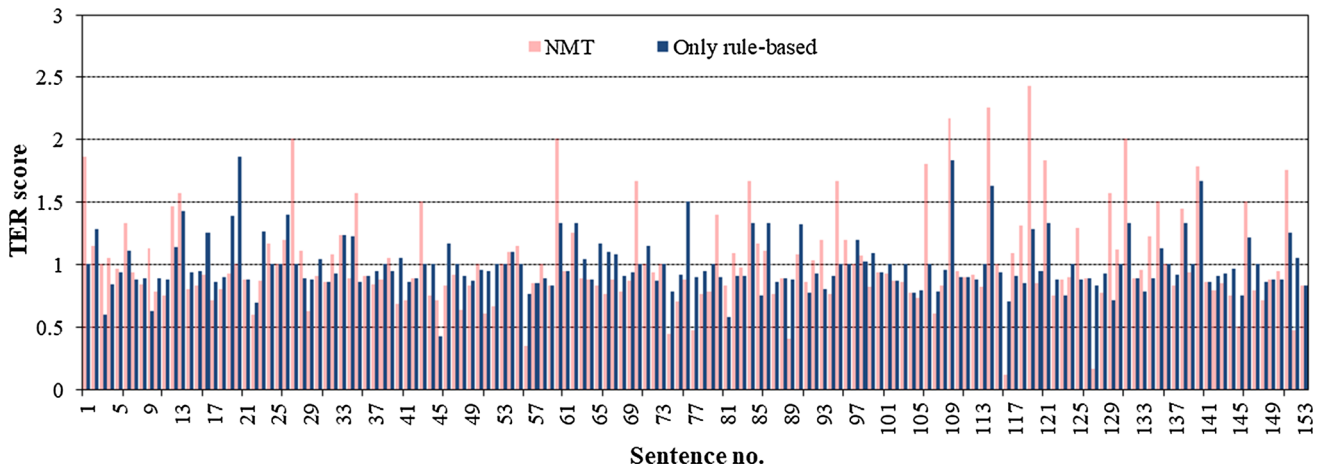


Fig. 14 NMT versus only rule-based TER score

Actually, these two figures reflect the performance of our best blending technique in terms of METEOR and TER scores.

7.1.3 Comparison between NMT and ‘rule-based followed by NMT’ approach

After that, we present the results of ‘rule-based followed by NMT’ (rule-based+NMT), in terms of METEOR and TER scores in Figs. 17 and 18 respectively. Here, deep blue lines indicate the scores obtained using ‘rule-based

followed by NMT blending’ approach. In Fig. 17, we see that the red lines (i.e, NMT) exceed the blue lines (i.e., rule-based+NMT) in most of the cases for METEOR scores, and vice-versa in Fig. 18 for TER scores. Hence, these two figures reflect that ‘rule-based followed by NMT’ approach performs poorly when compared to the classical NMT. In fact, this blending technique proves itself to be the worst performer among all the approaches.

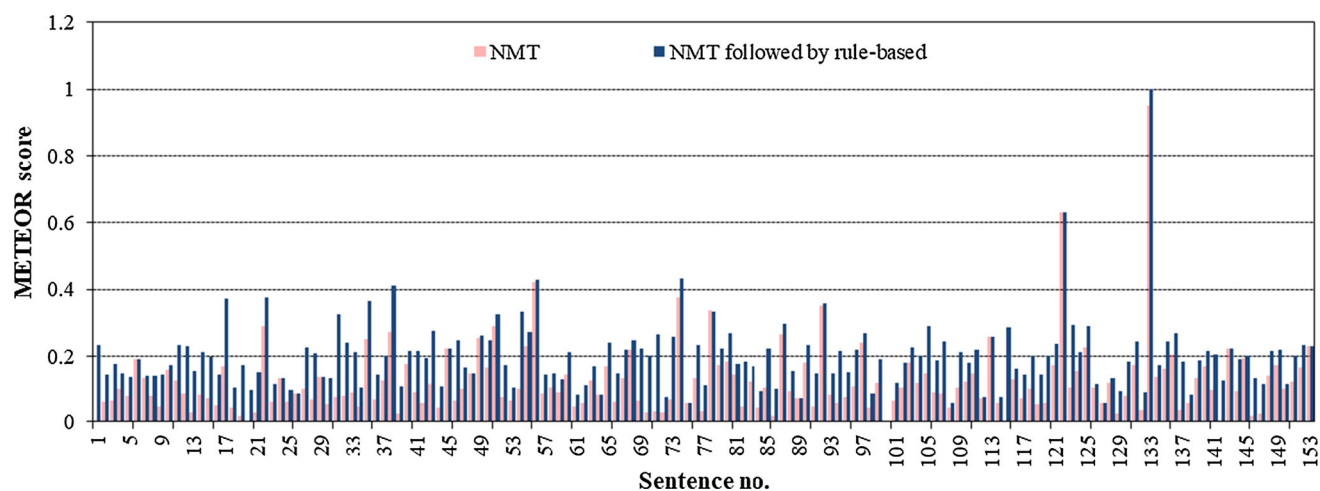


Fig. 15 NMT versus NMT followed by rule-based METEOR score

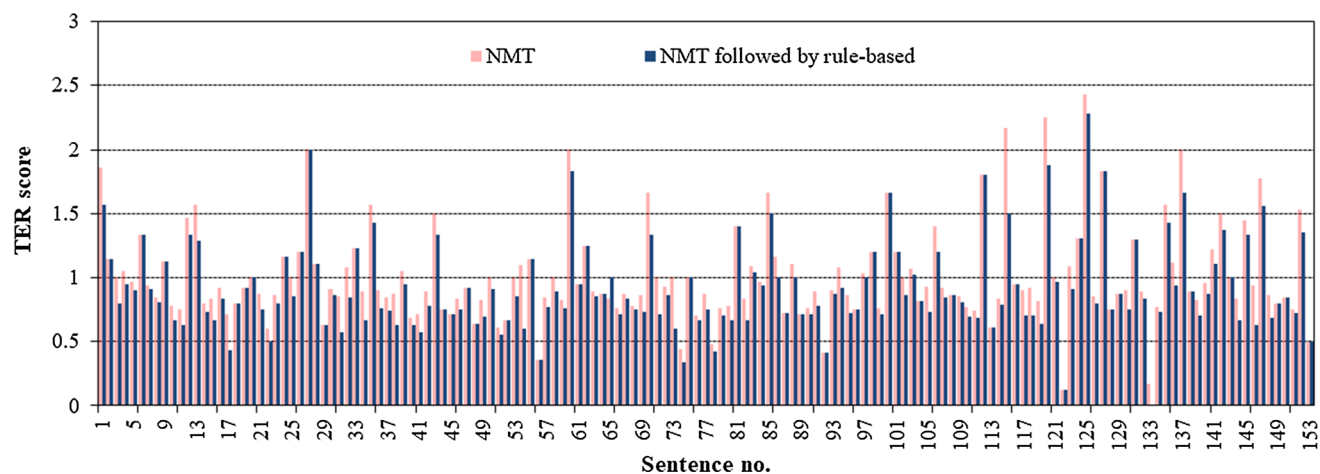


Fig. 16 NMT versus NMT followed by rule-based TER score

7.1.4 Comparison between NMT and ‘either NMT or rule-based’ approach

Finally, we present the results of ‘either NMT or rule-based’ blending technique in Figs. 19 and 20. Here, deep blue lines indicate the scores obtained using this blending approach. In both figures, we see that the red lines (i.e., NMT) are on the same level with the blue lines (i.e., NMT or rule-based) in most of the cases for METEOR and TER scores. Thus, this approach performs on par with classical NMT. Main reason behind this result is that most of the test sentences are lengthy (more than 6 words) in this dataset. Here, we choose such definition for lengthy or large sentence by performing an experimental analysis to test the performance of rule-based translator over different test datasets aggregated based on different length categories as shown in Table 4.

Table 4 shows that performance of rule-based translator drastically drops (e.g., from BLEU = 35.29 to BLEU = 2.69

for 120 implemented rules) if the average length of sentences exceeds 6 words, which justifies our criteria of defining lengthy sentence in this blending approach. Since this approach chooses NMT generated translation if the length of the sentence is large, it chooses NMT generated translations mostly. However, this approach performs at least as good as classical NMT.

To recapitulate, we can clearly notice that light red lines exceed deep blue lines for most of the sentences in Figs. 13 and 17. That means, both only rule-based approach and ‘rule-based followed by NMT’ approach perform worse than NMT in isolation. In contrast, deep blue lines exceed light red lines in Fig. 15 for almost all the sentences, which reflects the clear victory of our ‘NMT followed by rule-based’ approach over NMT in isolation. In addition to that, we notice that light red lines and deep blue lines are mostly at the same level in Fig. 19, which reflects the on par performance of our ‘either NMT or rule-based’ approach as discussed above.

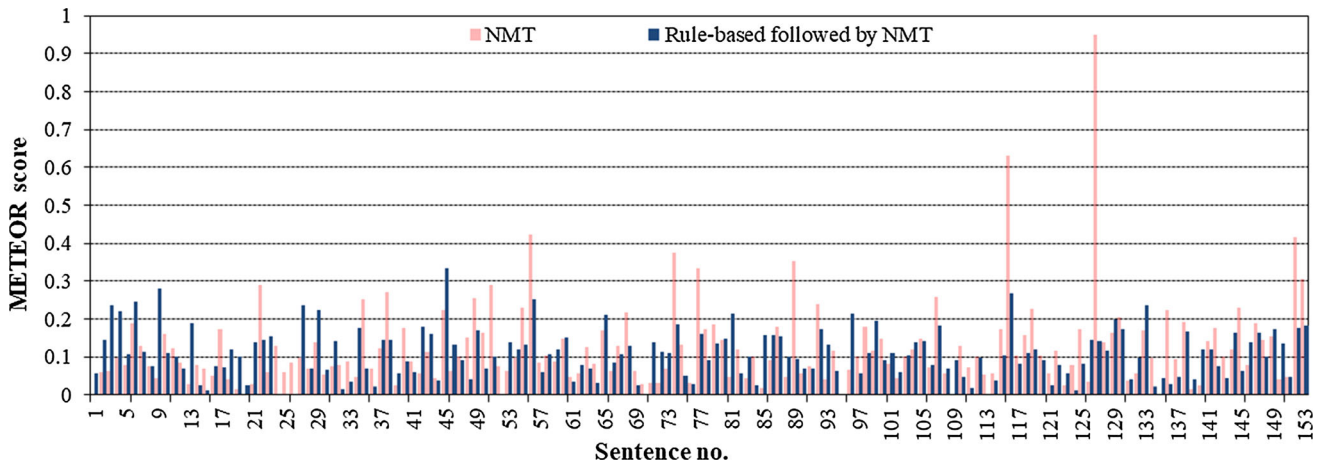


Fig. 17 NMT versus rule-based followed by NMT METEOR score

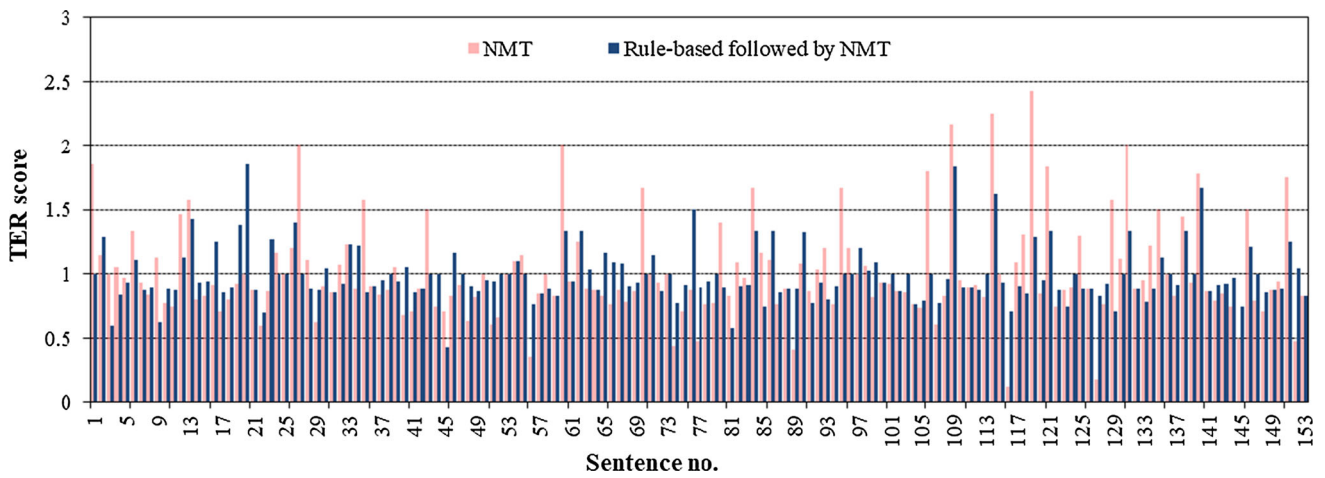


Fig. 18 NMT versus rule-based followed by NMT TER score

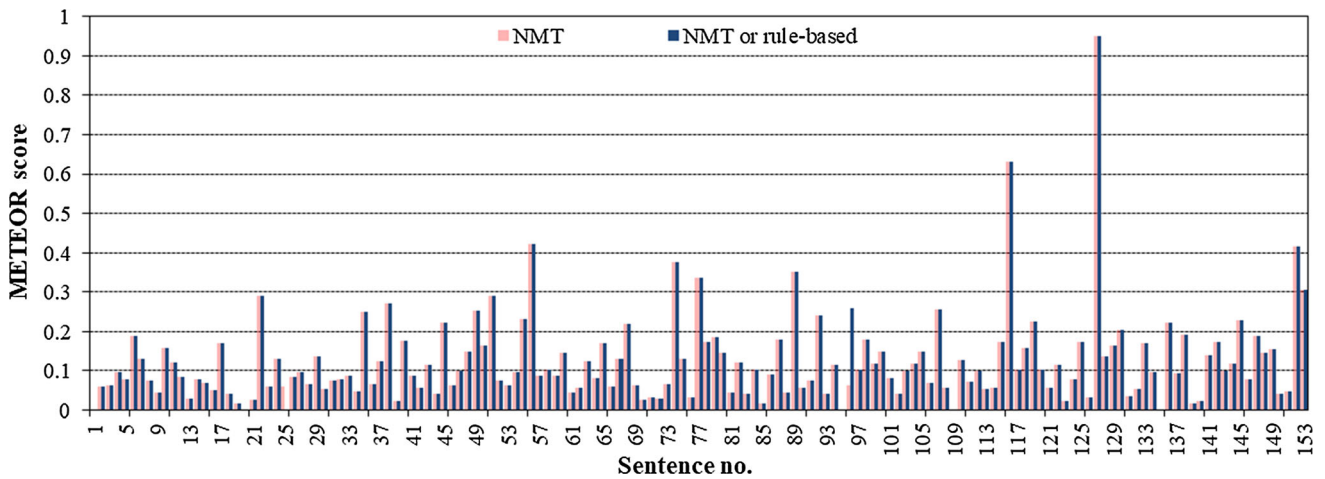


Fig. 19 NMT versus NMT or rule-based METEOR score

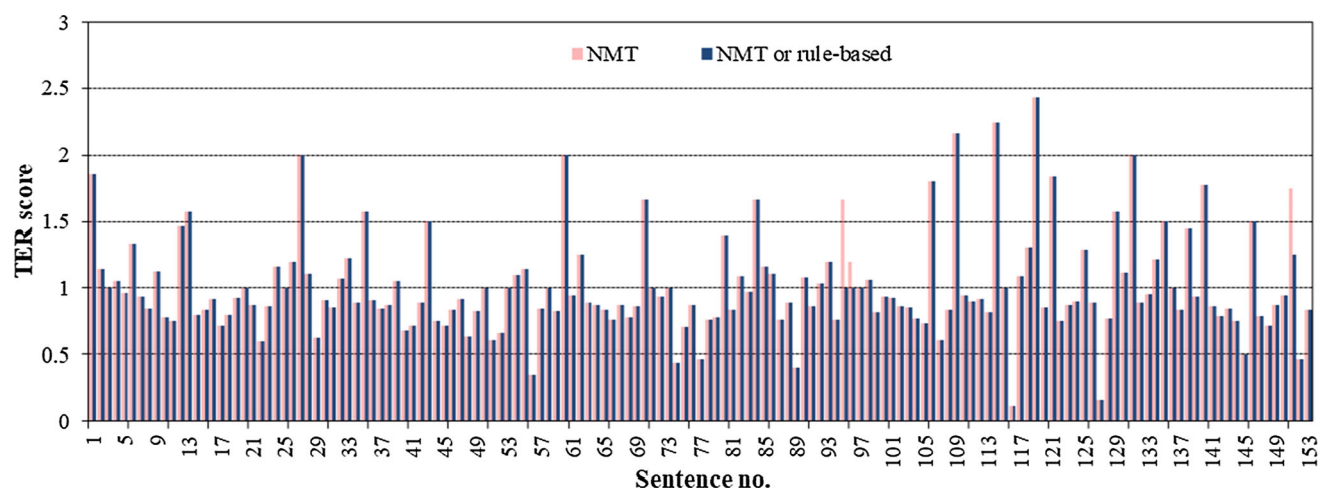


Fig. 20 NMT versus NMT or rule-based TER score

Table 4 Performance of rule-based translator with different number of implemented rules over different length categories of test sentences in terms of BLEU score

Implemented rules	<4 words	4–6 words	7–10 words	11–15 words	>15 words
30	18.43	25.32	2.06	1.95	0.77
50	27.20	33.14	2.57	2.25	1.06
120	28.86	35.29	2.69	2.48	1.14

7.1.5 Analysis on sensitivity of our operational parameter

Performance of our adopted rule-based translator changes as we increase the number of rules or we add more rules. However, adding rules seems like a never-ending process. Therefore, we analyze how implementation of different numbers of rules impacts on the performance scores of our different approaches.

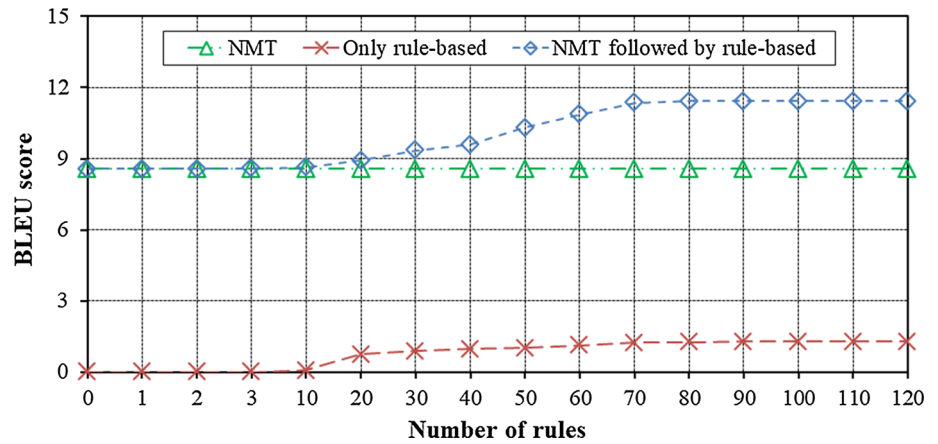
BLEU score increases as number of implemented rules increases in our system as shown in Fig. 21. In this figure, we show performance of three different approaches with respect to an increase in the number of added rules—only rule-based approach, NMT, and ‘NMT followed by rule-based’ approach. We notice that the curves of only rule-based approach and ‘NMT followed by rule-base’ approach show a gradual increase (initially sharp) in BLEU score as the number of implemented rules increases. Besides, the curves tend to become flat after implementing around 90–100 rules in our system. It depends on the order in which different rules are being added. In our system, we implement more basic and important grammatical rules such as basic sentence structures, verb identification, tenses, etc., first. That is why, the curve shows a sharp rise in between first 3–10 implemented rules, and then rises consistently until 70–80 rules are added. In our system, we add the most important rules that significantly improve the

translation performance within around first 50 rules. Afterwards, addition of more rules merely impacts on changing the performance score significantly, since those rules such as detection of subject’s gender, punctuations, etc., seem to be less contributing compared to the previously added (first 50–60) rules.

Nonetheless, the curve for NMT remains flat (parallel to X axis) since performance of NMT does not change with the number of implemented rules. Moreover, although the curve of ‘NMT followed by rule-based’ approach exhibits characteristics nearly similar to that of only rule-based approach, it does not directly originate from the curve of rule-based approach using any mathematical formula. However, if the performance of translation generated by only rule-based approach improves then our blending (‘NMT followed by rule-based’ approach) also improves its performance to some extent since our system blends with that improved rule-based translation after generating translation by NMT. This is why, we notice such similarity between these two curves.

Similarly, we illustrate variation of METEOR scores with respect to the number of added rules. Figure 22 presents the results for only rule-based approach, NMT, and ‘NMT followed by rule-based’ approach in terms of METEOR score.

Fig. 21 Variation of BLEU scores with an increase in the number of implemented rules



Trends in the curves for METEOR scores of these three approaches are similar to what we have just presented for BLEU scores above. Here, we show variation for only ‘NMT followed by rule-based’ approach, as this approach leads all other approaches. Note that the curves in this approach does not start from zero score, as NMT already sets a positive score that our blending system further increases by applying rules.

Next, we also show variation of TER scores with an increase in the number of rules in Fig. 23 for only rule-based approach, NMT, and ‘NMT followed by rule-based’ approach. Expectedly, apart from curve of NMT, behaviour of remaining two curves for TER scores is exactly opposite to that of the previous curves, as TER scores decrease with an increase in the number of rules. Here, less score refers to better performance as TER score basically refers to an error rate. Similar to the previous cases, the curves go almost flat after the addition of first 70 rules.

Finally, we present a combined graph (Fig. 24) containing normalized values of all the metrics for both rule-based approach and ‘NMT followed by rule-based’ approach. Here, in case of values of each metric, we normalize the values with respect to our found maximum values. The combined presentation of all the normalized

values in Fig. 24 demonstrates efficacy of our proposed best blending approach, as its application improves performance scores in all cases.

That is all about experimentation on performance scores using our literature-based dataset. However, we also perform similar experimentation using another dataset (full dataset) since scores obtained from only one dataset may not be enough to draw any convincing conclusion on translation performance.

7.2 Results using full dataset

Next, we perform experimentation using our combined (literature-based and custom) or full dataset. We choose around 12,000 training steps for this dataset too, where average length of test sentences is 15 words. Table 5 shows summary of results obtained using this dataset.

Table 5 strongly supports the results obtained earlier (Table 2) using our literature-based dataset with similar reasoning as discussed in Sect. 7.1. Here, ‘NMT followed by rule-based’ blending approach again outperforms all other approaches. In addition to that, ‘Either NMT or rule-based’ approach remains as our second best approach.

Fig. 22 Variation of METEOR scores with an increase in the number of implemented rules

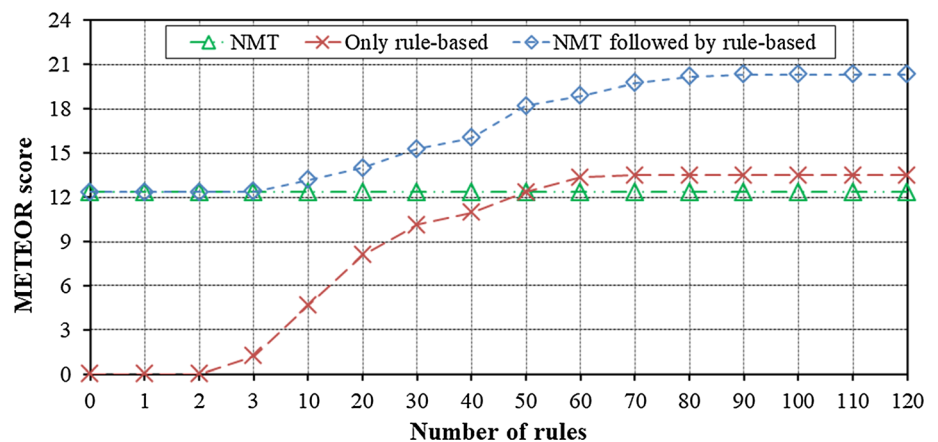


Fig. 23 Variation of TER scores with an increase in the number of implemented rules

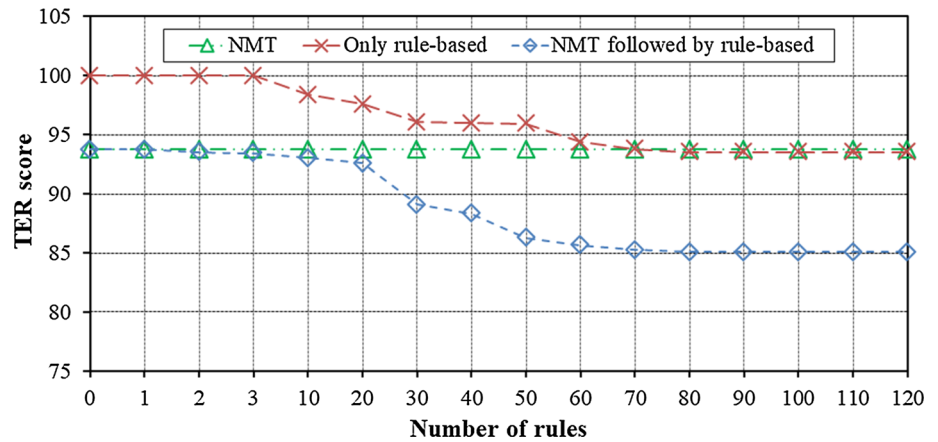


Fig. 24 Comparison of normalized performance scores with an increase in the number of implemented rules for literature-based dataset

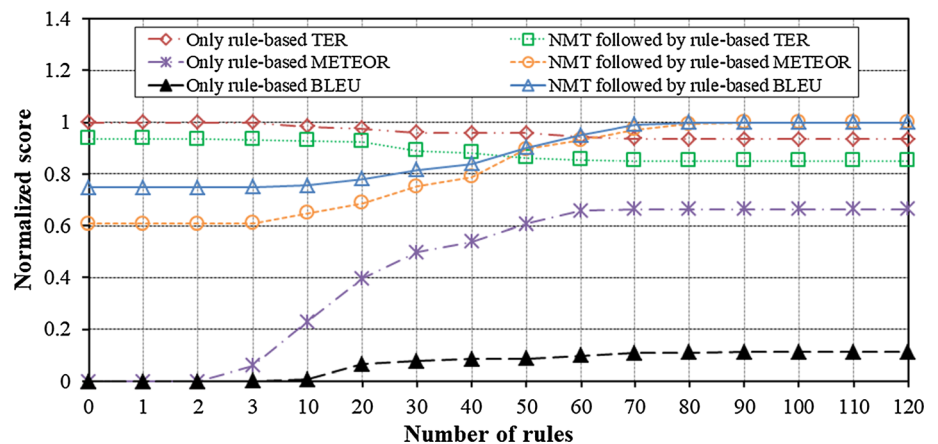


Table 5 Comparison among different translation approaches for full (combined) dataset

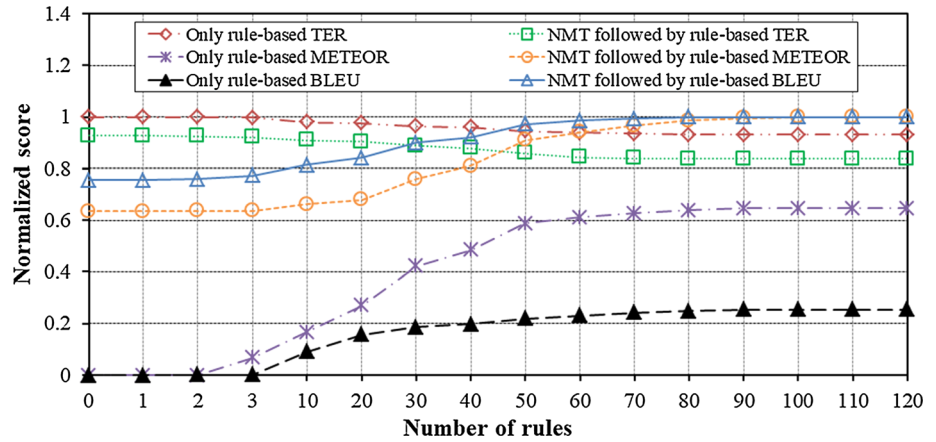
Score	NMT	Rule-based (RB)	NMT+RB	RB+NMT	NMT or RB
BLEU ↑	9.28	3.13	12.26	1.34	9.87
METEOR ↑	14.18	14.43	22.32	12.86	14.92
TER ↓	92.78	93.21	83.83	95.52	92

Afterwards, similar to our literature-based dataset, we also present another combined graph (Fig. 25) containing normalized values of all the metrics for both rule-based approach and ‘NMT followed by rule-based’ approach using our full dataset. Figure 25 reflects that the graph for our full dataset exhibits similar behaviour with respect to our previous dataset (literature-based). Therefore, we have just double-checked and justified our observation on performance scores of different approaches discussed earlier (for literature-based dataset), using our combined dataset this time.

8 Overall experimental findings

At this stage, we present our overall experimental findings in terms of average percentage (%) improvement of our different blending approaches over different parameters such as BLEU, METEOR, and TER in Tables 6, 7, 8, and 9. We show this results based on the performance scores presented in Table 2 (discussed in Sect. 7.1) and Table 5 (discussed in Sect. 7.2) for literature-based dataset and full dataset, respectively. Here, Tables 6 and 7 reflect the results (average percentage (%) improvement) for literature-based dataset and full dataset, respectively, with respect to NMT. Note that we find these percentage

Fig. 25 Comparison of normalized performance scores with an increase in the number of implemented rules for full dataset



improvements of our different approaches keeping NMT as baseline.

Similarly, Tables 8 and 9 reflect the results (average percentage (%) improvement) for literature-based dataset and full dataset, respectively, with respect to only rule-based approach. Here, we find that % improvements of our different blending approaches with respect to rule-based approach is much higher than NMT approach. This is because, our adopted NMT model performs superior to the *Bengali* → *English* rule-based translator in isolation.

9 Extension of our experimental results with statistical machine translation

Machine translation is in practice for long time in different forms such as Example-based Machine Translation, Phrase-based Machine Translation, Statistical Machine Translation (SMT), Neural Machine Translation (NMT), etc. NMT is the most recent technology in machine translation, which outperforms all other translation approaches. This is why, we attempt to contribute in machine translation keeping NMT as our prime focus, and adopt NMT in our system. Furthermore, to justify the efficacy of our blending approaches, we extend our experimentation on another popular corpus-based machine translation technology, SMT. SMT was used by popular Google Translator just before NMT, not more than five years earlier.

Besides, Mumin et al. [24] reported a phrase-based statistical machine translation system between English and

Bengali languages in both directions claiming to have achieved a promising BLEU score 17.43 for Bengali to English translation. In this regard, we adopt their baseline SMT system and follow their mechanism to investigate the performance of SMT using our dataset. To do so, first, we implement a popular SMT toolkit, Moses¹³, and we configure the system following their configuration process. Next, we train the SMT system with our full (literature-based and custom) dataset. Finally, we evaluate the performance of SMT using our dataset.

We achieve a BLEU score of 12.31 using the baseline SMT. In addition to that, we investigate the performance of our different blending approaches. We blend our rule-based translator with SMT this time. We present the performance scores of different approaches in Table 10.

Table 10 reflects that our ‘SMT (or NMT) followed by rule-based’ approach stills remains the best translation approach. Interestingly, performance of ‘SMT followed by rule-based’ approach (BLEU = 16.43) is even better than ‘NMT followed by rule-based’ approach (BLEU = 12.26) since in this case, SMT (BLEU = 12.31) performs better than NMT (BLEU = 9.28) in isolation. This happens because, our dataset is not large enough to train an NMT system efficiently. SMT perhaps takes this advantage to outperform NMT for this dataset. Besides, our best approach (BLEU = 16.43) lags behind their proposed approach (BLEU = 17.43 [24]) in terms of overall performance score because of our insufficient training data this time. They trained their system with a much larger dataset than our current (full) dataset. However, their training dataset is not made publicly available.

Table 6 Overall percentage (%) improvement over different parameters with respect to NMT for literature-based dataset

Parameters	NMT+rule-based (%)	Rule-based+NMT (%)	NMT or rule-based (%)
BLEU	34	– 90	3
METEOR	65	– 14	1
TER	9	– 3	0

Table 7 Overall percentage (%) improvement over different parameters with respect to NMT for full dataset

Parameters	NMT+rule-based (%)	Rule-based+NMT (%)	NMT or rule-based (%)
BLEU	32	– 86	6
METEOR	57	– 9	5
TER	10	– 3	1

Table 8 Overall percentage (%) improvement over different parameters with respect to rule-based approach for literature-based dataset

Parameters	NMT+rule-based (%)	Rule-based+NMT (%)	NMT or rule-based (%)
BLEU	793	– 34	588
METEOR	50	– 21	– 8
TER	9	– 3	0

Table 9 Overall percentage (%) improvement over different parameters with respect to rule-based approach for full dataset

Parameters	NMT+rule-based (%)	Rule-based+NMT (%)	NMT or rule-based (%)
BLEU	292	– 57	215
METEOR	55	– 11	3
TER	10	– 2	1

Table 10 Comparison among different translation approaches considering SMT as baseline system

Score	SMT	Rule-based (RB)	SMT+RB	RB+SMT	SMT or RB
BLEU ↑	12.31	3.13	16.43	2.16	14.14
METEOR ↑	15.35	14.43	22.33	13.48	20.92
TER ↓	88.14	93.21	82	93.35	85.38

Nonetheless, baseline SMT scores 16.91 using their dataset [24], whereas it scores 12.31 using our full dataset. Afterwards, they achieve BLEU score 17.43 in their approach¹⁹ over SMT score 16.91, which offers an improvement of 3% over baseline SMT. However, our best translation (blending) approach achieves BLEU score 16.43 over SMT score 12.31, which offers 34% improvement over baseline SMT. Therefore, we expect to achieve a higher BLEU score when we can plausibly match their dataset (discussed in the next section). Furthermore, this extended experimentation leads to an important finding—“Any corpus-based translation (NMT or SMT) generated by machine can be significantly improved after blending with rule-based translation”.

10 Extending our study to a high-resource context

Performance of corpus-based translators (NMT or SMT) largely depends on availability of significant amount of training data. However, the largest dataset used in our experimentation presented so far consists of up to 11,500

parallel Bengali–English sentences. Only 11,500 sentences may not really satisfy the need for significant amount of training data for an NMT system mimicking the context of extremely low-resource language.

However, we are yet to show what would happen if we take our approach to a high-resource context. Therefore, we extend our study to a high-resource context by developing a larger Bengali–English parallel corpus containing more than one million sentence pairs¹⁸. Here, the average length of sentences in test dataset is 11 words. We summarize the performance scores of our different approaches obtained using this dataset in Table 11 for NMT-based approaches and in Table 12 for SMT-based approaches. Results reflected in Tables 11 and 12 clearly establish that our ‘NMT/SMT followed by rule-based’ approach performs the best over all other alternative approaches. We also notice that BLEU scores of both corpus-based translators increase in high-resource context when compared to their scores over our literature-based dataset and full dataset due to larger training data. This contributes in boosting the performance scores of our blending approaches, achieving a new benchmark (BLEU = 18.73) for *Bengali* → *English* translation.

Next, we present the improvement in performance scores of all the approaches with respect to an increase in

¹⁹ They term it as ‘shu-torjoma’ [24].

Table 11 Comparison among different NMT-based translation approaches for a high-resource context

Score	NMT	Rule-based (RB)	NMT+RB	RB+NMT	NMT or RB
BLEU ↑	13.43	4.16	18.73	2.89	14.51
METEOR ↑	24.82	16.22	31.30	13.65	26.87
TER ↓	85.79	88.50	77.94	92.84	83.14

the size of dataset in Fig. 26. The figure shows that performance improves with an increase in the size of dataset. Here, we also show a comparison between NMT and ‘NMT followed by rule-based’ approach in terms of BLEU scores using our different datasets. Note that we find the best performance score (BLEU = 18.73) after extending our experimentation to the high-resource context (with one million sentence pairs), which is substantially higher than our previous best score (BLEU = 12.26) obtained for the low-resource context (with 11,500 sentence pairs).

Afterwards, we present the improvement in performance scores of all the approaches with respect to an increase in the number of training steps in the high-resource context in Fig. 27. The figure shows that performance improves as we increase the number of steps (up to 100k) for training the corpus-based translators. We choose up to 100,000 training steps for this dataset, where our best approach achieves the maximum score. However, for this dataset, we find that NMT tends to underperform with more than 100k training steps such as BLEU = 18.20 for 110k steps and BLEU = 17.14 for 120k steps.

11 Comparative analysis with benchmark translation approaches for low-resource languages

Finally, we show a comparative analysis of our proposed blending approaches with several recent work on low-resource language translations (Table 13). Sennrich et al. [37] proposed an approach for translating in a low-resource setting (*Turkish* → *English*) by pairing monolingual training data with an automatic back-translation (BT) to treat it as additional parallel training data. They developed synthetic source sentences (Turkish) using BT of target (English)²⁰ sentences to generate a synthetic parallel training set (*Gigaword_{synth}*), which they coupled with available parallel corpus containing only 320k Turkish-English sentence pairs (mimicking low-resource setting) to achieve an improvement over state-of-the-art approaches for *Turkish* → *English* translation. Therefore, we follow their training details (for *Turkish* → *English* translation) to adopt and test their proposed best approach (training with

Gigaword_{synth}) for *Bengali* → *English* translation over our different datasets.

In addition to that, we summarize comparisons of our approaches with NMT and SMT-based approaches (baseline SMT and shu-torjoma) presented by Wu et al. [41] and Mumin et al. [24] respectively in Table 13. Here, we show all the results in terms of BLEU scores only. Table 13 exhibits that our best blending approach ‘NMT (or SMT) followed by rule-based’ approach outperforms all other approaches for *Bengali* → *English* translation context. We also carefully notice that NMT overtakes SMT this time mainly because of larger training data. However, this training data contain numerous misaligned Bengali-English translation pairs, which definitely reduces the translation performance of both NMT and SMT, where SMT suffers worse than NMT.

Furthermore, we strongly anticipate that BLEU score for *Bengali* → *English* translation can be further improved by blending these corpus-based machine translation approaches (presented in [37] and [24]) with rule-based translator as we believe that any corpus-based translation generated by machine can be significantly improved after blending with rule-based translation. For now, we leave this further investigation as a future work.

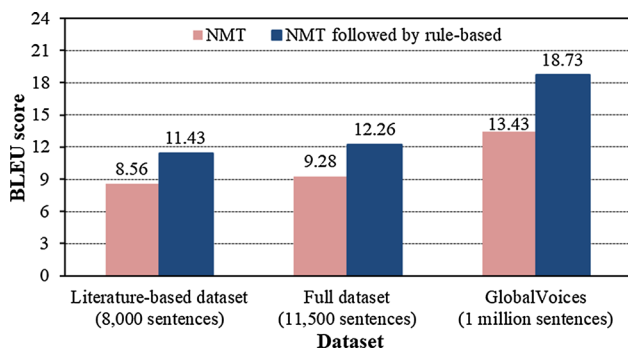
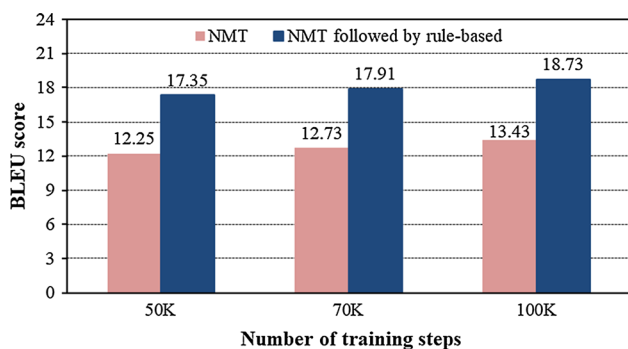
12 Discussions

Although we consider Bengali as the pilot source language for translating into English in this article, the proposed blending approaches are equally applicable for translating any other language given an appropriate rule-based translator (for rule-based translation) and a plausible parallel corpus comprising that language (for corpus-based translation). Similar to Bengali, Hindi, Arabic, Nepali, etc., are examples of grammatically rich and complex low-resource languages. However, these languages differ from each other in terms of inflections, accentuation marks, etc., which makes their interpretation and translation fairly distinct. Hence, translation of each language primarily depends on either the availability of good quality training data or the development of an appropriate rule-based translator that encompasses such language specific distinct features. Next, irrespective of translation qualities of both

²⁰ They used the English LDC Gigaword corpus (Fifth Edition).

Table 12 Comparison among different SMT-based translation approaches for a high-resource context

Score	SMT	Rule-based (RB)	SMT+RB	RB+SMT	SMT or RB
BLEU \uparrow	13.10	4.16	18.02	2.37	14.43
METEOR \uparrow	22.96	16.22	27.50	13.52	24.91
TER \downarrow	86.58	88.50	79.22	92.97	84.45

**Fig. 26** Comparison between NMT and ‘NMT followed by rule-based’ approach in terms of BLEU scores with different datasets**Fig. 27** Comparison between NMT and ‘NMT followed by rule-based’ approach in terms of BLEU scores with respect to an increase in the number of training steps

corpus-based (e.g., NMT) and rule-based translations of any source sentence in these languages, our proposed approaches can blend them using the PoS tagging

information of the translated sentences as discussed in Sect. 4.1, and exhibit improvements over NMT and/or rule-based translation. Table 14 shows two concrete examples of application of our proposed blending approaches for translating the following Hindi source sentences into English:

Source 1: उसने यह कार्य पूरा कर ली
[usane yah kaary poora kar lee]

Source 2: ओइशे भी अपना काम पूरा कर रही थी
[oishee bhee apana kaam poora kar rahee thee]

In Table 14, we get the corpus-based (NMT) translations from the Google Translate (collected on November 14, 2020), and assume the rule-based translations presuming a rule-based *Hindi* \rightarrow *English* translator that encompasses basic rules for translating simple Hindi sentences (i.e., exhibiting performance level at least similar to the basic *Bengali* \rightarrow *English* rule-based translator used in this article). Next, we show the PoS tagging of these two translations (NMT and rule-based in isolation) for one of the source sentences (Source 2) as follows:

NMT [(‘Oise’, ‘NNP’), (‘was’, ‘VBD’), (‘also’, ‘RB’), (‘doing’, ‘VBG’), (‘his’, ‘PRP\$’), (‘work’, ‘NN’)]

RB [(‘Oishee’, ‘NNP’), (‘also’, ‘RB’), (‘was’, ‘VBD’), (‘completing’, ‘VBG’), (‘her’, ‘PRP\$’), (‘work’, ‘NN’)]

Then, as discussed in Sect. 4.1, in ‘NMT followed by rule-based (NMT+RB)’ blending approach, words from the NMT generated translation gets replaced with words of the rule-based translation based on having similar PoS_tags (e.g., *Oise* \rightarrow *Oishee* (NNP), *doing* \rightarrow *completing* (VBG), and *his* \rightarrow *her* (PRP\$)); whereas, the reverse happens in

Table 13 Comparison among our proposed blending approaches and several (four) state-of-the-art approaches in terms of BLEU scores

Approach	Training data size (literature-based/ full/ GlobalVoices)	Literature-based	Full	GlobalVoices
<i>Gigaword_{synth}</i> [37]	6.5 k/9 k/825 k (parallel) + 3200 k (BT)	10.31	11	15.59
shu-torjoma [24]	6.5 k/9 k/825 k	–	12.86	13.61
NMT [41]	6.5 k/9 k/825 k	8.56	9.28	13.43
SMT [24]	6.5 k/9 k/825 k	–	12.31	13.10
SMT or RB	6.5 k/9 k/825 k	–	14.14	14.43
NMT or RB	6.5 k/9 k/825 k	8.80	9.87	14.51
SMT+RB	6.5 k/9 k/825 k	–	16.43	18.02
NMT+RB	6.5 k/9 k/825 k	11.43	12.26	18.73

Bold represents the maximum/highest score achieved for corresponding dataset

Table 14 Translation of two example Hindi sentences using NMT-based blending approaches

Category	Source sentence 1	Source sentence 2
Reference	He completed this task	Oishee was also completing her work
NMT	He did it	Oise was also doing his work
RB	He completed this task	Oishee also was completing her work
NMT+RB	He <i>completed</i> it	<i>Oishee</i> was also <i>completing</i> her work
RB+NMT	He <i>did</i> this work	<i>Oise</i> also was <i>doing</i> his work
NMT or RB	He completed this task	Oise was also doing his work

Table 15 Application of the proposed blending approaches in translating a Nepali sentence and an Arabic sentence

Category	Nepali	Arabic
Source	एक व्यक्तिले यी वाक्यहरु अनुवाद गर्नु पर्छ	لم يستطع أويش القيام بهذا العمل
Reference	He completed this task	Oishee could not do this work
NMT	He finished it	Uesh could not do the job
RB	He completed this task	Oishee could do this work
NMT+RB	He <i>completed</i> it	<i>Oishee</i> could not do <i>this</i> work
RB+NMT	He <i>finished</i> this work	<i>Uesh</i> could do <i>the</i> job
NMT or RB	He finished it	Oishee could do this work

‘RB+NMT’ approach. Besides, in ‘NMT or RB’ approach, the NMT generated translation is selected for Source 2 (size of source > 6) and rule-based translation for Source 1 (size of source < = 6) as shown in Table 14. Note that this selection criteria of this approach (i.e., length of the source sentence) needs to be fine-tuned depending on the nature of the source language and its rule-based translator. Hence, analogous to our findings in context of *Bengali* → *English* translation, our best blending approach ‘NMT+RB’ improves the translation quality of neural machine translations of both the source sentences (i.e., *Hindi* → *English* translation context).

Apart from this, in Table 15, we provide examples of translation of source sentences from Arabic and Nepali (high similarity with Hindi) languages to further demonstrate the implication of our blending approaches in other languages. Here, ‘NMT+RB’ again outperforms all other blending approaches.

In this section, we have considered very basic source sentences from three complex and grammatically sensitive languages—Hindi, Nepali, and Arabic. Such simple sentences can be plausibly realised and translated by corpus-based and rule-based translators.²¹ However, for randomly considered larger and more practical sentences (i.e., from a

corpus), both the translators tend to deviate more from the accurate translations (references). As a result, the impact of applying our different blending approaches (e.g., improvement achieved by ‘NMT+RB’ approach) becomes more clearly perceptible. In summary, the proposed blending approaches can be directly applied in improving the translation performance between different language pairs, when compared to NMT and/or rule-based translation in isolation.

13 Conclusion and future work

Millions of immigrants thrive for working knowledge on popular non-native languages such as English, as this creates many opportunities in international communities. Machine translators can offer a great help to accomplish such a laborious task using artificial intelligence. Moreover, corpus-based systems perform poorly for translating low-resource languages such as Bengali, Arabic, etc. Therefore, the importance of an efficient translator for such languages is noteworthy. In this article, we make our contribution from two perspectives. First, we adopt a Bengali–English rule-based translator, and separately incorporate two popular corpus-based machine translation approaches (NMT and SMT). Next, we explore different possible approaches for blending these two translation schemes (rule-based translation and corpus-based machine translation). Besides, we contribute in creating novel

²¹ In this study, we choose such simple sentences from these languages in order to exhibit examples of the least contribution scope of our blending approaches. We leave further analysis of different languages with more pragmatic and complex sentences (i.e., corpora) as a future work.

parallel corpora containing Bengali–English sentence pairs. We also evaluate performance of each of the blending approaches in terms of standard performance metrics for machine translation.

A number of critical issues always make natural language processing and translation tasks more complex. For a rule-based translator, there remain a number of exceptions that violate the standard rules of grammar, which are quite tough to tackle by implementing any number of rules. Hence, the efficiency of a rule-based translator in translating languages with complex grammatical structures is very low. On the other side of the coin, translations generated by corpus-based machine translators can be unreliable, offensively wrong, or utterly unintelligible sometimes [19]. Besides, such machine translation systems have a steeper learning curve with respect to the amount of training data, resulting in worse quality in low-resource settings. Thus, the performance of a rule-based translator is constrained by the number of incorporated rules, whereas the performance of a corpus-based translator is constrained by the amount of data fed to it for learning or training. In reality, it is very difficult to ensure sufficiency either in terms of the number of rules or in terms of the amount of data. Accordingly, neither of these two different types of approaches can suffice all alone.

Considering these aspects, in this article, we explore different approaches of blending between rule-based translator and corpus-based machine translators (NMT and SMT) mimicking the application of artificial intelligence to investigate whether and how a synergy between these translators can be attained. Our study leads to some promising outcomes as two of our blending approaches outperform both NMT and SMT in isolation, one of which achieves new state-of-the-art performance scores for Bengali to English translation.

While conducting our study, we have found that it is extremely difficult to get a large and effective parallel corpus for Bengali to English translation. Accordingly, we plan to advance our work on building such a corpus in future. Besides, we will also focus on improving the neural network level architecture used in NMT considering specific aspects of translating low-resource languages. In addition to that, we plan to explore other possible modes of blending such as phrase-based blending, trained blending, etc., in future. Finally, exploring and evaluating our proposed blending approaches for other language pairs remains yet another potential future work of this study.

Funding The authors did not receive support from any organization for the submitted work. No funding was received to assist with the preparation of this manuscript. No funding was received for conducting this study. No funds, Grants, or other support was received.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose. The authors have no conflicts of interest to declare that are relevant to the content of this article. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript. The authors have no financial or proprietary interests in any material discussed in this article.

References

- Almansor EH, Al-Ani A (2018) A hybrid neural machine translation technique for translating low resource languages. In: Perner P (ed) Machine learning and data mining in pattern recognition. Springer International Publishing, Cham, pp 347–356
- Artetxe M, Labaka G, Agirre E (2018) Unsupervised statistical machine translation. In: Proceedings of the 2018 conference on empirical methods in natural language processing, Association for Computational Linguistics, Brussels, Belgium, pp 3632–3642. <https://doi.org/10.18653/v1/D18-1399>
- Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. ArXiv 1409
- Bal S, Mahanta S, Mandal L, Parekh R (2019) Bilingual machine translation: English to bengali. In: Chakraborty M, Chakrabarti S, Balas VE, Mandal JK (eds) Proceedings of international ethical hacking conference 2018, Springer Singapore, Singapore, pp 247–259
- Banerjee S, Lavie A (2005) METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, Association for Computational Linguistics, Ann Arbor, Michigan, pp 65–72. <https://www.aclweb.org/anthology/W05-0909>
- Bojar O, Federmann C, Fishel M, Graham Y, Haddow B, Koehn P, Monz C (2018) Findings of the 2018 conference on machine translation (WMT18). In: Proceedings of the third conference on machine translation: shared task papers, Association for Computational Linguistics, Belgium, Brussels, pp 272–303. <https://doi.org/10.18653/v1/W18-6401>
- Dasgupta S, Wasif A, Azam S (2004) An optimal way towards machine translation from english to bengali. In: Proceedings of the international conference on computer and information technology (ICCIT), IEEE, Dhaka, Bangladesh, pp 648–653
- Farzad A, Mashayekhi H, Hassanpour H (2019) A comparative performance analysis of different activation functions in lstm networks for classification. Neural Comput Appl. <https://doi.org/10.1007/s00521-017-3210-6>
- Gangadharaiah R, Brown RD, Carbonell J (2011) Phrasal equivalence classes for generalized corpus-based machine translation. In: Gelbukh A (ed) Proceedings of the international conference on intelligent text processing and computational linguistics: computational linguistics and intelligent text processing, Springer, Berlin, pp 13–28
- Gu J, Hassan H, Devlin J, Li VO (2018) Universal neural machine translation for extremely low resource languages. In: Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics: human language technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, pp 344–354. <https://doi.org/10.18653/v1/N18-1032>

11. Hasan HMM, Islam MA (2020) Emotion recognition from bengali speech using rnn modulation-based categorization. In: 2020 third international conference on smart systems and inventive technology (ICSSIT), pp 1131–1136. <https://doi.org/10.1109/ICSSIT48917.2020.9214196>
12. Hasan HMM, Islam MA, Hasan MT, Hasan MA, Rumman SI, Shakib MN (2020) A spell-checker integrated machine learning based solution for speech to text conversion. In: 2020 third international conference on smart systems and inventive technology (ICSSIT), pp 1124–1130. <https://doi.org/10.1109/ICSSIT48917.2020.9214205>
13. Islam MA, Islam ABMAA (2016) Polygot: Going beyond database driven and syntax-based translation. In: Proceedings of the 7th annual symposium on computing for development, ACM, Nairobi, Kenya, pp 28:1–28:4. <https://doi.org/10.1145/3001913.3006637>
14. Islam MA, Islam ABMAA, Anik MSH (2017) Polygot: an approach towards reliable translation by name identification and memory optimization using semantic analysis. In: 4th international conference on networking, systems and security (NSysS), Dhaka, Bangladesh, pp 1–8
15. Islam Z, Tiedemann J, Eisele A (2010) English to bangla phrase-based machine translation
16. Kann K, Cho K, Bowman SR (2019) Towards realistic practices in low-resource natural language processing: the development set. Computing Research Repository (CoRR). [arXiv:1909.01522](https://arxiv.org/abs/1909.01522)
17. Kim JD, Brown RD, Carbonell JG (2010) Chunk-based ebmt. In: Proceedings of the 14th annual conference of european association for machine translation, Figshare, St Raphael, France, pp 1–8
18. Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: Bengio Y, LeCun Y (eds) 3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
19. Koehn P, Knowles R (2017) Six challenges for neural machine translation. Computing Research Repository (CoRR). [arXiv:1706.03872](https://arxiv.org/abs/1706.03872)
20. Lample G, Denoyer L, Ranzato M (2017) Unsupervised machine translation using monolingual corpora only. Computing Research Repository (CoRR). [arXiv:1711.00043](https://arxiv.org/abs/1711.00043)
21. Luong M, Brevdo E, Zhao R (2017) Neural machine translation (seq2seq) tutorial. <https://github.com/tensorflow/nmt>
22. Luong M, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. [arXiv:1508.04025](https://arxiv.org/abs/1508.04025)
23. Mandelbaum A, Shalev A (2016) Word embeddings and their use in sentence classification tasks. [arXiv:1610.08229](https://arxiv.org/abs/1610.08229)
24. Mumin M, Seddiqui M, Iqbal M, Islam MJ (2019) shu-torjoma: an english-bangla statistical machine translation system. *J Comput Sci* 15:1022–1039. <https://doi.org/10.3844/jcssp.2019.1022.1039>
25. Naskar SK, Bandyopadhyay S (2005) A phrasal ebmt system for translating english to bengali. In: Proceedings of the 10th MT Summit, AAMT, Phuket, Thailand, pp 372–379
26. Pal S (2018) A hybrid machine translation framework for an improved translation workflow
27. Papineni K, Roukos S, Ward T, Zhu WJ (2002) Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the association for computational linguistics, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, pp 311–318. <https://doi.org/10.3115/1073083.1073135>. <https://www.aclweb.org/anthology/P02-1040>
28. Powers DMW (1998) Applications and explanations of Zipf's law. In: New methods in language processing and computational natural language learning. <https://www.aclweb.org/anthology/W98-1218>
29. Rahman F, Begum M, Islam A, Khan H, Mahanaz S, Hossain Z, Islam A (2019) An annotated bangla sentiment analysis corpus. In: International conference on bangla speech and language processing (ICBSLP), 27:28
30. Rahman M, Kabir MF, Huda M (2019) A corpus based n-gram hybrid approach of bengali to english machine translation. In: Proceedings of the international conference on computer and information technology (ICCI), IEEE, Dhaka, Bangladesh, pp 1–6. <https://doi.org/10.1109/ICCITECHN.2018.8631938>
31. Rahman MK, Tarannum N (2012) A rule based approach for implementation of bangla to english translation. In: Proceedings of the 2012 international conference on advanced computer science applications and technologies (ACSAT), IEEE, Kuala Lumpur, Malaysia, pp 13–18. <https://doi.org/10.1109/ACSAT.2012.98>
32. Rahman MS (2011) Translation of three different texts types from english to bengali. Ph.D. thesis, East West University
33. Roy M (2009) A semi-supervised approach to bengali-english phrase-based statistical machine translation. In: Gao Y, Japkowicz N (eds) Proceedings of the Canadian conference on artificial intelligence: advances in artificial intelligence, Springer, Berlin, pp 291–294
34. Ruder S (2016) An overview of gradient descent optimization algorithms. [arXiv:1609.04747](https://arxiv.org/abs/1609.04747)
35. Saha D (2005) A semantics-based english-bengali ebmt system for translating news headlines. In: Proceedings of the 10th MT Summit, AAMT, Phuket, Thailand, pp 125–133
36. Sakiba SN, Shuvo MMU, Hossain N, Das SK, Mela JD, Islam MA (2020) A memory-efficient tool for bengali parts of speech tagging. In: Hemanth D, Vadivu G, Sangeetha M, Balas V (eds) Artificial intelligence techniques for advanced computing applications, vol 131. Springer, Berlin, pp 67–78
37. Sennrich R, Haddow B, Birch A (2015) Improving neural machine translation models with monolingual data. [arXiv:1511.06709](https://arxiv.org/abs/1511.06709)
38. Sennrich R, Haddow B, Birch A (2015) Neural machine translation of rare words with subword units. [arXiv:1508.07909](https://arxiv.org/abs/1508.07909)
39. Snover M, Dorr B, Schwartz R, Micciulla L, Makhoul J (2006) A study of translation edit rate with targeted human annotation. In: Proceedings of association for machine translation in the americas, AMTA, Cambridge, America, pp 223–231
40. Wang X, Liu Y, Sun CJ, Wang B, Wang X (2015) Predicting polarities of tweets by composing word embeddings with long short-term memory. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (Volume 1: Long Papers), pp 1343–1353
41. Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J, Shah A, Johnson M, Liu X, Kaiser L, Gouws S, Kato Y, Kudo T, Kazawa H, Stevens K, Kurian G, Patil N, Wang W, Young C, Smith J, Riesa J, Rudnick A, Vinyals O, Corrado G, Hughes M, Dean J (2016) Google's neural machine translation system: bridging the gap between human and machine translation. [arxiv:1609.08144](https://arxiv.org/abs/1609.08144)