

# Low Resource Multimodal Neural Machine Translation of English-Hindi in News Domain

Loitongbam Sanayai Meetei<sup>1</sup>, Thoudam Doren Singh<sup>1</sup>, and Sivaji Bandyopadhyay<sup>1</sup>

<sup>1</sup>Centre for Natural Language Processing (CNLP) & Dept. of CSE, NIT Silchar, India  
{loisanayai,thoudam.doren,sivaji.cse.ju}@gmail.com

## Abstract

Incorporating multiple input modalities in a machine translation (MT) system is gaining popularity among MT researchers. Unlike the publicly available dataset for Multimodal Machine Translation (MMT) tasks, where the captions are short image descriptions, the news captions provide a more detailed description of the contents of the images. As a result, numerous named entities relating to specific persons, locations, etc., are found. In this paper, we acquire two monolingual news datasets reported in English and Hindi paired with the images to generate a synthetic English-Hindi parallel corpus. The parallel corpus is used to train the English-Hindi Neural Machine Translation (NMT) and an English-Hindi MMT system by incorporating the image feature paired with the corresponding parallel corpus. We also conduct a systematic analysis to evaluate the English-Hindi MT systems with 1) more synthetic data and 2) by adding back-translated data. Our finding shows improvement in terms of BLEU scores for both the NMT (+8.05) and MMT (+11.03) systems.

## 1 Introduction

With the implementation of encoder-decoder architecture (Cho et al., 2014; Luong et al., 2015; Vaswani et al., 2017), MT systems have undergone quality enhancement. Instead of using text as the only input in an MT system, the current trend has also started exploring Multimodal Machine Translation (MMT), where multiple input modalities such as visual modality are incorporated along with the text as an input to the MT system. Using the MMT system has shown improvement in the translated text output as compared to the NMT system (Huang et al., 2016; Caglayan et al., 2016; Elliott and Kádár, 2017; Caglayan et al., 2019). To analyse the benefits of using multiple modalities of input, various shared tasks

are organized (WAT2019 Multi-Modal Translation Task<sup>1</sup>, WMT2018<sup>2</sup>, VMT Challenge<sup>3</sup>). However, the image descriptions in the majority of current datasets are made up of user-captioned or created by crowdsourcing. On the other hand, the captions present in the news details the contents of the image with better clarity, and as a result, contain many named entities relating to specific individuals, locations, organizations, etc. For example, in Figure 1, the caption “*The old looking ship is sailing at sunset*” correctly depicts the image on some levels, yet it fails to portray the picture’s higher-level scenario as described in the caption on the left.

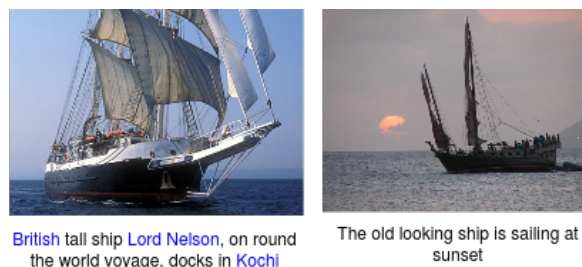


Figure 1: Examples from our dataset (left) and multi30k (right).

Limited data resources set back the development of a machine learning-based system. The lack of high-quality training parallel dataset poses a considerable challenge in developing an MT system for low resource languages. For an extremely low resource language pair, training with NMT, which is a data-driven approach, often reports to poor performance of the MT system (Singh and Hujon, 2020; Singh and Singh, 2020). As such, re-

<sup>1</sup><https://ufal.mff.cuni.cz/hindi-visual-genome/wat-2019-multimodal-task>

<sup>2</sup><http://www.statmt.org/wmt18/multimodal-task.html>

<sup>3</sup>[https://eric-xw.github.io/vatex-website/translation\\_2020.html](https://eric-xw.github.io/vatex-website/translation_2020.html)

searchers have investigated various models to augment the dataset using the monolingual corpus, such as back-translation (Sennrich et al., 2015a), incorporating language model train on monolingual dataset (Gulcehre et al., 2015), etc. The approach reported in (Sennrich et al., 2015a; Calixto et al., 2017a) acquire an additional training dataset by back-translating from a monolingual target dataset. In this paper, we acquire monolingual news datasets reported in English and Hindi, which are used to generate a synthetic parallel corpus. English→Hindi NMT systems are trained by using the parallel corpus. We train English→Hindi MMT systems by incorporating the image as a feature paired with the corresponding parallel corpus. We also conduct a systematic analysis to evaluate the MT systems by training with 1) more synthetic data and 2) adding back-translated data. Belonging to the same language family, Indo-European, English, and Hindi follow different word orders: Subject Verb Object (SVO) and Subject Object Verb (SOV).

The remainder of this paper is organized as follows: Section 2 discuss the previous related works followed by the framework of our model in Section 3. Section 4 details our system set up and Section 5 illustrates the analysis of our result. Section 6 sums up the conclusion and future works.

## 2 Related Works

A review of the machine translation-related works is discussed in this section. Based on the encoder-decoder model of the NMT model, various architectures are built to improve the performance of MT systems (Sutskever et al., 2014; Bahdanau et al., 2014). For both the encoder and the decoder, Sutskever et al. (2014) stacked numerous layers of an RNN with a Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) hidden unit. Bahdanau et al. (2014) introduced an attention mechanism where the decoder attends to various sections of the source text at each step of generation of the output. While the model enhances the translation of long sentences due to its sequential nature, each hidden state depends on the output of the previous hidden state resulting in a large consumption of computational power.

Gage (1994) introduced a method for data compression, BPE, which iteratively substitutes single, unused bytes for common bytes pairs in a sequence. Sennrich et al. (2015b) proposed a method

for word segmentation to deal with an open vocabulary problem. Instead of common byte pairs, the method combines characters or character sequences. Provilkov et al. (2019) achieves a better MT system by introducing a dropout in BPE (Sennrich et al., 2015b), the BPE-dropout excludes some merges randomly, resulting in the same word with different segmentation.

Vinyals et al. (2015) introduced a neural and probabilistic framework to generate image captions. The model comprises a vision Convolution Neural Network (CNN), which is followed by a Recurrent Neural Network (RNN) to generate a language. Extracting global features from an image to incorporate into attention-based NMT, Calixto et al. (2017b) introduced various multimodal neural machine translation models. Using the features of an image to initialize the encoder hidden state is reported to be the best performing among other models. Using Hindi Visual Genome (Parida et al., 2019) dataset, Meetei et al. (2019) carried out an MMT for the English-Hindi language pair. The author reported that the use of multiple modalities as an input improves the MT system.

Various approaches to data augmentation are applied to mitigate the scarcity of parallel training datasets for MT tasks. Gulcehre et al. (2015) used a language model train on a monolingual dataset, achieving an improvement of up to 1.96 BLEU on Turkish-English, a low resource language pair. The author also reported that domain similarity between monolingual dataset and target task was the key factor to use an external language model to improve the MT system. Sennrich et al. (2015a) carried out the back-translation of monolingual target text into the source text, thereby generating an additional training dataset. The author reported that even the limited amount of back-translated in-domain monolingual datasets could be utilized efficiently for domain adaptation. Calixto et al. (2017a) used a text-only NMT model train on Multi30k (Elliott et al., 2016) dataset (German-English), without images to back-translate German descriptions in the Multi30k into English and included it as additional training data.

## 3 Methodology

In our experiment, news articles reported in English and Hindi along with the corresponding images in the articles are collected. After subjecting to a pre-processing step, the collected dataset is

	sentences	images
$en-hi_{st}$	80900	80900
$hi-en_{st}$	42400	42400

Table 1: Machine translated datasets,  $en-hi_{st}$  and  $hi-en_{st}$

machine translated to generate a synthetic parallel dataset. MT systems are trained with various settings by using the synthetic parallel dataset.

### 3.1 Building synthetic English-Hindi and Hindi-English dataset

News articles reported in English along with their corresponding images are collected from a national news channel, *India TV*<sup>4</sup>, for the period June 2010 to May 2020. After filtering the articles where the image is absent, the collected dataset comprises 80900 and 42400 news articles reported in English and Hindi, respectively. The dataset is gathered by utilizing a web-scraper built in-house. In order to prepare the experimental dataset, we separate the headline from each of the news article items, which is considered as the description for the corresponding image. Apart from a standard single sentence, the image description comprises single or multiple phrases. Using IndicTrans<sup>5</sup> (Kakwani et al., 2020), which is an NMT system, we build two machine translated parallel datasets, namely English-Hindi ( $en-hi_{st}$ ) and Hindi-English ( $hi-en_{st}$ ), Table 1.

### 3.2 Machine Translation Systems

Our experiment used both NMT and MMT approaches to train English→Hindi MT systems in our parallel news dataset.

#### 3.2.1 Neural Machine Translation (NMT)

For a source sentence  $x$ , the translation task tries to find a target sentence  $y$  that maximizes the conditional probability of  $y$  given  $x$ . We followed the attention model of Bahdanau et al. (2014) by using a bi-LSTM (Sutskever et al., 2014) in the encoder and an alignment model paired with an LSTM in the decoder model. The bi-LSTM generates a sequence of annotations  $(h_1, h_2, \dots, h_N) = h_i$  for each input sentence,  $x = (x_1, x_2, \dots, x_N)$ .  $h_i = [\vec{h}_i; \overleftarrow{h}_i]$  is the concatenation of forward hidden state,  $\vec{h}_i$  and backward hidden state,  $\overleftarrow{h}_i$  in the encoder at time

<sup>4</sup><https://www.indiatvnews.com/>

<sup>5</sup><https://indicnlp.ai4bharat.org/indic-trans/>

step  $i$ . The attention mechanism focuses on specific input vectors in the input sequence based on the attention weights.

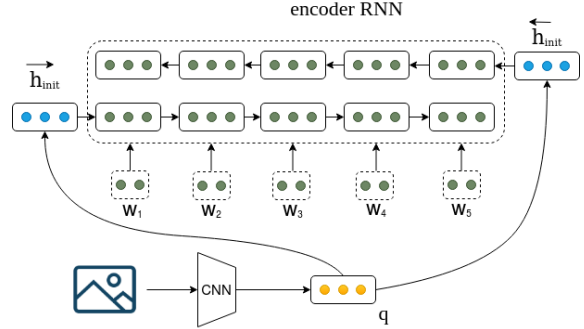


Figure 2: MMT model

#### 3.2.2 Multimodal Machine Translation (MMT)

We train the MMT systems using the image from the news article paired with the English-Hindi parallel dataset. Following the multimodal neural machine translation (MNMT) model (Calixto et al., 2017b), a deep CNN-based model is utilized to extract global features from the image, Figure 2. Global image feature vector ( $q \in \mathbb{R}^{4096}$ ) is used to compute a vector  $d$  as follows:

$$d = W_I^2 \cdot (W_I^1 \cdot q + b_I^1) + b_I^2 \quad (1)$$

where  $W$  = image transformation matrices and  $b$  = bias vector.

Instead of using  $\vec{0}$  (Bahdanau et al., 2014) to initialize encoder hidden states, two new single-layer feed-forward networks are utilized to initialize the states as:

$$\vec{h}_{init} = \tanh(W_f d + b_f) \quad (2)$$

$$\overleftarrow{h}_{init} = \tanh(W_b d + b_b) \quad (3)$$

where  $W_f$  and  $W_b$  are the multi-modal projection matrices that project the image features  $d$  into the encoder forward and backward hidden states dimensionality, respectively, and  $b_f$  and  $b_b$  as bias vectors.

## 4 Experimental Design

### 4.1 Dataset

With limited availability of parallel corpus, it is often difficult to train a data driven NMT system. Using the generated synthetic parallel dataset, we

	<b>Text-Image</b>	<b>types</b>	<b>unique types</b>	<i>Avg<sub>SL</sub></i>
train ( $T_b$ )	45000	<i>en</i> :472496, <i>hi</i> :548172	<i>en</i> :52841, <i>hi</i> :33617	<i>en</i> :10, <i>hi</i> :12
train ( $T_{ad}$ )	$T_b + 30900$	<i>en</i> :796074, <i>hi</i> :923313	<i>en</i> :70371, <i>hi</i> :43945	<i>en</i> :10, <i>hi</i> :12
train ( $T_{bt}$ )	$T_b + 30900$	<i>en</i> :806767, <i>hi</i> :1005549	<i>en</i> :64057, <i>hi</i> :52992	<i>en</i> :10, <i>hi</i> :13
train ( $T_{all}$ )	$T_b + 61800$	<i>en</i> :1130335, <i>hi</i> :1380674	<i>en</i> :79574, <i>hi</i> :61419	<i>en</i> :10, <i>hi</i> :12
dev	3000	<i>en</i> :31437, <i>hi</i> :36576	<i>en</i> :10673, <i>hi</i> :7871	<i>en</i> :10, <i>hi</i> :12
test ( $t_1$ )	2000	<i>en</i> :21038, <i>hi</i> :24415	<i>en</i> :8089, <i>hi</i> :6176	<i>en</i> :10, <i>hi</i> :12
test ( $t_2$ )	2000	<i>en</i> :17105, <i>hi</i> :20854	<i>en</i> :5439, <i>hi</i> :5874	<i>en</i> :8, <i>hi</i> :10

Table 2: Statistics of our dataset and data partitioning.  
*en*: English, *hi*: Hindi, *Avg<sub>SL</sub>*: average sentence length

carry out a systematic analysis to evaluate the MT system by training with four experimental data settings.

- $T_b$ : By randomly selecting 45000 parallel dataset from *en-hi<sub>st</sub>* as the baseline training dataset.
- $T_{ad}$ :  $T_b$  + an additional dataset of randomly selected 30900 from the remaining *en-hi<sub>st</sub>* dataset.
- $T_{bt}$ :  $T_b$  + an additional back-translated dataset of 30900 which are randomly selected from 42400 sentences (*hi-en<sub>st</sub>*).
- $T_{all}$ : Combining the above three training datasets i.e.  $T_b + T_{ad} + T_{bt}$ .

We use two holdouts test datasets,  $t_1$  and  $t_2$  from *en-hi<sub>st</sub>* and *hi-en<sub>st</sub>* (back-translated) respectively. The development dataset, however, is used from *en-hi<sub>st</sub>* only. A detailed statistics of our dataset is shown in Table 2 where *en-hi<sub>st</sub>* is split into training, development, and test datasets, whereas *hi-en<sub>st</sub>* is used for training and test datasets.

Normalization and tokenization of English sentences are carried by using Koehn et al. (2007) and for Hindi sentences, we use *Indic NLP*<sup>6</sup>. By employing BPE-dropout (Provilkov et al., 2019), words in the pre-processed parallel corpus are segment into subword units for word embedding presentation before training the MT systems. A full regularization is applied with a dropout of 0.1 to the training dataset. Following the system design described in SubSection 3.2, we train our NMT and MMT systems using the processed dataset.

<sup>6</sup>[https://anoopkunchukuttan.github.io/indic\\_nlp\\_library/](https://anoopkunchukuttan.github.io/indic_nlp_library/)

## 4.2 MT systems

Based on the four settings of training dataset, we train the following eight MT systems:

- NMT( $T_b$ ) and MMT( $T_b$ ): NMT and MMT systems trained with  $T_b$  respectively, *baseline models*.
- NMT( $T_{ad}$ ) and MMT( $T_{ad}$ ): NMT and MMT systems trained with  $T_{ad}$  respectively.
- NMT( $T_{bt}$ ) and MMT( $T_{bt}$ ): NMT and MMT systems trained with  $T_{bt}$  respectively.
- NMT( $T_{all}$ ) and MMT( $T_{all}$ ): NMT and MMT systems trained with  $T_{all}$  respectively.

## 4.3 NMT system settings

The size of our encoder and decoder LSTM hidden states is set to 512. We use a batch size of 128 and a word embedding size of 512D for both source and target. The normalization method of the gradient is set to tokens. Along with other parameters such as learning rate at 0.01, Adam optimizer (Kingma and Ba, 2014), a dropout rate of 0.1, we train the system using early stopping, where training is stopped if a model does not progress on the validation set for more than 10 epochs.

## 4.4 MMT system settings

A CNN-based pre-trained model, VGG19 (Simonyan and Zisserman, 2014), is used to extract the global features of an image. By incorporating the features from the image and the processed text, we train our MMT systems with stochastic gradient descent and a batch size of 128. Early stopping is applied to stop the training when the MT system does not improve for 10 epochs on the development set. We carry out the implementation of our MT systems by using an NMT open-source tool

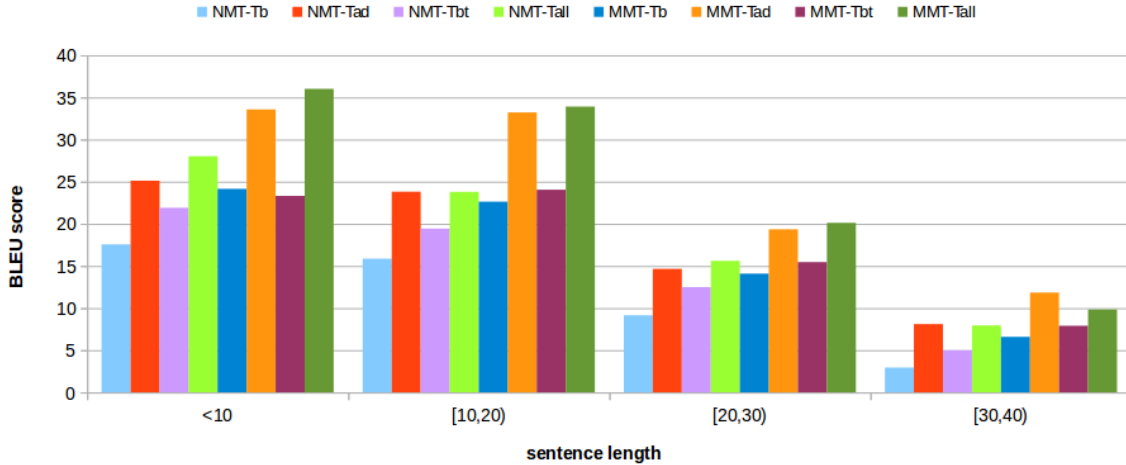


Figure 3: Evaluation on test dataset  $t_1$

		BLEU	
	train	$t_1$	$t_2$
NMT	$T_b$	15.56	8.96
	$T_{ad}$	23.26(↑7.7)	13.47(↑4.51)
	$T_{bt}$	19.23(↑3.67)	11.70(↑2.74)
	$T_{all}$	23.61(↑8.05)	13.75(↑4.79)
MMT	$T_b$	22.20	14.54
	$T_{ad}$	32.30(↑10.1)	17.13(↑2.59)
	$T_{bt}$	23.43(↑1.23)	17.12(↑2.58)
	$T_{all}$	<b>33.23(↑11.03)</b>	<b>18.81(↑4.27)</b>

Table 3: Evaluation of NMT and MMT systems in terms of BLEU score.

based on OpenNMT (Klein et al., 2017). Subword-nmt<sup>7</sup> is used for encoding-decoding of the text dataset to and from subword units.

## 5 Results and Analysis

### 5.1 Based on Evaluation Metric

The automatic evaluation of our MT systems is reported using BLEU (Papineni et al., 2002). Table 3 shows a detailed evaluation of our MT systems on two test datasets,  $t_1$  and  $t_2$ .

- **NMT systems:** NMT( $T_{all}$ ) outperforms the remaining NMT systems in both the test datasets,  $t_1$  and  $t_2$ .
- **MMT systems:** MMT( $T_{all}$ ) outperforms the MMT systems in both the test datasets.
- **NMT vs MMT System:** The best MMT system, MMT( $T_{all}$ ) outperforms the best NMT system, NMT( $T_{all}$ ) by up to 9.62 BLEU score in  $t_1$  and up to 5.34 BLEU score in  $t_2$ .

<sup>7</sup><https://github.com/rseennrich/subword-nmt>

Training with additional datasets shows improvements in terms of BLEU scores for both the NMT and MMT systems. Although improvements are observed with the MT systems trained with the data augmentation approach, the BLEU score increases only by 1.23 while training with the additional back-translated dataset,  $T_{bt}$  for the MMT system. This shows that using an additional back-translated dataset improves our MMT system only by a small margin. It is observed that the performance of NMT( $T_{ad}$ ) and NMT( $T_{all}$ ) are almost comparable in terms of BLEU score, which indicates the poor effectiveness of  $T_{bt}$  in our experimental settings. Whereas, in the MMT system, MMT( $T_{all}$ ) outperforms the other MMT systems in terms of BLEU score by a reasonable margin. This indicates that incorporating image features in the MT system negates the bias introduced by the synthetic dataset to some extent. Furthermore, a large gap in terms of BLEU score is observed between  $t_1$  and  $t_2$ . A likely cause is using more training dataset from  $en-hi_{st}$  and the development dataset from  $en-hi_{st}$  only.

**Bucket Analysis:** Figure 3 and Figure 4 shows bucket analysis where salient statistics are computed by assigning sentences over the bucket. After computing the BLEU score based on the length of the reference sentence, the analysis displays how well a system performs with shorter and longer sentences.

- $t_1$ : Sentences in  $t_1$  dataset are grouped into four buckets as shown in Figure 3. MMT( $T_{all}$ ) outperforms all the other MT systems by a large margin in most of the cases i.e. sentences with length less than 30. Although

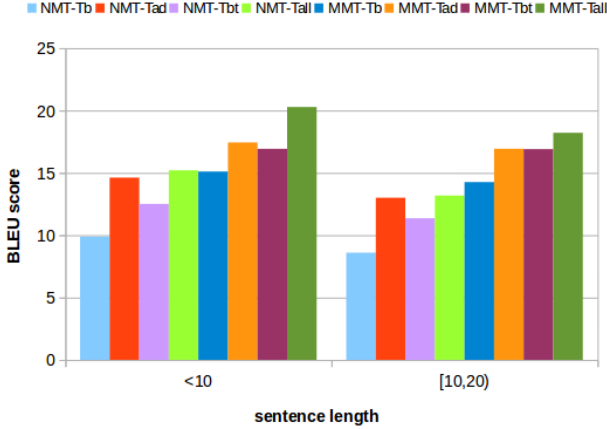


Figure 4: Evaluation on test dataset  $t_2$

Score	Level Interpretation
0	No information is retained
1	Small amount of information is retained
2	Moderately retained information
3	Most of the information is retained
4	All information is retained

Table 4: Scale for Adequacy score.

the overall BLEU score of  $MMT(T_{bt})$  is better than  $MMT(T_b)$ ,  $MMT(T_b)$  is observed to perform better than  $MMT(T_{bt})$  when the sentence length is less than 10.

- $t_2$ : As the maximum length of a sentence in the  $t_2$  dataset is not more than 20,  $t_2$  is grouped into two buckets, Figure 4.  $MMT(T_{all})$  outperforms the other MT systems by a large margin irrespective of the sentence length.  $MMT(T_{ad})$  is almost comparable with  $MMT(T_{bt})$  in sentences with length [10-20).

Overall, the MT systems performs better when the length of a sentence is up to 10, and the performance declines as the length of a sentence increase above 10.

## 5.2 Adequacy and Fluency Analysis

Using adequacy and fluency indicators, we carried out human evaluations on our machine translated outputs. Adequacy indicates information retained in the generated translations, whereas fluency analyses generated translations primarily on grammatical rules. In our experiment, both adequacy and fluency are computed in the range of 0 to 4 scores. The meanings of the various score are summarized

Score	Level Interpretation
0	Incomprehensible
1	Disfluent
2	Non-native
3	Acceptable in terms of grammatical rules
4	Flawless and correct in terms of grammatical rules

Table 5: Scale for Fluency score.

	train	Adequacy	Fluency
NMT	$T_b$	1	1.5
	$T_{ad}$	1.55	2.1
	$T_{bt}$	1.35	1.85
	$T_{all}$	1.4	1.77
MMT	$T_b$	1.4	1.92
	$T_{ad}$	<b>1.95</b>	<b>2.35</b>
	$T_{bt}$	1.68	1.87
	$T_{all}$	1.8	2.1

Table 6: Evaluation of NMT and MMT systems in terms of Adequacy and Fluency score.

in Table 4 and Table 5. We use a sample output of randomly selected 100 sentences from each MT system to evaluate adequacy and fluency scores. The average score of the individual MT system is considered as our final score.

Table 6 shows the adequacy and fluency scores reported by our human evaluators. Comparison among the different NMT systems indicates no correlation between the manual evaluation and BLEU score. In the NMT system, adding a back-translated dataset in the  $T_{all}$  setting shows a negative effect in the fluency score, with  $NMT(T_{all})$  scoring less than the  $NMT(T_{bt})$ . Similar observation is also observe among the different MMT systems. The adequacy and fluency score of  $MMT(T_{ad})$  shows better results than  $MMT(T_{all})$  by a small margin. When comparing between the NMT and MMT systems, correlation is found between the manual evaluation and BLEU score. Overall, in terms of adequacy and fluency score, the MMT system is more robust than the NMT system.

## 5.3 Translation Analysis

Table 7 and Table 8 shows a qualitative analysis on two examples from the test dataset ( $t_1$ ) for the NMT and MMT systems. The words in “blue” highlights incorrect word(s) or gram-


Input	
<b>Src:</b> Pune: One nurse tests COVID-19 positive, 30 others quarantined	
<b>Ref:</b> पुणे में एक नर्स कोविड-१९ पॉजिटिव, ३० अन्य पृथक-वास में (pune mein ek nurse covid-19 positive, 30 any prthak-vaas mein)	
NMT Model Outputs	
$T_b$	पुणे में एक महीने का कोविड-१९ पॉजिटिव पॉजिटिव आया, ३० अन्य लोगों को क्वारंटीन किया गया (pune mein ek maheene ka covid-19 positive positive anya, 30 anya logon ko quarantine kiya gaya) “One month of Covid-19 positive positive others came in Pune, 30 others were quarantined”
$T_{ad}$	पुणे में एक व्यक्ति कोविड-१९ से संक्रमित, ३० अन्य लोगों को क्वारंटीन किया गया (pune mein ek vyakti covid-19 se sankramit, 30 anya logon ko quarantine kiya gaya) “One person infected with Covid-19 in Pune, 30 others have been quarantined”
$T_{bt}$	पुणे में एक कोविड-१९ से संक्रमित पाए गए कोविड-१९ से संक्रमित (pune mein ek covid-19 se sankramit pae gae covid-19 se sankramit) “One found infected with Covid-19 in Pune infected with Covid-19”
$T_{all}$	पुणे में एक नर्स कोविड-१९ पॉजिटिव पाए गए (pune mein ek nurse covid-19 positive pae gae ) “One nurse in Pune found to be Covid-19 positive”
MMT Model Outputs	
$T_b$	पुणे में एक नर्स कोविड-१९ से संक्रमित, ३० अन्य संक्रमित (pune mein ek nurse Covid-19 se sankramit, 30 anya sankramit) “One nurse infected with Covid-19 in Pune, 30 others <b>infected</b> ”
$T_{ad}$	पुणे: एक नर्स का कोविड-१९ टेस्ट पॉजिटिव आया, ३० अन्य लोग क्वारंटीन (pune: ek nurse ka Covid-19 test positive aaya, 30 anya log quarantine) “Pune: One nurse tested positive for COVID-19, 30 others in quarantine”
$T_{bt}$	पुणे में एक नर्स कोविड-१९ से संक्रमित, ३० अन्य को क्वारंटीन किया गया (pune mein ek nurse covid-19 se sankramit, 30 anya ko quarantine kiya gaya) “One nurse infected with Covid-19 in Pune, 30 others were quarantined”
$T_{all}$	पुणे में नर्स का कोविड-१९ टेस्ट पॉजिटिव आया, ३० अन्य को क्वारंटीन किया गया (pune mein nurse ka covid-19 test positive aya, 30 anya ko quarantine kiya gaya) “Covid-19 test of a nurse came out positive in Pune, 30 others were quarantined”

Table 7: Input Output sample 1

matically error in the translation output whereas, words in “magenta” highlights incorrectly translated word(s).  $NMT(T_b)$  and  $NMT(T_{bt})$  generate translations with low fluency, where part of the sentence is grammatically incorrect, which in turn affects the adequacy of the translated text. Though the  $MMT(T_b)$  and  $MMT(T_{bt})$  generates a translation with good fluency, the model fails to convey the words like “quarantined”, “landslide” and instead translates as “quarantined” → “infected”, “landslide” → “disturbance”, and “landslide” → “collision” thereby reducing the adequacy of the translated text. As reported in the adequacy and

fluency evaluation in Table 6,  $T_{all}$  performs poorly in the NMT system with the translation output missing part of the source sentence as shown in the sample examples.  $MMT(T_{ad})$  and  $NMT(T_{all})$  generate translations that are grammatically correct and convey correct meaning as the input sentence.

## 6 Conclusion and Future Work

The lack of a high-quality parallel dataset for the MT tasks is one of the major challenges, especially for low resource languages. In this work, we collected two monolingual news datasets re-

Input	
<b>Src:</b>	After major landslide on Mumbai-Pune Expressway, Railways announces special
<b>Ref:</b>	मुंबई-पुणे एक्सप्रेसवे पर भूस्खलन के बाद रेलवे ने विशेष ट्रेनों की घोषणा की (mumbai-pune expressway par bhooskhalan ke baad railway ne vishesh trenon kee ghoshana kee)
	
NMT Model Outputs	
$T_b$	पुणे-पुणे एक्सप्रेसवे पर भगदड़, रेलवे ने स्पेशल स्पेशल ट्रेनें रद्द की (pune-pune expressway par bhagadad, railway ne special special trainon radd kee) “Stampede on Pune-Pune Expressway, Railways canceled special special trains”
$T_{ad}$	मुंबई-पुणे एक्सप्रेसवे पर भूस्खलन के बाद रेलवे ने विशेष ट्रेनों की घोषणा की (mumbai-pune expressway par bhooskhalan ke baad railway ne vishesh trainon kee ghoshana kee) “Railways announce special trains after landslide on Mumbai-Pune Expressway”
$T_{bt}$	पुणे-वे पर भूस्खलन के बाद रेलवे ने जारी की स्पेशल ट्रेनें (pune-ve par bhooskhalan ke baad railway ne jaaree kee special trainon) “Railways issued special trains after landslide on Pune-way”
$T_{all}$	मुंबई-पुणे एक्सप्रेसवे पर भूस्खलन के बाद भूस्खलन के बाद रेलवे की विशेष ट्रेनें (mumbai-pune expressway par bhooskhalan ke baad bhooskhalan ke baad railway kee vishesh trainon) “Railway special trains after landslide on Mumbai-Pune Expressway”
MMT Model Outputs	
$T_b$	एक्सप्रेसवे पर बड़ी गड़बड़ी के बाद रेलवे ने विशेष ट्रेनों की घोषणा की (expressway par badee gadabadee ke baad railway ne vishesh trainon kee ghoshana kee) “Railway announces special trains after major disturbance on Expressway”
$T_{ad}$	मुंबई-पुणे एक्सप्रेसवे पर भूस्खलन, रेलवे ने स्पेशल ट्रेनों की घोषणा की (mumbai-pune expressway par bhooskhalan, railway ne special trainon kee ghoshana kee) “Landslide on Mumbai-Pune Expressway, Railways announces special trains”
$T_{bt}$	रेलवे एक्सप्रेस एक्सप्रेसवे पर बड़ा टक्कर, रेलवे ने स्पेशल ट्रेनों की घोषणा की (railway express expressway par bada takkar, railway ne special trainon kee ghoshana kee) “Major collision on Railway Express Expressway, Railways announced special trains”
$T_{all}$	मुंबई-पुणे एक्सप्रेसवे पर बड़ा भूस्खलन, रेलवे ने की विशेष ट्रेनों की घोषणा (mumbai-pune expressway par bada bhooskhalan, railway ne kee vishesh trainon kee ghoshana) “Major landslide on Mumbai-Pune Expressway, Railways announces special trains”

Table 8: Input Output sample 2

ported in English and Hindi paired with the images to create a synthetic English-Hindi parallel corpus. A systematic analysis of English→Hindi NMT and English→Hindi MMT systems with various experimental datasets set up is conducted. An analysis of the dataset augmentation with the lack of a parallel dataset is carried out. We observe an improvement in the MT systems in BLEU scores for both the NMT and MMT systems with the data augmentation approach. Our results also show that when the training dataset is comprised of a synthetic dataset from both English→Hindi and Hindi→English directions, the back-translated dataset in  $T_{all}$  setting is more effective in the MMT

system as compared to the NMT system. In the future, we would like to incorporate multiple modalities to improve the MT system.

### Acknowledgments

This work is supported by Scheme for Promotion of Academic and Research Collaboration (SPARC) Project Code: P995 of No: SPARC/2018-2019/119/SL(IN) under Ministry of Education (erstwhile MHRD), Govt. of India. The authors thank the anonymous reviewers for their careful reading and their many insightful comments. The authors also thank the volunteers for their help in human evaluation tasks.



## References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Ozan Caglayan, Walid Aransa, Yaxing Wang, Marc Masana, Mercedes Garcia-Martínez, Fethi Bougares, Loïc Barrault, and Joost Van de Weijer. 2016. Does multimodality help human and machine for translation and image captioning? *arXiv preprint arXiv:1605.09186*.
- Ozan Caglayan, Pranava Madhyastha, Lucia Specia, and Loïc Barrault. 2019. Probing the need for visual context in multimodal machine translation. *arXiv preprint arXiv:1903.08678*.
- Iacer Calixto, Qun Liu, and Nick Campbell. 2017a. Doubly-attentive decoder for multi-modal neural machine translation. *arXiv preprint arXiv:1702.01287*.
- Iacer Calixto, Qun Liu, and Nick Campbell. 2017b. Incorporating global visual features into attention-based neural machine translation. *arXiv preprint arXiv:1701.06521*.
- Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *arXiv preprint arXiv:1406.1078*.
- Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30k: Multilingual english-german image descriptions. *arXiv preprint arXiv:1605.00459*.
- Desmond Elliott and Ákos Kádár. 2017. **Imagination improves multimodal translation**. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 130–141, Taipei, Taiwan. Asian Federation of Natural Language Processing.
- Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.
- Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Hui-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Po-Yao Huang, Frederick Liu, Sz-Rung Shiang, Jean Oh, and Chris Dyer. 2016. Attention-based multimodal neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 639–645.
- Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLP Suite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.
- Diederik P Kingma and Jimmy Ba. 2014. **Adam: A method for stochastic optimization**. *arXiv preprint arXiv:1412.6980*.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.
- Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Loitongbam Sanayai Meetei, Thoudam Doren Singh, and Sivaji Bandyopadhyay. 2019. Wat2019: English-hindi translation on hindi visual genome dataset. In *Proceedings of the 6th Workshop on Asian Translation*, pages 181–188.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. **Bleu: a method for automatic evaluation of machine translation**. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.
- Shantipriya Parida, Ondřej Bojar, and Satya Ranjan Dash. 2019. Hindi visual genome: A dataset for multimodal english-to-hindi machine translation. *arXiv preprint arXiv:1907.08948*.
- Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2019. Bpe-dropout: Simple and effective subword regularization. *arXiv preprint arXiv:1910.13267*.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015a. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015b. **Neural machine translation of rare words with subword units**. *arXiv preprint arXiv:1508.07909*.
- Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

- Salam Michael Singh and Thoudam Doren Singh. 2020. Unsupervised neural machine translation for english and manipuri. In *Proceedings of the 3rd Workshop on Technologies for MT of Low Resource Languages*, pages 69–78.
- Thoudam Doren Singh and Aiusha Vellintihun Hujon. 2020. Low resource and domain specific english to khasi smt and nmt systems. In *2020 International Conference on Computational Performance Evaluation (ComPE)*, pages 733–737. IEEE.
- Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. [Show and tell: A neural image caption generator](#). In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.