




# Machine translation by projecting text into the same phonetic-orthographic space using a common encoding

AMIT KUMAR<sup>1,\*</sup>, SHANTIPRIYA PARIDA<sup>2</sup>, AJAY PRATAP<sup>1</sup> and ANIL KUMAR SINGH<sup>1</sup>

<sup>1</sup>Indian Institute of Technology (BHU), Varanasi, India

<sup>2</sup>Silo AI, Helsinki, Finland

e-mail: amitkumar.rs.cse17@iitbhu.ac.in; shantipriya.parida@siloi.ai; ajay.cse@iitbhu.ac.in; aksingh.cse@iitbhu.ac.in

MS received 17 April 2021; revised 19 May 2023; accepted 7 July 2023

**Abstract.** The use of subword embedding has proved to be a major innovation in Neural machine translation (NMT). It helps NMT to learn better context vectors for Low resource languages (LRLs) so as to predict the target words by better modelling the morphologies of the two languages and also the morphosyntax transfer. Some of the NMT models that achieve state-of-the-art improvement on LRLs are Transformer, BERT, BART, and mBART, which can all use sub-word embeddings. Even so, their performance for translation in Indian language to Indian language scenario is still not as good as for resource-rich languages. One reason for this is the relative morphological richness of Indian languages, while another is that most of them fall into the extremely low resource or zero-shot categories. Since most major Indian languages use Indic or Brahmi origin scripts, the text written in them is highly phonetic in nature and phonetically similar in terms of abstract letters and their arrangements. We use these characteristics of Indian languages and their scripts to propose an approach based on common multilingual Latin-based encoding (WX notation) that takes advantage of language similarity while addressing the morphological complexity issue in NMT. Such multilingual Latin-based encodings in NMT, together with Byte Pair Embedding allow us to better exploit their phonetic and orthographic as well as lexical similarities to improve the translation quality by projecting different but similar languages on the same orthographic-phonetic character space. We verify the proposed approach by demonstrating experiments on similar language pairs (Gujarati↔Hindi, Marathi↔Hindi, Nepali↔Hindi, Maithili↔Hindi, Punjabi↔Hindi, and Urdu↔Hindi) under low resource conditions. The proposed approach shows an improvement in a majority of cases, in one case as much as  $\sim 10$  BLEU points compared to baseline techniques for similar language pairs. We also get up to  $\sim 1$  BLEU points improvement on distant and zero-shot language pairs.

**Keywords.** Neural machine translation; common phonetic-orthographic space; similar languages; byte pair encoding; transformer model.

## 1. Introduction

Machine Translation (MT) has an interesting history in computation and research [1] with new paradigms being introduced over decades. MT achieved a watershed moment after the introduction of numerous algorithmic, architectural and training enhancements, such as Statistical Machine Translation (SMT) and Neural Machine Translation (NMT) [2]. SMT is a statistical-based MT paradigm, operating at the granularity of words and phrases, consisting of a translation model, a language model, and a decoder [3–5]. Further, the relatively recent success of deep neural networks has given us end-to-end variations of translation models such as recurrent NMT [6, 7], attention-based NMT, and self-attention-based Transformer [8].

There have been parallel and related developments in language models, such as Bidirectional Encoder Representations from Transformers (BERT) [9] and ALBERT [10]. Another variant of this, mBART, has provided benchmark solutions in NMT as well [11]. However, training an effective and accurate MT system still requires a large amount of parallel corpus consisting of source and target language pairs. When we talk about low-resource languages, the first problem is to find a fair amount of parallel corpus, sometimes even monolingual corpus, which makes it challenging to create tools and applications for extremely poor resource languages. Creating a large parallel corpus for MT for each language pair that falls into the low resource category is an expensive, time-consuming, and labor-intensive task.

So, one solution to improve NMT in a low-resource context is to bootstrap the process by leveraging the

\*For correspondence

morphological, structural, functional, and perhaps deep semantic features of such languages. Fortunately, for similar languages, it also is possible to exploit the similarities for better modeling of closely related languages. We need to focus on features that help the MT system better learn the close relationships between such languages. Conference on Machine Translation (WMT) has also conducted shared tasks for similar language translations from 2019 [12].

When we talk about Indian languages, most languages except Hindi come under extremely low resource categories. Even Hindi is, from some points of view either a low or medium resource language [13, 14]. India being a country with rich linguistic diversity, there is a need for MT systems across the Indian (or South Asian) languages. India is also inhabited by a vast population who speak languages belonging to three prominent families, Indo-Aryan (a sub-family of Indo-European), Dravidian, and Tibeto-Burman, but due to very long contact and interactions, they have gone through a process of ‘convergence’, forming India as a linguistic area [15]. Due to this long term contact, there are more similarities among these languages than we would otherwise expect. In addition, significant fractions of their vocabularies, to varying degrees, have words originating in or borrowed from Sanskrit, Persian, Arabic, Turkish and English, among other languages.

For some of the major languages, and even for some of the ‘regional’ or ‘minority languages’ (since they were widely used for a long duration in the past for literary purposes), there are records available and there is a varying degree of well-developed tradition of at least (spoken) literary usage. However, only some languages, most of which are officially recognized, have some written tradition, particularly for non-literary prose. The rest have very little written data, or even if it is there, it is usually not in a machine-readable format. Therefore, they can be treated as extremely low or zero-resource languages. There is a need for development of MT systems for such languages, and the similarity between these languages helps in developing such MT systems.

In this article, we propose an approach based on leveraging the features of similar languages by simply, programmatically<sup>1</sup>, converting them into an intermediate Latin-based multilingual notation. The notation that we use here is the commonly used WX-notation [16], which is often used in NLP tools and systems for Indian languages developed in India. This notation (like many other similar notations) can project all the Indic or Brahmi origin scripts [17], which have—in many cases—different Unicode blocks, into a common character space. Our intuition, is that this should help in capturing phonological, orthographic, and, to some extent, morphosyntactic similarities that will help a neural network-based model in better multilingual learning and translation across these languages [18–20]. We do this by using this WX-converted text to

learn byte pair encoding-based embeddings. The effect of this is that the similar but different languages are projected onto the same orthographic-phonetic space [21], and hence also in the same common morphological and lexical space, allowing better modeling of multilingual relationships in the context of India as a linguistic area.

In addition, using WX has another benefit, even for a single script such as Devanagari. Brahmi-derived scripts have different symbols for dependent vowels (called *maatraas*) which modify a consonant and independent vowels (written as *aksharas*) which are pronounced as syllables. WX uses the same symbols for these two variants of the same vowel, while Unicode uses different codes and the scripts themselves use different graphical symbols. This gives an advantage in terms of learning a representation with less number of abstract letters.

After conversion to WX, we apply some of the state-of-the-art NMT techniques to build our MT systems. These NMT systems, such as the Transformer, should learn better the relationships between languages through learnt representations.

We select six pairs of similar languages: Gujarati (GU)↔Hindi (HI), Marathi (MR)↔Hindi (HI), Nepali (NE)↔Hindi (HI), Maithili (MAI)↔Hindi (HI), Punjabi (PA)↔Hindi (HI), and Urdu (UR)↔Hindi (HI). table 1 contains some of the language features that help in figuring out how the selected languages are similar to Hindi. For example, Hindi, Gujarati, Marathi, Nepali, Maithili, Punjabi, and Urdu belong to Indo-Aryan Language families, and all the selected languages except Punjabi and Urdu share a common Devanagari script<sup>2</sup>. The word order of all the selected languages is mostly *Subject + Object + Verb*. Apart from this, all these languages share lexical similarities with Hindi in terms of common words derived from Sanskrit and other languages as mentioned earlier. Also, these languages have phonological similarities with Hindi. We also note that though Urdu and Hindi are linguistically almost the same language, yet due to the great divergence in their vocabularies and linguistic styles in their written and sometimes spoken forms, they have only a relatively small overlap in their corpus-based vocabularies, albeit this overlap consists mainly of core words which form a major component of the linguistic identity of a language in terms of linguistic typology and phylogenetic classification.

This papers is the first part of a series of three papers exploring and then extending the idea of using common phonetic-orthographic space for better NMT in the Indian context [22, 23]. The contributions of this paper are summarized as follows:

<sup>1</sup>Using encoding converters, such as <https://pypi.org/project/wxconv/>

<sup>2</sup>Punjabi is written in two scripts, Gurumukhi and Shahmukhi, of which the former is a Brahmi-derived script, while the latter is a variant of Perso-Arabic. Urdu is written in a similar variant of Perso-Arabic, also called Nastaliq.

**Table 1.** Some details about the languages used in our experiments.

Languages	Family	Script	Word order	Ergative	Place
Hindi	Indo-Aryan	Devanagari	SOV	Yes	Mainly North India
Gujarati		Gujarati		No	Mainly Gujarat
Marathi		Balbodh version of Devanagari		No	Mainly Maharashtra
Nepali		Devanagari		Yes	Mainly Nepal
Maithili		Devanagari		No	Mainly Bihar and parts of Nepal
Punjabi		Gurumukhi		No	Mainly Punjab
Urdu		Variant of Perso-Arabic		No	Mainly North India

**Table 2.** Comparison of some existing work. ✓ and ✗ represent presence and absence of a particular feature, respectively.

Paper	Similar Language	Reducing morphological complexity	Statistical	Neural	WX	Language pair
[24]	✓	✗	✗	✓	✗	HI↔MR, ES↔PT
[25]	✓	✗	✗	✓	✗	HI↔MR
[26]	✓	✗	✗	✓	✗	HI↔MR
[27]	✓	✗	✗	✓	✗	NE↔HI
[28]	✓	✗	✓	✓	✗	HI↔MR
[29]	✓	✗	✗	✓	✗	HI↔MR
[31]	✓	✗	✗	✓	✗	HI↔MR
[32]	✓	✗	✗	✓	✗	ES↔PT, CS↔PL, NE↔HI
[33]	✓	✗	✗	✓	✗	11 Indian languages
[34]	✓	✗	✗	✓	✗	11 Indic languages and English
Proposed approach	✓	✓	✗	✓	✓	{GU,MR,NE,MAI,PA,UR}↔HI

HI: Hindi, MR: Marathi, ES: Spanish, PT: Portuguese, NE: Nepali, CS: Czech, PL: Polish, GU: Gujarati, MAI: Maithili, PA: Punjabi, UR: Urdu

- (1) Propose a WX-based machine translation approach that leverages orthographic and phonological similarities between pairs of Indian languages.
- (2) Proposed approach achieves an improvement of  $+0.01$  to  $+10$  BLEU points compared to baseline state-of-the-art techniques for similar language pairs in most cases. We also get  $+1$  BLEU points improvement on distant and zero-shot language pairs.

The rest of the paper is organized as follows. Section 2 discusses closely related works. Section 3 describes some background and the NMT models that we extend or compare with. Section 4 describes the proposed approach in more detail. Section 5 discusses corpus statistics and experimental settings used to conduct the experiments. Results and ablation studies are reported in sections 6 and 7, respectively. Finally, the paper is summarized in Section 8 and includes some directions for future work.

## 2. Related works

This section briefly describes some of the related work (table 2) on language similarity, morphological richness, statistical and neural models, and the language pairs used as discussed below.

Although there had been work in the past, the recent sharper focus on machine translation for similar languages is also due to the shared tasks on this topic organized as part of the WMT conferences from 2019 to 2021. In [24], authors demonstrated that pre-training could help even when the language used for fine-tuning is absent during pre-training. In [25], authors experimented with attention-based recurrent neural network architecture (seq2seq) on HI↔MR and explored the use of different linguistic features like part-of-speech and morphological features, along with back translation for HI→MR and MR→HI machine translation. In [26], authors ensembled two Transformer models to try to allow the NMT system to learn the nuances of translation for low-resource language pairs by taking advantage of the fact that the source and target languages are written using the same script. In [27], authors' work relied on NMT with attention mechanism for the similar language translation in the WMT19 shared task in the context of NE↔HI language pair.

In [28], the authors conducted a series of experiments to address the challenges of translation between similar languages. Based on these experiments, the authors developed one phrase-based SMT system and one NMT system using byte-pair embedding for the HI↔MR pair. In [29], authors used a Transformer-based NMT with *sentencepiece* for subword embedding on HI↔MR language pair [30]. In

[31], authors used the Transformer-NMT for multilingual model training and evaluated the result on the HI $\leftrightarrow$ MR pair. In [32], authors focused on incorporating monolingual data into NMT models with a back-translation approach. In [33], authors introduced NLP resources for 11 major Indian languages from two major language families. These resources include: large-scale sentence-level monolingual corpora, pre-trained word embeddings, pre-trained language models, and multiple NLU evaluation datasets. In [34], authors presented IndicBART, a multilingual, sequence-to-sequence pre-trained model focusing on 11 Indic languages and English. IndicBART also utilizes the orthographic similarity between Indic scripts to improve transfer learning between similar Indic languages.

### 2.1 Shortcomings of existing works

In most of the existing work on MT for related languages (e.g., [29, 31, 32]), authors have discussed improving the NMT models using extra monolingual corpora in addition to bi-lingual data. However, the proposed approach improves translation quality using only bilingual corpora with the help of WX-transliteration. The proposed approach reduces language complexity by transliterating the text to a Latin-based notation, which helps the NMT models to better learn the context information by exploiting language similarities. In this way, where applicable, it can complement the approaches which use extra monolingual data.

## 3. Background

This section provides some background on the recent most successful machine translation techniques. From vanilla NMT to more robust and advanced BART, a denoising autoencoder for pre-training sequence-to-sequence models, remarkable advances in NMT techniques have been made in a relatively short time.

### 3.1 NMT

Many of the NMT techniques use an encoder-decoder architecture based on neural networks that performs translation between language pairs. Numerous enhancements, toolkits, and open frameworks are available to train NMT models, such as OpenNMT. OpenNMT is one of the open-source NMT frameworks [35], used to model natural language tasks such as text summarization, tagging, and text generation. This toolkit is used for model architectures, feature representations, and source modalities in NMT research. Multilingual and zero-shot NMT have also been

applied for NMT to achieve state-of-the-art results on different language pairs by using a single standard NMT model for multiple languages [36]. Furthermore, the introduction of attention in NMT has drastically improved the results significantly [37], as for many other problems. As shown in figure 1, early NMT was an encoder-decoder sequence-based model consisting of recurrent neural network (RNN) units. The encoder consists of RNN units ( $E_0, E_1, E_2$ ) and takes as input the embedding of words from sentences and produces the context vector ( $\mathbf{C}$ ) as follows:

$$\mathbf{C} = \text{Encoder}(\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \dots, \mathbf{X}_n) \quad (1)$$

where,  $\{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \dots, \mathbf{X}_n\}$  is the input source sequence.

The decoder consists of RNN units ( $D_0, D_1, D_2, D_3$ ) and it decodes these context vectors into target sentences with an  $\langle \text{END} \rangle$  (end of a sentence) symbol as follows:

$$\text{Decoder}(\mathbf{C}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \dots, \mathbf{Y}_n) = \mathbf{Y}'_1, \mathbf{Y}'_2, \mathbf{Y}'_3, \dots, \mathbf{Y}'_m \quad (2)$$

where,  $\{\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \dots, \mathbf{Y}_n\}$  and  $\{\mathbf{Y}'_1, \mathbf{Y}'_2, \mathbf{Y}'_3, \dots, \mathbf{Y}'_m\}$  are the target and the predicted sequences, respectively.

### 3.2 Transformer-based NMT

The Transformer can be characterized by its breakthrough in combining five innovations elegantly in a single architecture. The first is the attention mechanism [8]. It maps a query and a set of key-value pairs to an output. A compatibility function of the query with the corresponding key computes the weights. The second extends the first by using multi-head self-attention. The third is the use of positional encoding in terms of relative positions, which allows it to learn temporal relationships and dependencies. The fourth is the use of masking, which has proved to be immensely effective in many other later models. The fifth is the use of residual connections. Together, the elegant combination of these innovations not only allows the model to learn much better models, but also obviates the need for recurrent units in the architecture, which in turn allows a great degree of parallelism during training the models. In other words, the Transformer model not only learns much better models, but does so in much less time during the training phase. Moreover, the problem of overfitting is also much less with the Transformer-based models.

There are numerous state-of-the-art results reported for machine translation systems using a Transformer. Currey and Heafield [38] incorporated syntax into the Transformer using a mixed encoder model and multi-task machine translation. Multi-head attention is one key feature of self-attention. Fixing the attention heads on the encoder side of the Transformer increases BLEU scores by up to 3 points in low-resource scenarios [39]. The most common attention

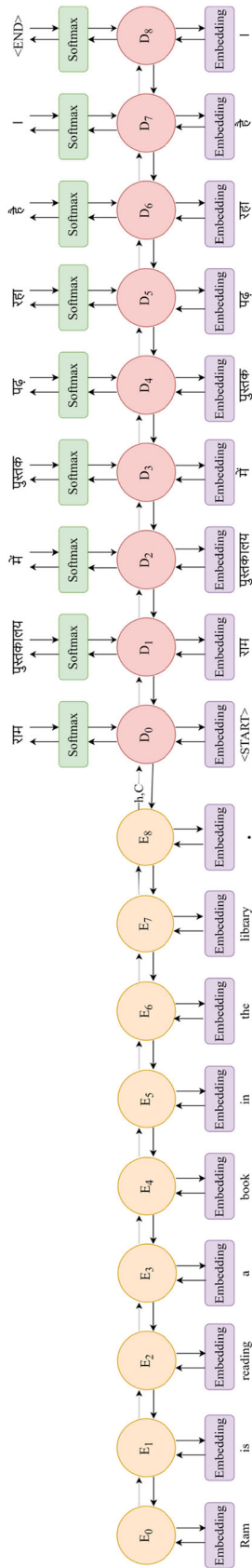


Figure 1. Vanilla NMT.

functions are additive attention and dot product attention. Transformer generates the scaled dot-product attention as follows [8]:

$$\text{attn}_i = \text{softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_i^T}{\sqrt{d_k}}\right) \mathbf{V}_i \quad (3)$$

where,  $\mathbf{Q}_i$ ,  $\mathbf{K}_i$ ,  $\mathbf{V}_i$  and  $d_k$  are query, key, value and the dimension of the key, respectively.

### 3.3 BART

BART is a denoising autoencoder for pretraining sequence-to-sequence models [40]. It uses a standard Transformer-based NMT architecture to generalize BERT, GPT, and many other recent pre-training schemes. BART uses the standard Transformer architecture, except it modifies ReLU activation functions to GeLUs. Its mBART variation is a sequence-to-sequence denoising auto-encoder pre-trained on monolingual corpora in multiple languages using the BART objective [11].

### 3.4 Back-translation

Back-translation is a method to prepare synthetic parallel corpus from a monolingual corpus for NMT [41]. In low-resource settings, back-translation can be a very effective method. Iterative back-translation is a further improvement [42]. It iterates over two back-translation systems multiple times.

### 3.5 Similar languages

Similar languages refer to a group of languages that share common ancestry or extensive contact for an extended period of time, or both, with each other, leading them to exhibit structural and linguistic similarities even across language families. Examples of languages that share common ancestors are Indo-Aryan languages, Romance languages, and Slavic languages. Languages in contact for a long period lead to the convergence of linguistic features even if languages do not belong to common ancestors. Prolonged contact among languages could lead to the formation of linguistic areas or *sprachbunds*. Examples of such linguistic areas are the Indian subcontinent [15], the Balkan [43], and Standard Average European [44] linguistic areas.

Similarities between languages depend on various factors. Some of the factors are lexical similarity, structural correspondence, and morphological isomorphisms. Lexical similarity means that the languages share many words with similar forms (spelling/ pronunciation) and meaning, e.g. Sunday is written as रविवार (ravivAra) in Hindi and रबिवार (rabiVra) in Bhojpuri (both are proximate and related Indo-



Aryan languages). These lexically similar words could be cognates, lateral borrowings, or loan words from other languages. Structural correspondence means, for example, that languages have the same basic word order, viz. SOV (Subject-Object-Verb) or SVO (Subject-Verb-Object). Morphological isomorphisms refers to the one-to-one correspondence between inflectional affixes. While content words are borrowed or inherited across similar languages, function words are generally not lexically similar across languages. However, function words in related languages (whether suffixes or free words) tend to have a one-one correspondence to varying degrees and for various linguistic functions.

### 3.6 Transformer-based NMT with Back-translation

Guzmán *et al* [45], in their work, first trained a Transformer on Nepali-English and Sinhala-English language pairs in both directions, and then they used the trained model to translate monolingual target language corpora to source languages. Finally, the source language sentence corpus was merged with generated source language sentences and was given as input to the Transformer for training and producing the translation.

## 4. Proposed approach

To tackle the morphological richness related problems in NMT training for Indian languages and to be able work with very little resources, we propose a simple but effective approach for translating low-resource languages that are similar in features and behaviour.

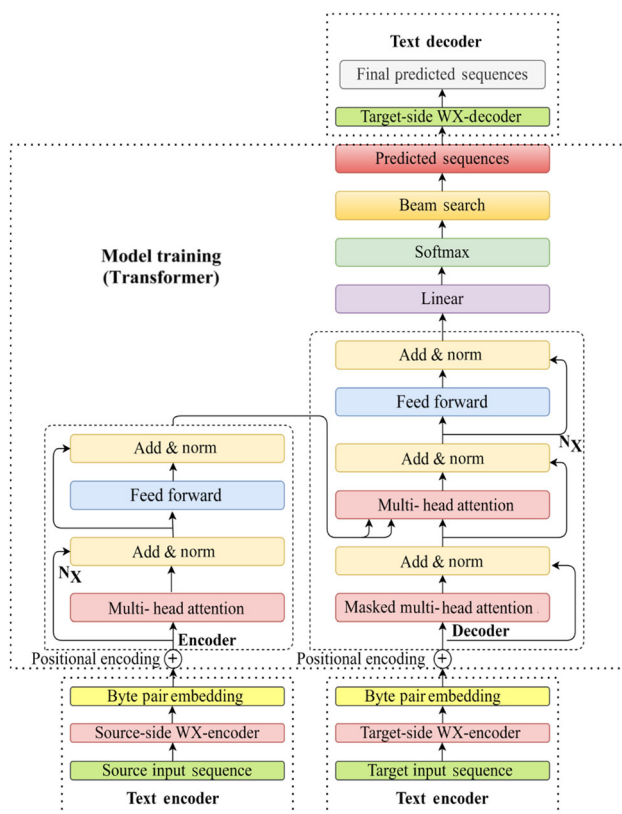
The proposed approach consists of three modules: Text Encoder, Model Trainer, and Text Decoder (figure 2), as discussed in the following section.

### 4.1 Text encoder

The proposed model first encodes the source and target corpora of parallel languages into an intermediate representation, the WX-notation<sup>3</sup> [46]. The primary reason behind encoding the source and target language corpora into WX-notation is to encode different languages with the same or different scripts into a common representation by projecting them onto a common phonetic-orthographic character space so that BPE can be linguistically better informed. WX-notation is a transliteration scheme for representing Indian languages in ASCII format, and as described earlier, it has many advantages as an intermediate representation, even compared to using Devanagari or any other single Brahmi-based script, as in the case of IndicBERT and IndicBART, which have conversion to Devanagari. It implicitly helps the Transformer encoder model more cognates, loan words, and morphologically similar words between the languages, as well as model other kinds of similarities for better translation.

### 4.2 Model training

The intermediate representation of the source language text is passed to the Transformer encoder. The Transformer encoder-decoder model learns the relationship between languages. We have used the SentencePiece<sup>4</sup> library for tokenization of the text. SentencePiece is used as a pre-processing task for the WX-encoded source-target text in the concerned language pair. SentencePiece is a language-independent sub-word tokenizer and detokenizer designed for Neural-based text processing, including neural machine translation. It implements two subword segmentation algorithms, Byte-Pair Encoding (BPE) and unigram language model, with direct training from raw sentences [47, 48]. Therefore, it already indirectly, to some extent, provides cognates, loan words, and morphologically similar words to the Transformer, and our prior conversion to WX allows it to do so better. It may be noted that the approach is generalizable to other multilingual transliteration



**Figure 2.** Proposed architecture.

<sup>3</sup><https://pypi.org/project/wxconv/>, <https://github.com/irshadbhat/indic-wx-converter>

<sup>4</sup><https://github.com/google/sentencepiece>

notations, perhaps even to IPA<sup>5,6</sup>, which is almost truly phonetic notation for written text.

### 4.3 Text decoder

After convergence of the training algorithm, the WX-encoded generated target sentences are decoded back to the plain text format to evaluate the model.

## 5. Corpus and experimental settings

In this section, we discuss the corpus statistics and experimental settings we used for our experiment (table 3).

### 5.1 Corpus description

We evaluate the proposed model in an extremely low-resource scenario on the mutually similar languages which we selected for our experiments. These are Hindi (HI), Gujarati (GU), Marathi (MR), Nepali (NE), Maithili (MAI), Punjabi (PA), Urdu (UR), Bhojpuri (BHO), Magahi (MAG), Malayalam (ML), Tamil (TA) and Telugu (TE). We perform experiments on the following language pairs involving Hindi: GU↔HI, NE↔HI, MR↔HI, MAI↔HI, PA↔HI, and UR↔HI. Parallel corpora of GU↔HI, ML↔HI, TA↔HI, and TE↔HI for training, testing, and validation are downloaded from CVIT-PIB [49]. MR↔HI parallel corpus is collected from WMT 2020 shared tasks<sup>7</sup>. NE↔HI language pair corpus is made up of those collected from WMT 2019 shared tasks<sup>8</sup>, Opus<sup>9</sup>, and TDIL<sup>10</sup> repositories. We use a monolingual corpus of Gujarati, Hindi, and Marathi for similarity computation in Section 6.1 from the PM India dataset described in [50]. The rest of the monolingual corpora are collected from the Opus collection for similarity computation in Section 5.1 [51]. We use SentencePiece [52] to pre-process the source and target sentences. We use 5K merge operations to learn BPE with the SentencePiece model and restrict the source and target vocabularies to at most 5K tokens. There are some places where code-switching occurs in the employed dataset. The WX-transliteration tool ignores code-switched data and keeps it in the datasets as it is.

<sup>5</sup>[https://en.wikipedia.org/wiki/International\\_Phonetic\\_Alphabet\\_chart](https://en.wikipedia.org/wiki/International_Phonetic_Alphabet_chart)

<sup>6</sup><https://www.internationalphoneticassociation.org/>

<sup>7</sup><http://www.statmt.org/wmt20/similar.html>

<sup>8</sup><http://www.statmt.org/wmt19/index.html>

<sup>9</sup><https://opus.nlpl.eu/>

<sup>10</sup><http://www.tdil-dc.in/index.php?lang=en>

### 5.2 Training details

**5.2.1 Proposed approach** We use the WX-notation tool<sup>11</sup> for transliterating the text and the fairseq<sup>12</sup> [53] toolkit, which is a sequence modelling toolkit, to train the Transformer. We use five encoder and decoder layers. The encoder and decoder embedding dimensions are set to 512. Feed-forward encoding and decoding embedding dimensions are set to 2048. The number of encoder and decoder attention heads is set to 2. The dropout, the attention dropout, and the ReLU dropout are set to 0.4, 0.2, and 0.2, respectively. The weight decay is set to 0.0001, and the label smoothing is set to 0.2. We use the Adam optimizer, with  $\beta_1$  and  $\beta_2$  set to 0.9 and 0.98. The learning rate schedule is inverse square root, with an initial learning rate of 1e-3 and a minimum learning rate of 1e-9. The maximum number of tokens used is set to 4000. The maximum number of epochs for training is set to 100. We use a beam size equal to 5 for generating data using the test set.

**5.2.2 Guzmán et al [45]** In Guzmán et al [45], authors have demonstrated the experiments on extremely low resource languages using Transformer. Our proposed approach is based on the Transformer described in Guzmán et al [45] with the addition of two extra modules, Text Encoder and Text Decoder. We use the Transformer model described in Guzmán et al [45] as a reasonably high baseline to compare the proposed approach without the intermediate representation of the WX-notation for Indian languages. The projection to WX could be used for any other NMT approach as well that uses a subword embedding.

**5.2.3 SMT** We use Moses<sup>13</sup>, an open-source toolkit to train SMT [54]. For obtaining the phrase/word alignments from parallel corpora, we use GIZA++ [55]. A 5-gram KenLM language model is used for training [56]. The parameters are tuned on the validation set using MERT and tested with a test set [57].

## 6. Results and analysis

We compare the proposed approach with the Moses-based SMT and the Transformer-based NMT model [45], where the latter is used as the baseline for NMT. We use six evaluation metrics, BLEU<sup>14</sup> [58], LEBLEU [59], WupLeBleu [60], TER [61], WER, and chrF2 [62] for better comparison of the proposed approach. We see from tables 4

<sup>11</sup><https://pypi.org/project/wxconv/>

<sup>12</sup><https://github.com/facebookresearch/fairseq>

<sup>13</sup><http://www2.statmt.org/ Moses/>

<sup>14</sup><https://github.com/mjpost/sacrebleu>

**Table 3.** Corpus Statistics showing the number of training, validation, and test sentences for each domain.

Lang-pairs	Train	Validation	Test	Domain
GU↔HI	15784	1000	1973	PM India
NE↔HI	136991	3000	3000	WMT 2019 corpus, Agriculture, Entertainment, Bible
MR↔HI	43274	1000	1411	News, PM India, Indic WordNet
PA↔HI	225576	7199	7200	GNOME, KDE4, Ubuntu, wikimedia, TED2020
MAI↔HI	93136	2972	2973	GNOME, KDE4, wikimedia, Ubuntu
UR↔HI	108176	3452	3453	Tanzil, GNOME, KDE4, wikimedia, Ubuntu
ML↔HI	17333	500	500	PM India
TA↔HI	43538	500	500	PM India
TE↔HI	2584	500	500	PM India
BHO↔HI	0	500	500	Movie subtitles, Literature, News
MAG↔HI	0	500	500	Movie subtitles, Literature, News

HI: Hindi, MR: Marathi, NE: Nepali, GU: Gujarati, MAI: Maithili, PA: Punjabi, UR: Urdu, ML: Malayalam, TA: Tamil, TE: Telgu, BHO: Bhojpuri, MAG: Magahi

**Table 4.** Experiment results (BLEU, chrF2, and TER scores).

Languages(xx)	BLEU		chrF2		TER	
	Guzmán <i>et al</i> [45]	Proposed	Guzmán <i>et al</i> [45]	Proposed	Guzmán <i>et al</i> [45]	Proposed
			<i>XX→HI</i>			
GU	33.14	<b>33.15</b>	<b>58</b>	57	<b>0.541</b>	0.548
NE	30.51	<b>41.97</b>	46	<b>49</b>	0.658	<b>0.652</b>
MR	16.87	<b>22.37</b>	43	<b>44</b>	<b>0.707</b>	0.709
PA	78.56	<b>81.05</b>	82	<b>82</b>	0.220	<b>0.216</b>
UR	28.74	<b>30.08</b>	45	<b>45</b>	0.668	<b>0.657</b>
MAI	79.49	<b>81.80</b>	<b>82</b>	81	<b>0.242</b>	0.251
			<i>HI→XX</i>			
GU	25.47	<b>25.82</b>	56	<b>56</b>	<b>0.616</b>	0.619
NE	32.89	<b>43.52</b>	50	<b>51</b>	<b>0.630</b>	0.637
MR	14.05	<b>14.76</b>	41	<b>44</b>	0.789	<b>0.762</b>
PA	80.01	<b>81.87</b>	83	<b>84</b>	0.206	<b>0.203</b>
UR	22.74	<b>24.35</b>	46	<b>47</b>	0.597	<b>0.596</b>
MAI	<b>86.58</b>	83.82	<b>89</b>	86	<b>0.148</b>	0.168

Bold indicates better scores

and 5 that the proposed approach improves upon the baseline for most of the pairs.

BLEU score, although a simple metric based on comparison of  $n$ -grams, is a standard metric accepted by NLP researchers to obtain the accuracy of predicted translated outputs compared to the human-translated reference sentences. This is because it has been observed that the value of the BLEU score correlates well with human-judged quality of translations. The formula for the BLEU score is as follows [58]:

$$BLEU = \min\left(1, \frac{output\_length}{reference\_length}\right) \left(\prod_{i=1}^4 precision_i\right), \quad (4)$$

where the *output\_length* and the *reference\_length* are the lengths of the predicted sentences and the reference sentences, respectively.

We also perform a comparison between SMT without WX-transliteration and SMT with it. These two sets of



**Table 5.** LEBLEU, WupLeBleu and WER scores.

Languages(xx)	LEBLEU		WupLeBLEU		WER	
	Guzmán <i>et al</i> [45]	Proposed	Guzmán <i>et al</i> [45]	Proposed	Guzmán <i>et al</i> [45]	Proposed
			<i>XX→HI</i>			
GU	<b>0.663</b>	0.657	<b>0.663</b>	0.657	66.77	<b>66.29</b>
NE	0.543	<b>0.547</b>	0.543	<b>0.547</b>	<b>66.99</b>	67.71
MR	0.495	<b>0.541</b>	0.495	<b>0.541</b>	<b>72.78</b>	73.36
PA	0.853	<b>0.853</b>	0.853	<b>0.853</b>	22.29	<b>21.83</b>
UR	0.564	<b>0.566</b>	0.564	<b>0.566</b>	68.34	<b>67.20</b>
MAI	<b>0.865</b>	0.851	<b>0.865</b>	0.851	<b>24.34</b>	25.23
			<i>HI→XX</i>			
GU	0.622	<b>0.623</b>	0.622	<b>0.623</b>	<b>73.11</b>	73.33
NE	<b>0.547</b>	0.519	<b>0.547</b>	0.519	<b>63.41</b>	65.31
MR	<b>0.485</b>	0.454	<b>0.485</b>	0.454	80.10	<b>77.46</b>
PA	0.858	<b>0.865</b>	0.858	<b>0.865</b>	20.88	<b>20.57</b>
UR	0.619	<b>0.629</b>	0.619	<b>0.629</b>	62.35	<b>62.27</b>
MAI	<b>0.916</b>	0.908	<b>0.916</b>	0.908	<b>14.83</b>	16.89

Bold indicates better scores

**Table 6.** BLEU score-based comparison of SMT, SMT + WX and the proposed approaches.

Languages(xx)	BLEU			
	SMT	SMT + WX	Proposed	
			<i>XX→HI</i>	
GU	<b>43.49</b>	30.69	33.15	
NE	40.14	<b>53.21</b>	41.97	
MR	<b>7.41</b>	1.46	22.37	
PA	68.34	<b>71.22</b>	81.05	
UR	19.21	<b>21.84</b>	30.08	
MAI	79.56	<b>81.46</b>	81.80	
			<i>HI→XX</i>	
GU	<b>39.20</b>	25.89	25.82	
NE	40.21	<b>54.84</b>	43.52	
MR	<b>7.36</b>	1.48	14.76	
PA	67.21	<b>70.64</b>	81.87	
UR	18.24	<b>18.41</b>	24.35	
MAI	79.12	<b>83.06</b>	83.82	

Bold indicates better scores

results are also compared with the proposed approach as shown in table 6. In the case of SMT also we can easily note that the performance improves in most cases by using WX as the intermediate notation, even though SMT is not using subword embeddings.

We also present some basic analysis of the scores as shown in tables 4 and 5. We use corpus-based language relatedness and complexity measures for further analysis for this purpose in the next section.

## 6.1 Similarity between languages

Since there are no definitive methods to judge the similarity between two languages, we use the following techniques to compute the similarity between the languages:

**6.1.1 SSNGLMScore** We use character-level  $n$ -gram language model based SSNGLMScore to measure the relatedness between languages [63, 64]. SSNGLMScore is computed as follows:

$$S_{sl,tl} = \sum_{tl=1}^m p_{sl,tl}(w_n | w_1^{n-1}), \quad (5)$$

where  $S$  stands for Scaled Sum of  $n$ -gram language model scores.

$$MS_{sl,tl} = \frac{S_{sl,tl} - \min(S_{SL,TL})}{\max(S_{SL,TL}) - \min(S_{SL,TL})}, \quad (6)$$

where,  $sl$  and  $tl$  represent the source language and the target language, respectively. Moreover,  $sl \in SL(\text{Gujarati, Marathi, Maithili, Nepali, Urdu, Punjabi, Hindi, Malayalam, Tamil, Telugu, Bhojpuri, Magahi})$  and  $m$  is the total number of sentences in the target language  $tl \in TL(\text{Gujarati, Marathi, Maithili, Nepali, Urdu, Punjabi, Hindi, Malayalam, Tamil, Telugu, Bhojpuri, Magahi})$ . We train the language model using a 6-gram character-level KenLM model on the source monolingual corpus ( $sl$ ). Each language model is tested on target language ( $tl$ ), and the scores are reported.

Table 7 lists the cross-lingual similarity scores of Hindi, Gujarati, Marathi, Nepali, Maithili, Punjabi, Malayalam, Tamil, Telugu, Bhojpuri, Magahi, and Urdu with each other. Based on SSNGLMScore, Bhojpuri, Maithili and Magahi are the closest to Hindi, which matches linguistic knowledge about them, whereas Urdu seems to as far from

**Table 7.** Similarity between languages using SSNGLMScore.

Model	BHO	GU	HI	MAG	MAI	ML	MR	NE	PA	TA	TE	UR
BHO	–	0.5659	0.6725	0.6997	0.7235	0.4090	0.5687	0.4979	0.4580	0.3233	0.5057	0.4237
GU	–	–	0.5483	0.5642	0.6449	0.3727	0.5411	0.3868	0.3408	0.2531	0.4578	0.3787
HI	–	–	–	0.6331	0.6598	0.3536	0.5717	0.4181	0.4046	0.2564	0.4567	0.3670
MAG	–	–	–	–	0.7762	0.4414	0.5724	0.5671	0.4827	0.3736	0.5248	0.5245
MAI	–	–	–	–	–	0.5833	0.6496	0.6968	0.5734	0.5453	0.6435	0.7040
ML	–	–	–	–	–	–	0.3736	0.3388	0.1968	0.3792	0.4507	0.2759
MR	–	–	–	–	–	–	–	0.4023	0.3496	0.2637	0.4771	0.3498
NE	–	–	–	–	–	–	–	–	0.2661	0.2784	0.3985	0.4354
PA	–	–	–	–	–	–	–	–	–	0.1449	0.2718	0.2938
TA	–	–	–	–	–	–	–	–	–	–	0.2972	0.2641
TE	–	–	–	–	–	–	–	–	–	–	–	0.3493
UR	–	–	–	–	–	–	–	–	–	–	–	–

**Table 8.** char-BLEU score on the training data.

Languages	char-BLEU
Gujarati↔Hindi	47.29
Marathi↔Hindi	35.05
Nepali↔Hindi	40.53
Maithili↔Hindi	66.70
Punjabi↔Hindi	37.17
Urdu↔Hindi	8.61

Applying char-BLEU score on the training data of both the languages of the pair

Hindi as Malayalam and more than Telugu. The reasons Urdu is far from Hindi is partly that Urdu is written in a different kind of script from Hindi which does not have a straightforward mapping to WX, but mainly because, though grammatically almost identical, the two use very different vocabularies in written and formal forms. Maithili is also the second official language of Nepal and is also highly similar to Nepali, perhaps due to prolonged close contact. What is more surprising is that the similarity between Urdu and Nepali is relatively high, whereas that between Urdu and Hindi is among the lowest. This could be because of the nature of the corpus. Going through tables 4 and 5, we find that there is an improvement in every metric except WER and TER in a majority of cases when we apply the proposed method on the translation direction from Maithili, Gujarati, Marathi, Nepali, Punjabi, and Urdu to Hindi. This observation allows us to assert that the proposed approach improves performance for translation between similar languages. Thus, even though the similarity measure we used mixes different kinds of similarities, it is suitable for our purposes because our method is based on sub-word and multilingual modelling.

We also see a gain of +1.34 BLEU points on Hindi to Urdu despite Urdu being far away from the rest of the language pairs in terms of the similarity score we used.

There is a considerable improvement of +11.46 BLEU points on HI→NE and +10.63 BLEU points on NE→HI language pairs.

**6.1.2 char-BLEU, TER and chrF2** To better understand the slight fall in BLEU points despite the similarity for MAI → HI and large increment in the case of NE↔Hi (where Nepali and Maithili are known to be close), we also compute similarity by applying char-BLEU [65], chrF2, and TER on a training dataset of all language pairs. The reason behind using char-BLEU and chrF2 for similarity is that since they are character-based metrics, there is a greater chance of covering the morphological aspects. Before calculating the char-BLEU, the TER, and the chrF2 evaluation metrics, data must be in the same script to evaluate the score. So, we convert the corpus from UTF-8 to WX-notation. Table 8 contains the char-BLEU score of language pairs, whereas table 9 contains the TER and chrF2 scores of each language pair. We see tables 8 and 9 and find out that HI and MAI are still more similar compared to other pairs. We can only hypothesize the reason being that this is due to the nature of the data that we have used.

## 6.2 Analysis on language complexity

**6.2.1 Morphological complexity** Since Indian languages are morphologically rich, machine translation systems based on word tokens have difficulty with them. Therefore, we also tried to relate the results obtained with estimates of such complexity obtained from character-level entropy. It is reasonable to assume that the greater the character-level entropy, the more morphologically complex a language is likely to be.

**Character-level entropy** We used character-level word entropy to estimate morphological redundancy, following Bharati *et al* [66] and Bentz and Alikaniotis [67].

A “word“ is defined in our experiments as a space-separated token, i.e., a string of alphanumeric Unicode

**Table 9.** TER and chrF2 scores on the training data.

Languages	<i>GU</i> → <i>HI</i>	<i>MR</i> → <i>HI</i>	<i>NE</i> → <i>HI</i>	<i>MAI</i> → <i>HI</i>	<i>PA</i> → <i>HI</i>	<i>UR</i> → <i>HI</i>
TER	1.066	1.300	1.052	0.610	0.988	1.093
chrF2	38	29	34	65	32	12
Languages	<i>HI</i> → <i>GU</i>	<i>HI</i> → <i>MR</i>	<i>HI</i> → <i>NE</i>	<i>HI</i> → <i>MAI</i>	<i>HI</i> → <i>PA</i>	<i>HI</i> → <i>UR</i>
TER	0.884	0.940	0.887	0.555	0.906	1.044
chrF2	39	29	36	62	30	10

Applying TER and chrF2 scores on the training data of both the languages of a pair

**Table 10.** Character-based entropy of languages with or without applying WX-notation.

Languages	Character entropy	Character entropy*	Difference
Gujarati	5.0368	3.7454	1.2914
Marathi	5.0220	3.6846	1.3374
Nepali	4.6722	3.5770	1.0952
Maithili	5.1159	3.9162	1.1997
Punjabi	5.0834	3.7932	1.2902
Urdu	4.8821	4.1198	0.7623
Hindi	5.2195	3.7974	1.4221

\* After applying WX-notation

characters delimited by white spaces. The average information content of character types for words is then calculated in terms of Shannon entropy [68]:

$$H(T) = - \sum_{i=1}^V p(c_i) \log_2(p(c_i)) \quad (7)$$

where  $V$  is number of characters ( $c_i$ ) in a word.

Table 10 lists the word (unigram) entropy of languages at character level, which indirectly represents languages' lexical richness, i.e., how complex – in terms of characters they are made up of – word forms are. Since we compute the unigram entropy based on characters, we can say that lexical richness also indicates morphological complexity, both derivational and inflectional. Based on the corpus-based word entropy values, it appears that Hindi is more morphologically complex than the other six languages. However, this may be more of derivational complexity rather than inflectional complexity, as Hindi is relatively simpler in terms of inflectional morphology. The high derivational complexity of Hindi is because it is the official language of India and is more standardized than most other Indian languages. It, therefore, has borrowed and coined a large number of complicated words and technical terms, whether from Persian or Sanskrit or English. This adds a great deal to the derivational complexity of written formal Hindi, compared to commonly spoken Hindi. At least, this is our hypothesis based on the similarity and complexity results.

We also find that our approach shows a considerable improvement of about more than 10 BLEU points in both directions for the Hindi-Nepali language pair, i.e., NE→HI and HI→NE. Such improvement may be attributed to the effect caused by projecting to a common multilingual orthographic-phonetic notation, that is, WX. This probably helps the Transformer learn the context between languages better with the help of a sentence piece tokenizer.

In tables 11, 12 and 13, we present the values of word entropy and redundancy at character level. These tables show that the entropy increases when converting to WX and redundancy decreases. This is evidence of the fact that because the projection to a common orthographic and phonetic space causes the entropy to increase and redundancy to decrease, it becomes possible to learn more compact representations from the data after conversion to WX in our case.

**6.2.2 Syntactic complexity** Perplexity Perplexity ( $PP$ ) of a language can be seen as a weighted average of the reciprocal of its branching factor [63]. Branching factor is the number of possible words that can succeed any given word based on the context. Therefore, perplexity – as a kind of the mean branching factor – is a mean representative of the possible succeeding words given a word. Thus, it can be seen as a rough measure of the syntactic complexity. If the model is a good enough representation of the true distribution for the language, then the  $PP$  value will actually indicate syntactic complexity.

**Table 11.** Entropy computed on vocabulary.

Language	Complete corpus						Restricted corpus					
	Without WX			With WX			Without WX			With WX		
	Max	Median	Average	Max	Median	Average	Max	Median	Average	Max	Median	Average
HI	3.1674	0.5897	0.6196	4.9433	1.2484	1.3148	3.1623	0.5929	0.6230	4.9414	1.2495	1.3158
GU	6.4712	0.8113	0.8389	17.9337	1.4677	1.5157	6.4735	0.8128	0.8410	22.2253	1.4681	1.5163
NE	3.0311	0.8008	0.8287	6.6845	1.4327	1.4835	1.8080	0.5350	0.5636	4.7487	1.1262	1.1575
MR	3.7534	0.5982	0.6281	7.7372	1.2331	1.2995	3.5845	0.8049	0.8459	7.7400	1.2130	1.2734
PA	2.2077	0.5778	0.6048	8.9978	1.0349	1.1105	2.1662	0.5500	0.5753	13.5759	0.9644	1.0405
UR	2.8580	0.6484	0.6786	3.092	0.7748	0.8088	2.2477	0.6282	0.6574	3.3297	0.7523	0.7828
MAI	2.0163	0.5097	0.5326	4.3135	1.0904	1.1432	1.6417	0.4773	0.5003	3.8923	1.0401	1.0888

**Table 12.** Redundancy.

Languages	Complete corpus		Restricted corpus	
	Without WX	WX	Without WX	WX
HI	0.8955	0.7693	0.8949	0.7691
GU	0.8606	0.7401	0.8603	0.7400
NE	0.8806	0.7866	0.9111	0.8147
MR	0.9050	0.7993	0.8610	0.7807
PA	0.9186	0.8502	0.9194	0.8554
UR	0.8941	0.8741	0.8968	0.8750
MAI	0.9125	0.8121	0.9172	0.8171

To estimate distances of other languages from Hindi using perplexity, we trained the perplexity model on the Hindi corpus and tested it on the corpora of other languages.

$$PP(C) = \sqrt[W]{\frac{1}{P(S_1, S_2, S_3, \dots, S_n)}} \quad (8)$$

where corpus  $C$  contains  $n$  sentences with  $W$  words.

Tables 14 and 15 contain the asymmetric and symmetric perplexity values—average of the two translation directions—between the concerned language pairs and indicate their distances from Hindi based on character-level language model. Pairs having higher perplexity scores means the languages are more distant. We see language pairs Urdu and Hindi have more perplexity scores. This is again mostly because these two languages, though almost identical in spoken form and in terms of core syntax and core vocabulary, use very different extended vocabularies for written and formal purposes, besides using very different writing systems. Standard written Urdu uses Persian, Arabic, and Turkish words heavily, whether adapted phonologically or not.

Given the small amounts of data, it is not surprising that the values of perplexity are different in the two translation directions.

Similarly, standard and written Hindi uses words much more heavily derived or borrowed or even coined from Sanskrit. Despite higher perplexity between these two languages, our approach gives a +2 increment in the BLEU score, probably because the common core syntax and core vocabulary manifest themselves in every phrase or sentence and thus have higher probabilistic weight. They are, in fact, completely mutually intelligible in the commonly spoken forms<sup>15</sup> and partly in the written form. There are also a lot of Indians who can comfortably read and understand both these languages, even in their standard, written, and literary forms. The use of WX perhaps allows the models to exploit the core similarities better.

## 7. Ablation study

This section discusses ablation studies conducted using the proposed method on distant and zero-shot language pairs and back-translation.

### 7.1 Analysis of the proposed approach on more distant language pairs

To see whether and to what extent our approach generalizes to more distant language pairs, we also analyze the performance of the proposed approach on (ML↔HI, TA↔HI, and TE↔HI). Malayalam, Tamil, and Telugu belong to the Dravidian family, and Hindi is from the Indo-Aryan family. We note that translating between these three Dravidian languages and Hindi still leads to improvement, considering both chrF2 and BLEU scores. The results are shown in table 16.

<sup>15</sup>Although there are communities in South Asia and probably in other countries who actually speak the refined version of Urdu, much more similar to its written or literary form.





**Table 16.** Experiments on distant language pairs.

Model	BLEU	chrF2	BLEU	chrF2	BLEU	chrF2
	HI → ML		HI → TA		HI → TE	
Guzmán <i>et al</i> [45]	<b>5.12</b>	30	7.57	41	<b>7.19</b>	26
Proposed	3.61	<b>32</b>	<b>7.86</b>	<b>44</b>	4.56	<b>27</b>
	ML → HI		TA → HI		TE → HI	
Guzmán <i>et al</i> [45]	9.08	29	14.55	37	7.97	27
Proposed	<b>9.96</b>	<b>33</b>	<b>15.43</b>	<b>40</b>	<b>9.09</b>	<b>30</b>

Bold indicates better scores

**Table 17.** Applying on zero-shot language pairs.

Model	HI → BHO		BHO → HI		HI → MAG		MAG → HI	
	BLEU	chrF2	BLEU	chrF2	BLEU	chrF2	BLEU	chrF2
Guzmán <i>et al</i> [45]	<b>3.34</b>	14	4.58	22	1.67	13	4.86	19
Proposed	3.13	<b>17</b>	<b>5.72</b>	<b>27</b>	<b>2.68</b>	<b>18</b>	<b>5.32</b>	<b>25</b>

**Table 18.** Experiments on back-translation.

Model	GU→HI				HI→GU			
	BLEU	chrF2	TER	WER	BLEU	chrF2	TER	WER
Guzmán <i>et al</i> [45] + BT(monolingual data)	34.26	55	0.564	58.24	28.32	54	0.619	62.47
Proposed + BT(monolingual data)	35.62	59	0.554	57.39	29.29	58	0.604	61.73

## 7.2 Unsupervised settings

We also demonstrate the proposed approach under unsupervised scenarios on zero-shot language pairs, Bhojpuri-Hindi and Magahi-Hindi, for which no cleaned parallel training corpora is available<sup>16</sup>. The validation datasets for zero-shot experiments are collected from LoResMT 2020 shared tasks<sup>17</sup>. For training the model, we use NE↔HI language pairs and use language transfer on zero-shot pairs to evaluate the model on validation datasets. The reason behind using NE↔HI language pairs for training the model in unsupervised experiments on Bhojpuri-Hindi and Magahi-Hindi is the higher similarity between NE↔HI language pairs with both Bhojpuri-Hindi and Magahi-Hindi zero-shot language pairs based on [69]. The results are shown in table 17, demonstrating the improvement in unsupervised settings also.

## 7.3 Back-translation

Finally we report results on using the approach along with Back-Translation, which has been shown to benefit machine translation for very low resource languages. We selected Gujarati↔Hindi language pairs for performing Back-Translation (BT) with the proposed approach. With Back-Translation also, the proposed approach shows an improvement of BLEU point +0.97 on HI→GU and +1.36 on GU→HI language pairs, as shown in table 18.

## 8. Conclusion and future Scope

In this work, we have proposed a simple but effective MT system approach by encoding the source and target script into an intermediate representation, WX-notation, that helps the models to be learnt in a common phonetic and orthographic space. This language projection reduces the surface complexity of the algorithm and allows the neural network to better model the relationships between languages to provide an improved translation. Further, we have investigated these results by estimating the similarities and complexities of language pairs and individual

<sup>16</sup>We do have parallel corpus, which is currently being cleaned and sentence aligned and will be available in the near future.

<sup>17</sup><https://sites.google.com/view/loresmt>

languages to verify that our results are consistent and agree with the intuitively known facts about the closeness or distances between various language pairs. Moreover, this approach works well under unsupervised settings and works fine for some distant language pairs. The proposed approach improves baseline approaches by 0.01 BLEU points to 11.46 BLEU points.

In future, we plan to extend this approach in the ways described below:

- a. *Multilingual NMT system* Since the proposed approach transforms all the Indian language scripts into a common notation called WX, this conversion favours the subword embeddings to work as character embedding. It may be, therefore, more beneficial to implement this approach in the multilingual system(s) for all Indian languages.
- b. *BART, MBART, and other representations* We tried the MBART-based translation of Gujarati to Hindi and Hindi to Gujarati, and the results are worse than a vanilla transformer. So, we plan to extend the proposed approach to more representations like BART, MBART, and other state-of-the-art representation techniques for Deep Learning.
- c. *Dravidian languages and the rest of the Indo-Aryan language family* We also plan to extend the proposed approach to the Dravidian language family and the rest of the Indo-Aryan languages.

## References

- [1] Booth A D 1955 Machine translation of languages, fourteen essays. Technology Press of the Massachusetts Institute of Technology and Wiley, New York
- [2] Banik D, Ekbal A, Bhattacharyya P and Bhattacharyya S 2019 Assembling translations from multi-engine machine translation outputs. *Appl. Soft Comput.* 78: 230–239
- [3] Koehn P 2009 Statistical machine translation. Cambridge University Press, Cambridge. <https://doi.org/10.1017/CBO9780511815829>.
- [4] Banik D, Ekbal A and Bhattacharyya, P 2020 Statistical machine translation based on weighted syntax-semantics. *Sādhanā*, 45: 1–12
- [5] Banik D 2021 Phrase table re-adjustment for statistical machine translation. *Int. J. Speech Technol.* 24: 903–911
- [6] Sutskever I, Vinyals O and Le Q V 2014 Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* 27: 3104–3112
- [7] Bao W, Zhang J, Pan J and Yin X 2022 A novel chinese dialect TTS frontend with non-autoregressive neural machine translation. arXiv preprint [arXiv:2206.04922](https://arxiv.org/abs/2206.04922).
- [8] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L and Polosukhin I 2017 Attention is all you need. *Adv. Neural Inf. Process. Syst.* 30: 5998–6008
- [9] Devlin J, Chang M W, Lee K and Toutanova K 2019 BERT: Pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Volume 1 (Long and Short Papers), pp. 4171–4186
- [10] Lan Z, Chen M, Goodman S, Gimpel K, Sharma P and Soricut R 2020 ALBERT: a lite BERT for self-supervised learning of language representations. In: *Proc. ICLR 2020*
- [11] Liu Y, Gu J, Goyal N, Li X, Edunov S, Ghazvininejad M, Lewis M and Zettlemoyer L 2020 Multilingual denoising pre-training for neural machine translation. *Trans. Assoc. Comput. Linguist.* 8: 726–742
- [12] Barrault L, Bojar O, Costa-jussà M R, Federmann C, Fishel M, Graham Y, Haddow B, Huck M, Koehn P, Malmasi S, Monz C, Müller M, Pal S, Post M and Zampieri M 2019 Findings of the 2019 conference on machine translation (WMT19). In: *Proceedings of the Fourth Conference on Machine Translation* (Volume 2: Shared Task Papers, Day 1), pp. 1–61
- [13] Cieri C, Maxwell M, Strassel S and Tracey J 2016 Selection criteria for low resource language programs. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 4543–4549.
- [14] Sitaram S 2015 Pronunciation modeling for synthesis of low resource languages. PhD thesis. Carnegie Mellon University, Pittsburgh
- [15] Emeneau M B 1956 India as a linguistic area. *Language* 32: 3–16
- [16] Diwakar S, Goyal P and Gupta R 2010 Transliteration among Indian languages using WX notation. In: *Proceedings of the conference on natural language processing 2010*. Saarland University Press, pp. 147–150
- [17] Singh A K 2006 A computational phonetic model for Indian language scripts. In: *Constraints on Spelling Changes: Fifth International Workshop on Writing Systems*, pp. 1–19
- [18] Singh A K 2010 Modeling and application of linguistic similarity. PhD thesis. IIIT, Hyderabad, India
- [19] Singh A K and Surana H 2007 Using a single framework for computational modeling of linguistic similarity for solving many NLP problems. In: *EUROLAN 2007 Summer School. Alexandru Ioan Cuza University of Iasi*, pp. 46
- [20] Bharathi Raja C, Rani P, Arcan M and McCrae J P 2021 A survey of orthographic information in machine translation. *SN Comput. Sci.* 2: 1–19
- [21] Singh A K, Rama T and Dasigi P 2009 A computational model of the phonetic space and its applications. LTRC IIIT, Hyderabad
- [22] Kumar A, Pratap A and Singh A K 2023 Generative adversarial neural machine translation for phonetic languages via reinforcement learning. *IEEE Trans. Emerg. Top. Comput. Intell.* 7: 190–199
- [23] Kumar A, Pratap A, Singh A K and Saha S 2022 Addressing domain shift in neural machine translation via reinforcement learning. *Expert Syst. Appl.* 201: 1117039
- [24] Madaan L, Sharma S and Singla P 2020 Transfer learning for related languages: submissions to the WMT20 similar language translation task. In: *Proc. Fifth Conference on Machine Translation*, pp. 402–408
- [25] Mujadia V and Sharma D 2020 NMT based similar language translation for Hindi-Marathi. In: *Proc. Fifth Conference on Machine Translation*, pp. 414–417
- [26] Rathinasamy K, Singh A, Sivasambagupta B, Prasad Neerchal P and Sivasankaran V 2020 Infosys machine translation

- system for WMT20 similar language translation task. In: *Proc. Fifth Conference on Machine Translation*, pp. 437–441
- [27] Laskar S R, Pakray P and Bandyopadhyay S 2019 Neural machine translation: Hindi-Nepali. In: *Proc. Fourth Conference on Machine Translation*, pp. 202–207
- [28] Ojha A K, Rani P, Bansal A, Chakravarthi B R, Kumar R and McCrae J P 2020 NUIG-Panlingua-KMI Hindi-Marathi MT systems for similar language translation task @ WMT 2020. In: *Proc. Fifth Conference on Machine Translation*, pp. 418–423
- [29] Kumar A, Baruah R, Mundotiya R K and Singh A K 2020 Transformer-based neural machine translation system for Hindi–Marathi: WMT20 shared task. In: *Proc. Fifth Conference on Machine Translation*, pp. 393–395
- [30] Balashov Y 2022 The boundaries of meaning: a case study in neural machine translation. *Inquiry*, 1–34
- [31] Pal S and Zampieri M 2020 Neural machine translation for similar languages: the case of Indo-Aryan languages. In: *Proc. Fifth Conference on Machine Translation*, pp. 424–429
- [32] Przystupa M and Abdul-Mageed M 2019 Neural machine translation of low-resource and similar languages with backtranslation. In: *Proc. Fourth Conference on Machine Translation*, pp. 224–235
- [33] Kakwani D, Kunchukuttan A, Golla S, Gokul N C, Bhat-tacharyya A, Khapra M M and Kumar P 2020 IndicNLP-Suite: monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. *Find. Assoc. Comput. Linguist. EMNLP 2020*: 4948–4961
- [34] Dabre R, Shrotriya H, Kunchukuttan A, Puduppully R, Khapra M and Kumar P 2022 IndicBART: a pre-trained model for Indic natural language generation. *Find. Assoc. Comput. Linguist. ACL 2022*: 1849–1863
- [35] Klein G, Kim Y, Deng Y, Senellart J and Rush A M 2017 OpenNMT: open-source toolkit for neural machine translation. In: *Proc. ACL 2017, System Demonstrations*, pp. 67–72
- [36] Johnson M, Schuster M, Le Q V, Krikun M, Wu Y, Chen Z, Thorat N, Viégas F, Wattenberg M, Corrado G, Hughes M and Dean J 2017 Google’s multilingual neural machine translation system: enabling zero-shot translation. *Trans. Assoc. Comput. Linguist.* 5: 339–351
- [37] Luong T, Pham H and Manning C D 2015 Effective approaches to attention-based neural machine translation. *Proc. EMNLP 2015*: 1412–1421
- [38] Currey A and Heafield K 2019 Incorporating source syntax into transformer-based neural machine translation. In: *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pp. 24–33
- [39] Raganato A, Scherrer Y and Tiedemann J 2020 Fixed encoder self-attention patterns in transformer-based machine translation. *Find. ACL: EMNLP 2020*: 556–568
- [40] Lewis M, Liu Y, Goyal N, Ghazvininejad M, Mohamed A, Levy O, Stoyanov V and Zettlemoyer L 2020 BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proc. ACL 2020*: 7871–7880
- [41] Edunov S, Ott M, Auli M and Grangier D 2018 Understanding back-translation at scale. *Proc. EMNLP 2018*: 489–500
- [42] Hoang V C D, Koehn P, Haffari G and Cohn T 2018 Iterative back-translation for neural machine translation. In: *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pp. 18–24
- [43] Trubetzkoy N S 1928 Proposition 16. In: *Acts of the First International Congress of Linguists*, pp. 17–18
- [44] Haspelmath M 2001 The European linguistic area: standard average European. In: *Halbband Language Typology and Language Universals 2*. edited by Teilband. Berlin: De Gruyter Mouton, pp. 1492–1510
- [45] Guzmán F, Chen P J, Ott M, Pino J, Lample G, Koehn P, Chaudhary V and Ranzato M A 2019 The FLORES evaluation datasets for low-resource machine translation: Nepali-English and Sinhala-English. *Proc. EMNLP-IJCNLP 2019*: 6098–6111
- [46] Gillon B S 1995 Review of Natural language processing: a Paninian perspective by Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal. Prentice-Hall of India 1995. *Comput. Linguist.* 21: 419–421
- [47] Sennrich R, Haddow B and Birch A 2016 Neural machine translation of rare words with subword units. *Proc. ACL 2016*: 1715–1725
- [48] Kudo T 2018 Subword regularization: improving neural network translation models with multiple subword candidates. *Proc. ACL 2018*: 66–75
- [49] Philip J, Siripragada S, Namboodiri V P and Jawahar C V 2021 Revisiting low resource status of indian languages in machine translation. In: *8th ACM IKDD CODS and 26th COMAD*, pp. 178–187
- [50] Haddow B and Kirefu F 2020 PMIndia– a collection of parallel corpora of languages of India. arXiv e-prints. [arXiv: 2001.09907](https://arxiv.org/abs/2001.09907)
- [51] Tiedemann J 2012 Parallel data, tools and interfaces in OPUS. *Proc. LREC 2012*: 2214–2218
- [52] Kudo T and Richardson J 2018 SentencePiece: a simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In: *Proc. EMNLP 2018: System Demonstrations*, pp. 66–71
- [53] Ott M, Edunov S, Baevski A, Fan A, Gross S, Ng N, Grangier D and Auli M 2019 fairseq: a fast, extensible toolkit for sequence modeling. In: *Proceedings of NAACL-HLT 2019: Demonstrations*, pp. 48–53
- [54] Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A and Herbst E 2007 Moses: open source toolkit for statistical machine translation. In: *Proc. 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pp. 177–180
- [55] Och F J and Ney H 2003 A systematic comparison of various statistical alignment models. *Comput. Linguist.* 29: 19–51.
- [56] Heafield K 2011 KenLM: faster and smaller language model queries. In: *Proc. Sixth Workshop on Statistical Machine Translation*, pp. 187–197
- [57] Och F J 2003 Minimum error rate training in statistical machine translation. In: *Proc. 41st Annual Meeting of the Association for Computational Linguistics*, pp. 160–167
- [58] Papineni K, Roukos S, Ward T and Zhu W J 2002 Bleu: a Method for Automatic Evaluation of Machine Translation. *Proc. ACL 2002*: 311–318
- [59] Virpioja S and Grönroos S 2015 LeBLEU: N-gram-based translation evaluation score for morphologically complex

- languages. In: *Proc. Tenth Workshop on Statistical Machine Translation*, pp. 411–416
- [60] Banik D, Ekbal A and Bhattacharyya P 2018 Wuplebleu: the wordnet-based evaluation metric for machine translation. In: *Proc. 15th International Conference on Natural Language Processing*, pp. 104–108
- [61] Snover M, Dorr B, Schwartz R, Micciulla L and Makhoul J 2006 A study of translation edit rate with targeted human annotation. In: *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pp. 223–231
- [62] Popović M 2015 chrF: character n-gram F-score for automatic MT evaluation. In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395
- [63] Mundotiya R K, Singh M K, Kapur R, Mishra S and Singh A K 2021 Linguistic resources for Bhojpuri, Magahi, and Maithili: statistics about them, their similarity estimates, and baselines for three applications. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 20: 1–37
- [64] Rama T and Singh A K 2009 From bag of languages to family trees from noisy corpus. In: *Proceedings of the International Conference RANLP-2009*, pp. 355–359
- [65] Denoual E and Lepage Y 2005 BLEU in characters: towards automatic MT evaluation in languages without word delimiters. In: *Companion Volume to the Proceedings of Conference including Posters/Demos and Tutorial Abstracts*, pp. 79–84
- [66] Bharati A, Rao K P, Sangal R and Bendre S M 2000 Basic statistical analysis of corpus and cross comparison among corpora. In: *Proceedings of the International Conference on Natural Language Processing*, pp. 10
- [67] Bentz C and Alikaniotis D 2016 The word entropy of natural languages. [arXiv. 1606.06996](https://arxiv.org/abs/1606.06996)
- [68] Shannon C E and Weaver W 1949 The mathematical theory of communication. The University of Illinois Press, Urbana
- [69] Kumar A, Mundotiya R K, Pratap A and Singh A K 2022 TLSPG: transfer learning-based semi-supervised pseudo-corpus generation approach for zero-shot translation. *J. King Saud Univ. Comput. Inf. Sci.* 34: 6552–6563