Exploiting Language Relatedness for Low Web-Resource Language Model Adaptation: An Indic Languages Study

Yash Khemchandani[1*] **Sarvesh Mehtani**[1*] **Vaidehi Patil**[1]
Abhijeet Awasthi[1] **Partha Talukdar**[2] **Sunita Sarawagi**[1]
[1]Indian Institute of Technology Bombay, India
[2]Google Research, India
{yashkhem,smehtani,awasthi,sunita}@cse.iitb.ac.in
vaidehipatil@ee.iitb.ac.in, partha@google.com

Abstract

Recent research in multilingual language models (LM) has demonstrated their ability to effectively handle multiple languages in a single model. This holds promise for low web-resource languages (LRL) as multilingual models can enable transfer of supervision from high resource languages to LRLs. However, incorporating a new language in an LM still remains a challenge, particularly for languages with limited corpora and in unseen scripts. In this paper we argue that *relatedness* among languages in a language family may be exploited to overcome some of the corpora limitations of LRLs, and propose RelateLM. We focus on Indian languages, and exploit relatedness along two dimensions: (1) *script* (since many Indic scripts originated from the Brahmic script), and (2) *sentence structure*. RelateLM uses transliteration to convert the unseen script of limited LRL text into the script of a Related Prominent Language (RPL) (Hindi in our case). While exploiting similar sentence structures, RelateLM utilizes readily available bilingual dictionaries to pseudo translate RPL text into LRL corpora. Experiments on multiple real-world benchmark datasets provide validation to our hypothesis that using a related language as pivot, along with transliteration and pseudo translation based data augmentation, can be an effective way to adapt LMs for LRLs, rather than direct training or pivoting through English.

1 Introduction

BERT-based pre-trained language models (LMs) have enabled significant advances in NLP (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2020). Pre-trained LMs have also been developed for the multilingual setting, where a single multilingual model is capable of handling inputs from many different languages. For example, the Multilingual BERT (mBERT) (Devlin et al., 2019) model was trained on 104 different languages. When fine-tuned for various downstream tasks, multilingual LMs have demonstrated significant success in generalizing *across* languages (Hu et al., 2020; Conneau et al., 2019). Thus, such models make it possible to transfer knowledge and resources from resource rich languages to Low Web-Resource Languages (LRL). This has opened up a new opportunity towards rapid development of language technologies for LRLs.

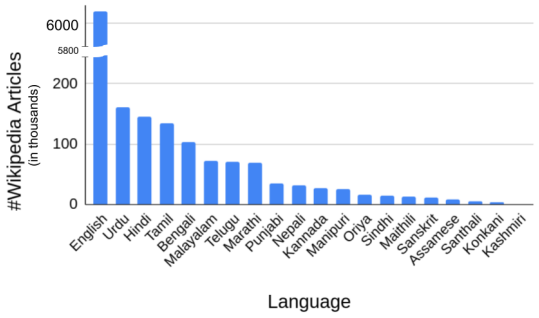


Figure 1: Number of wikipedia articles for top-few Indian Languages and English. The height of the English bar is not to scale as indicated by the break. Number of English articles is roughly 400x more than articles in Oriya and 800x more than articles in Assamese.

However, there is a challenge. The current paradigm for training Mutlilingual LM requires text corpora in the languages of interest, usually in large volumes. However, such text corpora is often available in limited quantities for LRLs. For example, in Figure 1 we present the size of Wikipedia, a common source of corpora for training LMs, for top-few scheduled Indian languages[1] and English. The top-2 languages are just one-fiftieth the size of

*Authors contributed equally

[1]According to Indian Census 2011, more than 19,500 languages or dialects are spoken across the country, with 121 of them being spoken by more than 10 thousand people.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 1312–1323
August 1–6, 2021.

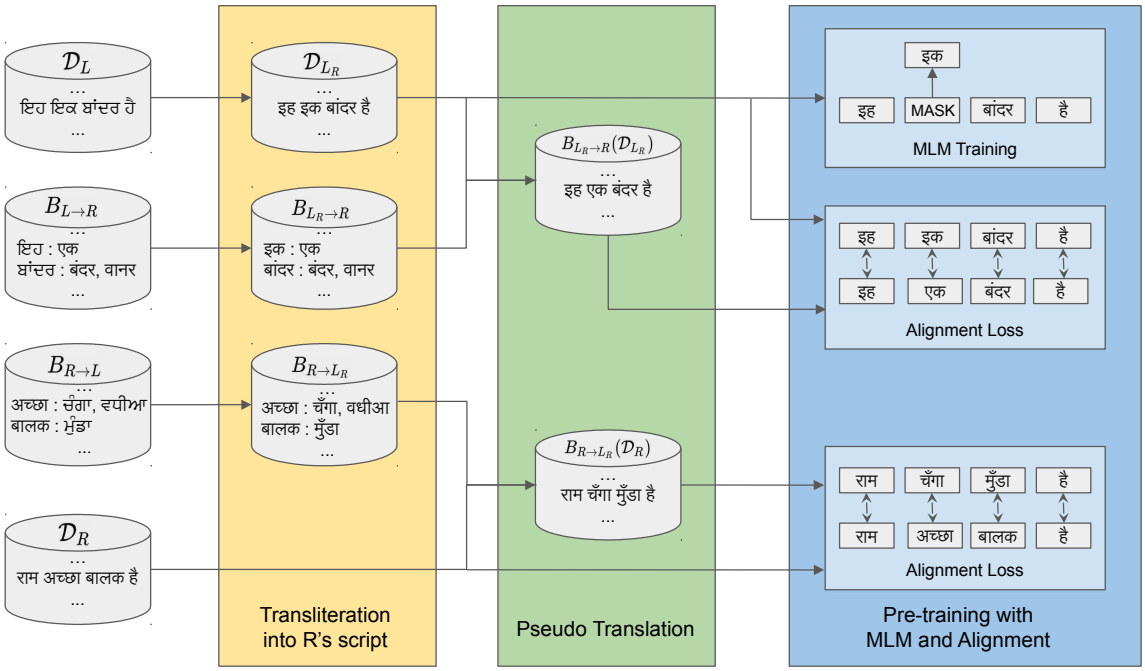


Figure 2: Pre-training with MLM and Alignment loss in RelateLM with LRL L as Punjabi (pa) in Gurumukhi script and RPL R as Hindi (hi) in Devanagari script. RelateLM first transliterates LRL text in the monolingual corpus ($\mathcal{D}_L$) and bilingual dictionaries ($B_{L\to R}$ and $B_{R\to L}$) to the script of the RPL R. The transliterated bilingual dictionaries are then used to pseudo translate the RPL corpus ($\mathcal{D}_R$) and transliterated LRL corpus ($\mathcal{D}_{L_R}$). This pseudo translated data is then used to adapt the given LM $\mathcal{M}$ for the target LRL L using a combination of Masked Language Model (MLM) and alignment losses. For notations and further details, please see Section 3.

English, and yet Hindi is seven times larger than the O(20,000) documents of languages like Oriya and Assamese which are spoken by millions of people. This calls for the development of additional mechanisms for training multilingual LMs which are not exclusively reliant on large monolingual corpora.

Recent methods of adapting a pre-trained multilingual LM to a LRL include fine-tuning the full model with an extended vocabulary (Wang et al., 2020), training a light-weight adapter layer while keeping the full model fixed (Pfeiffer et al., 2020b), and exploiting overlapping tokens to learn embeddings of the LRL (Pfeiffer et al., 2020c). These are general-purpose methods that do not sufficiently exploit the specific relatedness of languages within the same family.

We propose **RelateLM** for this task. RelateLM exploits *relatedness* between the LRL of interest and a **Related Prominent Language** (**RPL**). We focus on Indic languages, and consider Hindi as the RPL. The languages we consider in this paper are related along several dimensions of linguistic typology (Dryer and Haspelmath, 2013; Littell et al., 2017): phonologically, phylogenetically as they are all part of the Indo-Aryan family, geographically, and syntactically matching on key features like the Subject-Object-Verb (SOV) order as against the Subject-Verb-Object (SVO) order in English. Even though the scripts of several Indic languages differ, they are all part of the same Brahmic family, making it easier to design rule-based transliteration libraries across any language pair. In contrast, transliteration of Indic languages to English is harder with considerable phonetic variation in how words are transcribed. The geographical and phylogenetic proximity has lead to significant overlap of words across languages. This implies that just after transliteration we are able to exploit overlap with a Related Prominent Language (RPL) like Hindi. On three Indic languages we discover between 11% and 26% overlapping tokens with Hindi, whereas with English it is less than 8%, mostly comprising numbers and entity names. Furthermore, the syntax-level similarity between languages allows us to generate high quality data augmentation by exploiting pre-existing bilingual dictionaries. We generate pseudo parallel data by converting RPL text to LRL and vice-versa. These allow us to further align the learned embed-

dings across the two languages using the recently proposed loss functions for aligning contextual embeddings of word translations (Cao et al., 2020; Wu and Dredze, 2020).

In this paper, we make the following contributions:

- We address the problem of adding a Low Web-Resource Language (LRL) to an existing pre-trained LM, especially when monolingual corpora in the LRL is limited. This is an important but underexplored problem. We focus on Indian languages which have hundred of millions of speakers, but traditionally understudied in the NLP community.
- We propose RelateLM which exploits relatedness among languages to effectively incorporate a LRL into a pre-trained LM. We highlight the relevance of transliteration and pseudo translation for related languages, and use them effectively in RelateLM to adapt a pre-trained LM to a new LRL.
- Through extensive experiments, we find that RelateLM is able to gain significant improvements on benchmark datasets. We demonstrate how RelateLM adapts mBERT to Oriya and Assamese, two low web-resource Indian languages by pivoting through Hindi. Via ablation studies on bilingual models we show that RelateLM is able to achieve accuracy of zero-shot transfer with limited data (20K documents) that is not surpassed even with four times as much data in existing methods.

The source code for our experiments is available at https://github.com/yashkhem1/RelateLM.

2 Related Work

Transformer (Vaswani et al., 2017) based language models like mBERT (Devlin et al., 2019), MuRIL (Khanuja et al., 2021), IndicBERT (Kakwani et al., 2020), and XLM-R (Conneau et al., 2019), trained on massive multilingual datasets have been shown to scale across a variety of tasks and languages. The zero-shot cross-lingual transferability offered by these models makes them promising for low-resource domains. Pires et al. (2019) find that cross-lingual transfer is even possible across languages of different scripts, but is more effective for typologically related languages. However, recent works (Lauscher et al., 2020; Pfeiffer et al., 2020b; Hu et al., 2020) have identified poor cross-lingual transfer to languages with limited data when jointly pre-trained. A primary reason behind poor transfer is the lack of model's capacity to accommodate all languages simultaneously. This has led to increased interest in adapting multilingual LMs to LRLs and we discuss these in the following two settings.

LRL adaptation using monolingual data For eleven languages outside mBERT, Wang et al. (2020) demonstrate that adding a new target language to mBERT by simply extending the embedding layer with new weights results in better performing models when compared to bilingual-BERT pre-training with English as the second language. Pfeiffer et al. (2020c) adapt multilingual LMs to the LRLs and languages with scripts unseen during pre-training by learning new tokenizers for the unseen script and initializing their embedding matrix by leveraging the lexical overlap w.r.t. the languages seen during pre-training. Adapter (Pfeiffer et al., 2020a) based frameworks like (Pfeiffer et al., 2020b; Artetxe et al., 2020; Üstün et al., 2020) address the lack of model's capacity to accommodate multiple languages and establish the advantages of adding language-specific adapter modules in the BERT model for accommodating LRLs. These methods generally assume access to a fair amount of monolingual LRL data and do not exploit relatedness across languages explicitly. These methods provide complimentary gains to our method of directly exploiting language relatedness.

LRL adaptation by utilizing parallel data When a parallel corpus of a high resource language and its translation into a LRL is available, Conneau and Lample (2019) show that pre-training on concatenated parallel sentences results in improved cross-lingual transfer. Methods like Cao et al. (2020); Wu and Dredze (2020) discuss advantages of explicitly bringing together the contextual embeddings of aligned words in a translated pair. Language relatedness has been exploited in multilingual-NMT systems in various ways (Neubig and Hu, 2018; Goyal and Durrett, 2019; Song et al., 2020). These methods typically involve data augmentation for a LRL with help of a related high resource language (RPL) or to first learn the NMT model for a RPL followed by finetuning on the LRL. Wang et al. (2019) propose a soft-decoupled encoding approach for exploiting subword overlap between LRLs and HRLs to improve encoder representations for LRLs. Gao et al. (2020) address the issue of generating fluent

Percentage Overlap of Words		
LRL	Related Prominent (Hindi)	Distant Prominent (English)
Punjabi	**25.5**	7.5
Gujarati	**23.3**	4.5
Bengali	**10.9**	5.5

Table 1: **Motivation for transliteration**: % overlapping words between transliterated LRL (in Prominent Language's script) and prominent language text. % overlap is defined as the number of common distinct words divided by number of distinct words in the transliterated LRL. Overlap is much higher with Hindi, the Related Prominent Language (RPL), compared to English, the distant language. Overlapping words act as anchors during multilingual pre-training in RelateLM(Section 3.1)

BLEU Scores		
LRL (Target)	Related Prominent (Hindi) (Source)	Distant Prominent (English) (Source)
Punjabi	**24.6**	16.5
Gujarati	**20.3**	12.9
Bengali	**19.3**	12.4

Table 2: **Motivation for pseudo translation**: BLEU scores between pseudo translated prominent language sentences and LRL sentences. BLEU with Hindi, the RPL, is much higher than with English, the distant prominent language highlighting the effectiveness of pseudo translation from a RPL (Section 3.2). English and Hindi dictionary sizes same. For these experiments, we used a parallel corpus across these 5 languages obtained from TDIL (Section 4.1)

LRL sentences in NMT by extending the soft-decoupled encoding approach to improve decoder representations for LRLs. Xia et al. (2019) utilize data augmentation techniques for LRL-English translation using RPL-English and RPL-LRL parallel corpora induced via bilingual lexicons and unsupervised NMT. Goyal et al. (2020) utilize transliteration and parallel data from related Indo-Aryan languages to improve NMT systems. Similar to our approach they transliterate all the Indian languages to the Devanagri script. Similarly, Song et al. (2020) utilize Chinese-English parallel corpus and transliteration of Chinese to Japanese for improving Japanese-English NMT systems via data augmentation.

To the best of our knowledge no earlier work has explored the surprising effectiveness of transliteration to a related existing prominent language, for learning multilingual LMs, although some work exists in NMT as mentioned above.

3 Low Web-Resource Adaptation in RelateLM

Problem Statement and Notations Our goal is to augment an existing multilingual language model $\mathcal{M}$, for example mBERT, to learn representations for a new LRL L for which available monolingual corpus $\mathcal{D}_L$ is limited. We are also told that the language to be added is related to another language R on which the model $\mathcal{M}$ is already pre-trained, and is of comparatively higher resource. However, the script of $\mathcal{D}_L$ may be distinct from the scripts of existing languages in $\mathcal{M}$. In this section we present strategies for using this knowledge to better adapt $\mathcal{M}$ to L than the existing baseline of fine-tuning $\mathcal{M}$ using the standard masked language model (MLM) loss on the limited monolingual data $\mathcal{D}_L$ (Wang et al., 2020). In addition to the monolingual data $\mathcal{D}_R$ in the RPL and $\mathcal{D}_L$ in the LRL, we have access to a limited bilingual lexicon $B_{L\rightarrow R}$ that map a word in language L to a list of synonyms in language R and vice-versa $B_{R\rightarrow L}$.

We focus on the case where the RPL, LRL pairs are part of the Indo-Aryan language families where several levels of relatedness exist. Our proposed approach, consists of three steps, viz., Transliteration to RPL's script, Pseudo translation, and Adaptation through Pre-training. We describe each of these steps below. Figure 2 presents an overview of our approach.

3.1 Transliteration

First, the scripts of Indo-Aryan languages are part of the same Brahmic script. This makes it easier to design simple rule-based transliterators to convert a corpus in one script to another. For most languages transliterations are easily available. Example, the Indic-Trans Library [2] (Bhat et al., 2015). We use $\mathcal{D}_{L_R}$ to denote the LRL corpus after transliterating to the script of the RPL. We then propose to further pre-train the model $\mathcal{M}$ with MLM on the transliterated corpus $\mathcal{D}_{L_R}$ instead of $\mathcal{D}_L$. Such a strategy could provide little additional gains over the baseline, or could even hurt accuracy, if the two languages were not sufficiently related. For languages in the Indo-Aryan family because of strong phylogenetic and geographical overlap, many words across the two languages overlap and preserve the

[2] https://github.com/libindic/indic-trans

same meaning. In Table 1 we provide statistics of the overlap of words across several transliterated Indic languages with Hindi and English. Note that for Hindi the fraction of overlapping words is much higher than with English which are mostly numbers, and entity names. These overlapping words serve as anchors to align the representations for the non-overlapping words of the LRL that share semantic space with words in the RPL.

3.2 Pseudo Translation with Lexicons

Parallel data between a RPL and LRL language pair has been shown to be greatly useful for efficient adaptation to LRL (Conneau and Lample, 2019; Cao et al., 2020). However, creation of parallel data requires expensive supervision, and is not easily available for many low web-resource languages. Back-translation is a standard method of creating pseudo parallel data but for low web-resource languages we cannot assume the presence of a well-trained translation system. We exploit the relatedness of the Indic languages to design a pseudo translation system that is motivated by two factors:

- First, for most geographically proximal RPL-LRL language pairs, word-level bilingual dictionaries have traditionally been available to enable communication. When they are not, crowd-sourcing creation of word-level dictionaries[3] requires lower skill and resources than sentence level parallel data. Also, word-level lexicons can be created semi-automatically (Zhang et al., 2017) (Artetxe et al., 2019) (Xu et al., 2018).
- Second, Indic languages exhibit common syntactic properties that control how words are composed to form a sentence. For example, they usually follow the Subject-Object-Verb (SOV) order as against the Subject-Verb-Object (SVO) order in English.

We therefore create pseudo parallel data between R and L via a simple word-by-word translation using the bilingual lexicon. In a lexicon a word can be mapped to multiple words in another language. We choose a word with probability proportional to its frequency in the monolingual corpus $\mathcal{D}_L$. We experimented with a few other methods of selecting words that we discuss in Section 4.4. In Table 2 we present BLEU scores obtained by our pseudo translation model of three Indic languages from Hindi and from English. We observe much high BLEU for translation from Hindi highlighting the syntactic relatedness of the languages.

Let $(\mathcal{D}_R, B_{R\to L_R}(\mathcal{D}_R))$ denote the parallel corpus formed by pseudo translating the RPL corpus via the transliterated RPL to LRL lexicon. Likewise let $(\mathcal{D}_{L_R}, B_{L_R\to R}(\mathcal{D}_{L_R}))$ be formed by pseudo translating the transliterated low web-resource corpus via the transliterated LRL to RPL lexicon.

3.3 Alignment Loss

The union of the two pseudo parallel corpora above, collectively called $\mathcal{P}$, is used for fine-tuning $\mathcal{M}$ using an alignment loss similar to the one proposed in (Cao et al., 2020). This loss attempts to bring the multilingual embeddings of different languages closer by aligning the corresponding word embeddings of the source language sentence and the pseudo translated target language sentence. Let $\mathcal{C}$ be a random batch of source and (pseudo translated) target sentence pairs from $\mathcal{P}$, i.e. $\mathcal{C} = ((\mathbf{s}^1, \mathbf{t}^1), (\mathbf{s}^2, \mathbf{t}^2), ..., (\mathbf{s}^N, \mathbf{t}^N))$, where $\mathbf{s}$ and $\mathbf{t}$ are the source and target sentences respectively. Since our parallel sentences are obtained via word-level translations, the alignment among words is known and monotonic. Alignment loss has two terms:

$\mathcal{L} = \mathcal{L}_{align} + \mathcal{L}_{reg}$ where $\mathcal{L}_{align}$ is used to bring the contextual embeddings closer and $\mathcal{L}_{reg}$ is the regularization loss which prevents the new embeddings from deviating far away from the pre-trained embeddings. Each of these are defined below:

$$\mathcal{L}_{align} = \sum_{(\mathbf{s},\mathbf{t})\in\mathcal{C}} \sum_{i=1}^{\#\text{word}(\mathbf{s})} ||f(\mathbf{s}, l_{\mathbf{s}}(i)) - f(\mathbf{t}, l_{\mathbf{t}}(i))||_2^2$$

$$\mathcal{L}_{reg} = \sum_{(\mathbf{s},\mathbf{t})\in\mathcal{C}} \left(\sum_{j=1}^{\#\text{tok}(\mathbf{s})} ||(f(\mathbf{s}, j) - f_0(\mathbf{s}, j)||_2^2 + \sum_{j=1}^{\#\text{tok}(\mathbf{t})} ||f(\mathbf{t}, j) - f_0(\mathbf{t}, j)||_2^2 \right)$$

where $l_{\mathbf{s}}(i)$ is the position of the last token of i-th word in sentence $\mathbf{s}$ and $f(\mathbf{s}, j)$ is the learned contextual embedding of token at j-th position in sentence $\mathbf{s}$, i.e, for $\mathcal{L}_{align}$ we consider only the last tokens of words in a sentence, while for $\mathcal{L}_{reg}$ we consider all the tokens in the sentence. $f_0(\mathbf{s}, j)$ denotes the fixed pre-trained contextual embedding of the token at j-th position in sentence $\mathbf{s}$. $\#\text{word}(\mathbf{s})$ and $\#\text{tok}(\mathbf{s})$ are the number of (whole) words and tokens in sentence $\mathbf{s}$ respectively.

[3] Wiktionary is one such effort

4 Experiments

We carry out the following experiments to evaluate RelateLM's effectiveness in LRL adaptation:

- First, in the full multilingual setting, we evaluate whether RelateLM is capable of extending mBERT with two unseen low-resource Indic languages: Oriya (unseen script) and Assamese (seen script). (Section 4.2)
- We then move to the bilingual setting where we use RelateLM to adapt a model trained on a single RPL to a LRL. This setting allowed us to cleanly study the impact of different adaptation strategies and experiment with many RPL-LRL language pairs. (Section 4.3)
- Finally, Section 4.4, presents an ablation study on dictionary lookup methods, alignment losses, and corpus size.

We evaluate by measuring the efficacy of zero-shot transfer from the RPL on three different tasks: NER, POS and text classification.

4.1 Setup

LM Models We take m-BERT as the model $\mathcal{M}$ for our multilingual experiments. For the bilingual experiments, we start with two separate monolingual language models on each of Hindi and English language to serve as $\mathcal{M}$. For Hindi we trained our own Hi-BERT model over the 160K monolingual Hindi Wikipedia articles using a vocab size of 20000 generated using WordPiece tokenizer. For English we use the pre-trained BERT model which is trained on almost two orders of magnitude Wikipedia articles and more. When the LRL is added in its own script, we use the bert-base-cased model and when the LRL is added after transliteration to English, we use the bert-base-uncased model.

LRLs, Monolingual Corpus, Lexicon As LRLs we consider five Indic languages spanning four different scripts. Monolingual data was obtained from Wikipedia as summarized in Table 4. We extend m-BERT with two unseen low web-resource languages: Assamese and Oriya. Since it was challenging to find Indic languages with task-specific labeled data but not already in m-BERT, we could not evaluate on more than two languages. For the bilingual model experiments, we adapt each of Hi-BERT and English BERT with three different languages: Punjabi, Gujarati and Bengali. For these languages we simulated the LRL setting by downsampling their Wikipedia data to 20K documents. For experiments where we require English monolingual data for creating pseudo translations, we use a downsampled version of English Wikipedia having the same number of documents as the Hindi Wikipedia dump.

Dataset Split	Lang	Number of Sentences		
		NER	POS	TextC.
Train Data RPL	en	20.0	56.0	27.0
	hi	5.0	53.0	25.0
Val Data RPL	en	10.0	14.0	3.8
	hi	1.0	13.0	4.0
Test Data LRL	pa	0.2	13.4	7.9
	gu	0.3	14.0	8.0
	bn	1.0	9.7	5.8
	as	-	14.0	8.0
	or	0.2	4.0	7.6

Table 3: Statistics of Task-specific Datasets. All numbers are in thousands.

LRL	#Docs	Scripts	hi-Lexicon		en-Lexicon	
			Fw	Bw	Fw	Bw
pa	20	Gurumukhi	53	65	18	15
gu	20	Gujarati	29	43	18	10
bn	20	As-Bangla	23	40	12	10
or	20	Oriya	18	18	18	18
as	7	As-Bangla	19	17	19	17

Table 4: Statistics of resources used for LRLs in the experiments. All the numbers are in thousands. #Docs represents number of documents for each language. For each language, hi-Lexicon and en-Lexicon report sizes of bilingual Hindi and English dictionaries respectively in either direction. Fw represents the direction from a LRL to hi or en. Hindi uses the Devanagri script with a vocab size of 20K. For all other languages the vocab size is fixed at 10K. As-Bangla refers to the Bengali-Assamese script.

The addition of a new language to $\mathcal{M}$ was done by adding 10000 tokens of the new language generated by WordPiece tokenization to the existing vocabulary, with random initialization of the new parameters. For all the experiments, we use libindic's indictrans library (Bhat et al., 2015) for transliteration. For pseudo translation we use the union of Bilingual Lexicons obtained from CFILT [4] and Wiktionary [5] and their respective sizes for each language are summarized in Table 4

Tasks for zero-shot transfer evaluation After adding a LRL in $\mathcal{M}$, we perform task-specific fine-

[4] https://www.cfilt.iitb.ac.in/
[5] https://hi.wiktionary.org/wiki/

LRL Adaptation	Prominent Language	Punjabi			Gujarati			Bengali		
		NER	POS	TextC.	NER	POS	TextC.	NER	POS	TextC.
mBERT	-	41.7	86.3	64.2	39.8	87.8	65.8	70.8	83.4	75.9
EBERT (Wang et al., 2020)	en	19.4	48.6	33.6	14.5	56.6	37.8	31.2	50.7	32.7
RelateLM−PseudoT		38.6	58.1	54.7	15.3	58.5	57.2	**68.8**	59.8	58.6
EBERT (Wang et al., 2020)	hi	28.2	78.6	51.4	14.8	69.0	48.1	34.0	**73.2**	45.6
RelateLM−PseudoT		65.1	77.3	76.1	39.6	80.2	79.1	56.3	69.9	77.5
RelateLM		**66.9**	**81.3**	**78.6**	**39.7**	**82.3**	**79.8**	57.3	71.7	**78.7**

Table 5: Different Adaptation Strategies evaluated for zero-shot transfer (F1-score) on NER, POS tagging and Text Classification after fine-tuning with the Prominent Language (English or Hindi). mBERT, which is trained with much larger datasets and more languages is not directly comparable, and is presented here just for reference.

tuning on the RPL separately for three tasks: NER, POS and Text classification. Table 3 presents a summary of the training, validation data in RPL and test data in LRL on which we perform zero-shot evaluation. We obtained the NER data from WikiANN (Pan et al., 2017) and XTREME (Hu et al., 2020) and the POS and Text Classification data from the Technology Development for Indian Languages (TDIL)[6]. We downsampled the TDIL data for each language to make them class-balanced. The POS tagset used was the BIS Tagset (Sardesai et al., 2012). For the English POS Dataset, we had to map the PENN tagset in to the BIS tagset. We have provided the mapping that we used in the Appendix (B)

Methods compared We contrast RelateLM with three other adaptation techniques: (1) EBERT (Wang et al., 2020) that extends the vocabulary and tunes with MLM on $\mathcal{D}_L$ as-is, (2) RelateLM without pseudo translation loss, and (3) m-BERT when the language exists in m-BERT.

Training Details For pre-training on MLM we chose batch size as 2048, learning rate as 3e-5 and maximum sequence length as 128. We used whole word masking for MLM and BertWordPieceTokenizer for tokenization. For pre-training Hi-BERT the duplication was taken as 5 with training done for 40K iterations. For all LRLs where monolingual data used was 20K documents, the duplication factor was kept at 20 and and training was done for 24K iterations. For Assamese, where monolingual data was just 6.5K documents, a duplication factor of 60 was used with the same 24K training iterations. The MLM pre-training was done on Google v3-8 Cloud TPUs.

For alignment loss on pseudo translation we chose learning-rate as 5e-5, batch size as 64 and maximum sequence length as 128. The training was done for 10 epochs also on Google v3-8 Cloud TPUs. For task-specific fine-tuning we used learning-rate 2e-5 and batch size 32, with training duration as 10 epochs for NER, 5 epochs for POS and 2400 iterations for Text Classification. The models were evaluated on a separate RPL validation dataset and the model with the minimum F1-score, accuracy and validation loss was selected for final evaluation for NER, POS and Text Classification respectively. All the fine-tuning experiments were done on Google Colaboratory. The results reported for all the experiments are an average of 3 independent runs.

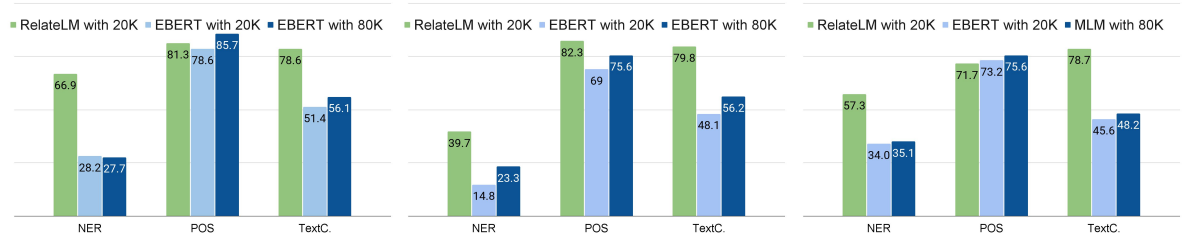
[6] https://www.tdil-dc.in

LRL adaptation	Prominent Language	NER	POS	TextC.
Oriya				
RelateLM−PseudoT	en	14.2	72.1	63.2
RelateLM		16.4	74.1	62.7
EBERT (Wang et al., 2020)	hi	10.8	71.7	53.1
RelateLM−PseudoT		22.7	74.7	76.5
RelateLM		**24.7**	**75.2**	**76.7**
Assamese				
RelateLM−PseudoT	en	-	78.2	74.8
RelateLM		-	77.4	74.7
EBERT (Wang et al., 2020)	hi	-	71.9	78.6
RelateLM−PseudoT		-	**79.4**	79.8
RelateLM		-	79.3	**80.2**

Table 6: mBERT+LRL with different adaptation strategies evaluated on NER, POS tagging and Text Classification with both English and Hindi as the fine-tuning languages. Accuracy metric is F1.

4.2 Multilingual Language Models

We evaluate RelateLM's adaptation strategy on mBERT, a state of the art multilingual model with two unseen languages: Oriya and Assamese. The script of Oriya is unseen whereas the script of Assamese is the same as Bengali (already in m-BERT). Table 6 compares different adaptation strategies in-



(a) Punjabi (b) Gujarati (c) Bengali

Figure 3: Comparison of F1-score between RelateLM-20K, EBERT-20K and EBERT-80K, where the number after method name indicates pre-training corpus size. We find that RelateLM-20K outperforms EBERT-20K in 8 out of 9 settings, and even outperforms EBERT-80K, which is trained over 4X more data, in 7 out of 9 settings.

cluding the option of treating each of Hindi and English as RPL for transliteration into. For both LRLs, transliterating to Hindi as RPL provides gains over EBERT that keeps the script as-is and English transliteration. We find that these gains are much more significant for Oriya than Assamese, which could be because Oriya is a new script. Further augmentation with pseudo translations with Hindi as RPL, provides significant added gains. We have not included the NER results for Assamese due to the absence of good quality evaluation dataset.

4.3 Bilingual Language Models

For more extensive experiments and ablation studies we move to bilingual models. Table 5 shows the results of different methods of adapting $\mathcal{M}$ to a LRL with Hi-BERT and BERT as two choices of $\mathcal{M}$. We obtain much higher gains when the LRL is transliterated to Hindi than to English or keeping the script as-is. This suggests that transliteration to a related language succeeds in parameter sharing between a RPL and a LRL. Note that the English BERT model is trained on a much larger English corpus than the Hi-BERT model is trained on the Hindi corpus. Yet, because of the relatedness of the languages we get much higher accuracy when adding transliterated data to Hindi rather than to English. Next observe that pre-training with alignment loss on pseudo translated sentence pairs improves upon the results obtained with transliteration. This shows that pseudo translations is a decent alternative when a parallel translation corpora is not available.

Overall, we find that RelateLM provides substantial gains over the baseline. In many cases RelateLM is even better than mBERT which was pretrained on a lot more monolingual data in that language. Among the three languages, we obtain lowest gains for Bengali since the phonetics of Bengali varies to some extent from other Indo-Aryan languages, and Bengali shows influence from Tibeto-Burman languages too (Kunchukuttan and Bhattacharyya, 2020). This is also evident in the lower word overlap and lower BLEU in Table 1 and Table 2 compared to other Indic languages. We further find that in case of Bengali, the NER results are best when Bengali is transliterated to English rather than Hindi, which we attribute to the presence of English words in the NER evaluation dataset.

Loss	Dict Lookup	NER	POS	Text C.
Punjabi				
MSE	first	62.4	80.0	77.6
MSE	max	68.2	81.3	77.6
MSE	root-weighted	64.9	78.9	76.9
MSE	weighted	66.9	81.3	78.6
cstv	weighted	68.2	80.8	79.4
Gujarati				
MSE	first	39.2	83.3	78.6
MSE	max	39.1	82.5	80.4
MSE	root-weighted	39.7	82.6	79.9
MSE	weighted	39.7	82.3	79.8
cstv	weighted	40.2	84.0	81.6
Bengali				
MSE	first	55.5	68.0	74.0
MSE	max	56.2	70.3	79.7
MSE	root-weighted	56.4	69.3	76.5
MSE	weighted	57.3	71.7	78.7
cstv	weighted	56.6	67.6	76.5

Table 7: Usefulness of Bilingual Dictionaries with MSE(Mean Squared Error Loss) and cstv(Contrastive Loss) evaluated on NER, POS tagging and Text Classification in RelateLM.

4.4 Ablation Study

Methods of Dictionary Lookups We experimented with various methods of choosing the translated word from the lexicon which may have multiple entries for a given word. In Table 7 we compare four methods of picking entries: *first* - en-

try at first position, *max*-entry with maximum frequency in the monolingual data, *weighted* - entry with probability proportional to that frequency and *root-weighted* - entry with probability proportional to the square root of that frequency. We find that these four methods are very close to each other, with the *weighted* method having a slight edge.

Alignment Loss We compare the MSE-based loss we used with the recently proposed contrastive loss (Wu and Dredze, 2020) for $\mathcal{L}_{align}$ but did not get any significant improvements. We have provided the results for additional experiments in the Appendix (A)

Increasing Monolingual size In Figure 3 we increase the monolingual LRL data used for adapting EBERT four-fold and compare the results. We observe that even on increasing monolingual data, in most cases, by being able to exploit language relatedness, RelateLM outperforms the EBERT model with four times more data. These experiments show that for zero-shot generalization on NLP tasks, it is more important to improve the alignment among languages by exploiting their relatedness, than to add more monolingual data.

5 Conclusion and Future Work

We address the problem of adapting a pre-trained language model (LM) to a Low Web-Resource Language (LRL) with limited monolingual corpora. We propose RelateLM, which explores *relatedness* between the LRL and a Related Prominent Language (RPL) already present in the LM. RelateLM exploits relatedness along two dimensions – script relatedness through transliteration, and sentence structure relatedness through pseudo translation. We focus on Indic languages, which have hundreds of millions of speakers, but are understudied in the NLP community. Our experiments provide evidence that RelateLM is effective in adapting multilingual LMs (such as mBERT) to various LRLs. Also, RelateLM is able to achieve zero-shot transfer with limited LRL data (20K documents) which is not surpassed even with 4X more data by existing baselines. Together, our experiments establish that using a related language as pivot, along with data augmentation through transliteration and bilingual dictionary-based pseudo translation, can be an effective way of adapting an LM for LRLs, and that this is more effective than direct training or pivoting through English.

Integrating RelateLM with other complementary methods for adapting LMs for LRLs (Pfeiffer et al., 2020b,c) is something we plan to pursue next. We are hopeful that the idea of utilizing relatedness to adapt LMs for LRLs will be effective in adapting LMs to LRLs in other languages families, such as South-east Asian and Latin American languages. We leave that and exploring other forms of relatedness as fruitful avenues for future work.

Acknowledgements We thank Technology Development for Indian Languages (TDIL) Programme initiated by the Ministry of Electronics Information Technology, Govt. of India for providing us datasets used in this study. The experiments reported in the paper were made possible by a Tensor Flow Research Cloud (TFRC) TPU grant. The IIT Bombay authors thank Google Research India for supporting this research. We thank Dan Garrette and Slav Petrov for providing comments on an earlier draft.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019. Bilingual lexicon induction through unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5002–5007, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Irshad Ahmad Bhat, Vandan Mujadia, Aniruddha Tammewar, Riyaz Ahmad Bhat, and Manish Shrivastava. 2015. Iiit-h system submission for fire2014 shared task on transliterated search. In *Proceedings of the Forum for Information Retrieval Evaluation*, FIRE '14, pages 48–53, New York, NY, USA. ACM.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in*

Neural Information Processing Systems, volume 32, pages 7059–7069. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Luyu Gao, Xinyi Wang, and Graham Neubig. 2020. Improving target-side lexical transfer in multilingual neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3560–3566, Online. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2019. Embedding time expressions for deep temporal ordering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4400–4406, Florence, Italy. Association for Computational Linguistics.

Vikrant Goyal, Sourav Kumar, and Dipti Misra Sharma. 2020. Efficient neural machine translation for low-resource languages via exploiting related languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 162–168, Online. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. inlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Proceedings of EMNLP 2020*.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Anoop Kunchukuttan and Pushpak Bhattacharyya. 2020. Utilizing language relatedness to improve machine translation: A case study on languages of the indian subcontinent.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880, Brussels, Belgium. Association for Computational Linguistics.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of ACL 2017*, pages 1946–1958.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020b. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2020c. Unks everywhere: Adapting multilingual language models to new scripts. *CoRR*, abs/2012.15562.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Madhavi Sardesai, Jyoti Pawar, Shantaram Walawalikar, and Edna Vaz. 2012. BIS annotation standards with reference to Konkani language. In *Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing*, pages 145–152, Mumbai, India. The COLING 2012 Organizing Committee.

Haiyue Song, Raj Dabre, Zhuoyuan Mao, Fei Cheng, Sadao Kurohashi, and Eiichiro Sumita. 2020. Pre-training via leveraging assisting languages for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 279–285, Online. Association for Computational Linguistics.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. UDapter: Language adaptation for truly Universal Dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Xinyi Wang, Hieu Pham, Philip Arthur, and Graham Neubig. 2019. Multilingual neural machine translation with soft decoupled encoding. *arXiv preprint arXiv:1902.03499*.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Do explicit alignments robustly improve multilingual encoders? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4471–4482, Online. Association for Computational Linguistics.

Mengzhou Xia, Xiang Kong, Antonios Anastasopoulos, and Graham Neubig. 2019. Generalized data augmentation for low-resource translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5786–5796, Florence, Italy. Association for Computational Linguistics.

Ruochen Xu, Yiming Yang, Naoki Otani, and Yuexin Wu. 2018. Unsupervised cross-lingual transfer of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2465–2474, Brussels, Belgium. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

Appendix

A Additional Experiments with Contrastive Loss

Apart from MSE loss, we also experimented with the recently proposed Contrastive Loss. We present the results of using contrastive loss with various methods of dictionary lookups as described in Section 4 of the paper, in Table 8

Loss	Dict Lookup	NER	POS	Text C.
Punjabi				
cstv	first	73.1	80.7	75.5
cstv	max	62.1	79.8	73.4
cstv	root-weighted	72.1	78.5	77.9
cstv	weighted	68.2	80.8	79.4
Gujarati				
cstv	first	39.9	83.3	80.4
cstv	max	38.9	84.1	80.8
cstv	root-weighted	39.9	83.1	76.0
cstv	weighted	40.2	84.0	81.6
Bengali				
cstv	first	56.2	67.7	77.2
cstv	max	56.9	69.2	76.9
cstv	root-weighted	58.5	71.1	70.9
cstv	weighted	56.6	67.6	76.5

Table 8: Evaluations on NER, POS tagging and Text Classification in RelateLM using Contrastive Loss with different methods of dictionary lookup

B POS Tagset mapping between Penn Treebank Tagset and BIS Tagset

For the POS experiments involving m-BERT as the base model, we fine-tune our trained model with both English and Hindi training data and calculate zero-shot results on the target language. However, the English dataset that we used was annotated using Penn Treebank Tagset while the rest of the languages were annotated using BIS Tagset. We came up with a mapping between the Penn Tags and the BIS Tags so that the English POS dataset becomes consistent with the Hindi counterpart. Table 9 contains the mapping that we used for the said conversion. Note that since we are using top-level tags (e.g Pronouns) instead of sub-level tags

(e.g Personal Pronouns, Possessive Pronouns) for the POS classification, the mapping is also done to reflect the same.

Penn Tagset	BIS Tagset	Penn Tagset	BIS Tagset
CC	CC	CD	QT
EX	RD	FW	RD
IN	PSP	JJ	JJ
JJR	JJ	JJS	JJ
LS	QT	MD	V
NN	N	NNS	N
NNP	N	NNPS	N
POS	PSP	PRP	PR
PRP$	PR	RB	RB
RBR	RB	RBS	RB
RP	RP	SYM	RD
TO	RP	UH	RP
VB	V	VBD	V
VBG	V	VBN	V
VBP	V	VBZ	V
WP	PR	WP$	PR
AFX	RD	-LRB-	RD
-RRB-	RD	# . , $ “ ( ) : - ‘’ ‘	RD
PDT	all, half: QT such: DM ”default”: QT	WDT	which, that : PR whatever: RP ”default”: PR
DT	some, every, both, all, another, a, an: QT this, these, the: DM those, that: PR ”default”: QT	WRB	how,wherever, when, where: PR whenever: RB why: RB ”default” : PR

Table 9: Tagset mapping between Penn Treebank and BIS. For some tags in Penn treebank (e.g. DT), we decided that a one-to-many mapping was appropriate based on a word-level division