Subword-based Cross-lingual Transfer of Embeddings from Hindi to Marathi and Nepali

Niyati Bafna and Zdeněk Žabokrtský,
Charles University, Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics
niyatibafna13@gmail.com,zabokrtsky@ufal.mff.cuni.cz

Abstract

Word embeddings are growing to be a crucial resource in the field of NLP for any language. This work introduces a novel technique for static subword embeddings transfer for Indic languages from a relatively higher resource language to a genealogically related low resource language. We primarily work with Hindi-Marathi, simulating a low-resource scenario for Marathi, and confirm observed trends on Nepali. We demonstrate the consistent benefits of unsupervised morphemic segmentation on both source and target sides over the treatment performed by fastText. Our best-performing approach uses an EM-style approach to learning bilingual subword embeddings; we also show, for the first time, that a trivial "copy-and-paste" embeddings transfer based on even perfect bilingual lexicons is inadequate in capturing language-specific relationships. We find that our approach substantially outperforms the fastText baselines for both Marathi and Nepali on the Word Similarity task as well as WordNet-Based Synonymy Tests; on the former task, its performance for Marathi is close to that of pretrained fastText embeddings that use three orders of magnitude more Marathi data.

1 Introduction

Subword-level embeddings are useful for many tasks, but require large amounts of monolingual data to train. While about 15 Indian languages such as Hindi, Bengali, and Marathi have the required magnitudes of data, most Indian languages are highly under-resourced; they have very little monolingual data and almost no parallel data, and not much digitization. For example, to the best of our knowledge, Marwadi, spoken by 14M people, has no available monolingual corpus; Konkani, spoken by about 3M people, has a monolingual corpus containing 3M tokens, and no parallel data.[1] However, many of these languages have very close syntactic, morphological, and lexical connections to surrounding languages including the mentioned high-resource languages. Our approach aims to leverage these connections in order to build embeddings for these low-resource languages, in the hope that this will aid further development of other NLP tools for these languages.[2]

While there is a growing interest in shifting towards contextual embeddings with BERT (Devlin et al., 2018), as well as extending them to low-resource languages, static embeddings retain value in being lightweight and less computationally expensive, especially as studies show that they can perform comparably to contextual embeddings in certain settings (Arora et al., 2020) and encode similar linguistic information (Miaschi and Dell'Orletta, 2020). Thus, an efficient method to develop static embeddings for languages with minimal or no NLP research remains a relevant step to building a basic range of resources in these languages. In this study, we primarily work with Hindi-Marathi as our genealogically and culturally related language pair, and use asymmetric resources (large data for Hindi, artificially small monolingual data for Marathi), confirming our final results for Nepali.

Most languages of the Indic/Indo-Aryan family, spoken over most parts of North India, are morphologically rich, including Hindi, Marathi, and Nepali. This means that while related language pairs may have a high number of cognates, these may be "disguised" by surrounding inflectional or derivational morphemes. Therefore, even with an identical underlying syntactic structure, lexical correspondences between languages may be obscured or rendered incongruent. Further, when working with small data, the corpus frequencies of

[1] The Opus Corpus (Tiedemann, 2012), one of the most popular collection of parallel texts, contains no parallel data for languages such as Konkani or Bundeli.

[2] While some languages may have a little parallel data, we assume none, so as to cater to languages that are just undergoing digitization.

19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 61 - 71
July 14, 2022 ©2022 Association for Computational Linguistics

fully inflected surface forms would be much less reliable than those of stem and affix morphemes, intuitively resulting in a less robust embeddings transfer. These factors add weight to the intuition that many Indic languages share morpheme-level correspondences with each other. This motivated us to apply unsupervised morphemic segmentation on both the source and target language data; we demonstrate the benefits of doing so in our evaluations. Note that this also makes it natural to work with subword-level embeddings rather than word embeddings; studies show that the former have an advantage over word embeddings especially for morphologically rich languages. (Chaudhary et al., 2018; Zhu et al., 2019b; Li et al., 2018).

The idea of the transfer is to project the low-resource language (LRL) subwords into a shared bilingual space with the high-resource language (HRL). We first attempt a trivial transfer that simply finds the "closest" HRL subword for each LRL subword, and copies its embedding. We demonstrate that this approach, while tempting, is not enough to capture the relationships between even identical words in both languages; embeddings spaces appear to encode more complex information that this approach would suggest. For our best performing approach, we adapt the EM-style algorithm described in Artetxe et al. (2017) to a subword-setting; the algorithm alternately optimizes the distance between pairs belonging to a bilingual mapping, and generates a bilingual mapping between words from the resulting bilingual embeddings. As far as we know, our work is the first to apply this algorithm in the context of embeddings transfer. We compare the resulting bilingual embeddings to data-intensive fastText models using the Word Similarity and WordNet-Based Synonymy Tests for Marathi; for Nepali, we evaluate on the latter task due to the lack of a Word Similiarity dataset.

2 Previous Work

2.1 Subwords in Embedding Spaces

In a seminal work, Bojanowski et al. (2017) present fastText embeddings, that work at a subword level by representing words as bags of chargrams. Kudo and Richardson (2018) present a subword tokenizer for neural text processing, and Kudo (2018) shows the benefits of using multiple subword segmentations in neural machine translation, especially in low-resource settings. Zhu et al. (2019b) look at the segmentation of a word, such as using chargrams, Byte Pair Encoding (BPE) (Gage, 1994; Sennrich et al., 2016), Morfessor, as well as the composition of the subword embeddings (addition, averaging, etc.) to construct the final word vector, and conclude that the best performing configuration is highly language and task dependent. A subsequent work (Zhu et al., 2019a) focuses on LRLs and finds the combination of BPE and addition largely robust, although they once again note language-dependent variability. They also find that encoding "affix" information with positional embeddings is beneficial, hinting that the embedding space may distinguish the importance of different kinds of subwords.

2.2 Cross-lingual embeddings

The problem of learning bilingual embeddings has usually been studied in a symmetric resources scenario. Xu et al. (2018) propose an unsupervised method of mapping two sets of monolingual static embeddings into a shared space; they present results for English paired with Spanish, Chinese, and French, evaluated on the bilingual lexicon induction and Word Similarity tasks. Chaudhary et al. (2018) experiment with joint and transfer learning for training bilingual subword embeddings for pairs of Indic LRLs from scratch, by projecting different scripts into the International Phonetic Alphabet (IPA). Kayi et al. (2020) present an extension of the BiSkip cross-lingual learning objective that leverages subword information to train English-paired bilingual embeddings for LRLs, using around 30K parallel sentences. We describe Artetxe et al. (2017) in some detail below, since we use this algorithm in our approach. There is also growing interest in multilingual contextual embeddings (Devlin et al., 2018; Kakwani et al., 2020; Ruder et al., 2019) such as multilingual BERT; Wang et al. (2020) propose an approach to extend multilingual BERT to low-resource languages without retraining it, Pfeiffer et al. (2020) suggest an approach towards incorporating previously unseen scripts into a multilingual BERT model.

2.3 Bilingual Lexicon Induction

This task is closely related to that of embeddings transfer; we see that these two tasks leverage each other in the literature. Older works such as Koehn and Knight (2002) and Haghighi et al. (2008) use monolingual features such as frequency heuristics, orthographic features, tags, and context vectors in order to find bilingual mappings for mainly European language pairs. Hauer et al. (2017) use word2vec embeddings (Mikolov et al., 2013) in order to iteratively train a translation matrix.

2.4 Summarizing Artetxe et al. (2017)

Artetxe et al. (2017) present an EM-style approach to training bilingual embeddings from monolingual embeddings without parallel data; however, it assumes high quality monolingual embeddings for both languages trained on at least 1 billion word corpora each. Given the two sets of word embeddings, they find a bilingual dictionary D by choosing the closest target word for each source word with respect to the cosine distance between source and target word embeddings. In the next step, they use the dictionary D to calculate a linear transformation matrix that minimizes the sum of cosine distances of the embeddings of all word pairs in D. They apply an orthogonality constraint on the transformation matrix in order to preserve monolingual invariance i.e. to prevent the degradation of the monolingual relationships in the resulting embedding space. These steps are repeated until convergence.

3 Note on languages

Hindi, spoken by about 340M people, is related to other large Indic languages such as Marathi, Punjabi, and Bangla, and has 48 recognized "dialects" over India, which makes it a good choice for the HRL in this project. Hindi is written in the Devanagari script, which is also used for over 120 other (often related) languages, including Marathi and Nepali. Hindi, Marathi, and Nepali share morpho-syntactic properties common within the Indic language family, such as (split) ergativity and primarily SOV structure with reordering allowed under constraints. For all three languages, (some) nouns inflect for case and number, verbs inflects for tense, number, gender, and person, and adjectives inflect for gender and number, and case in Hindi and Marathi. Some differences are that Marathi and Nepali exhibit more agglutinative tendencies than Hindi, both allowing suffix stacking with certain boundary changes. For example, a Marathi token may be a sequence of verb+nominalizing-morpheme+case-marker or noun+postposition+genitive, whereas Hindi separates these morphemes into tokens in many cases (while still exhibiting inflectional and some derivational morphology). See Figure 1.

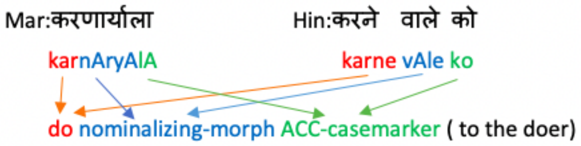

Figure 1: Tokens in Marathi and Hindi. The stem for "do" is the same (i.e. "kar") in both languages; Marathi uses one token whereas Hindi uses three.

4 Data and Resources

4.1 Training Data

For Hindi, we used 1M sentences containing roughly 18M tokens from the HindMonoCorp 0.5 (Bojar et al., 2014). For Marathi, we used 50K sentences containing 0.8M tokens from the IndicCorp Marathi monolingual dataset (Kakwani et al., 2020)[3], and for Nepali, we use 1.4M tokens from the Wortschatz corpus (Goldhahn et al., 2012). We choose these numbers for Marathi and Nepali because it seems to be the ballpark of the amount of monolingual data collected for newly digitized Indic languages.[4] All the above corpora, as well as following resources, are in the Devanagari script.

4.2 Pretrained Embeddings

We use pretrained fastText embeddings for Hindi, presented by Grave et al. (2018), in line with the assumption that we have good quality resources for the HRL. These embeddings (HIN-PRETR-2G[5]) are trained on the *Wikipedia* corpus as well as *Common Crawl*, containing a total of about 2G tokens. We also use the pretrained fastText embeddings (MAR-PRETR-334M, NEP-PRETR-393M) presented in the same work, solely for the purpose of evaluation; these embeddings are trained on 334M tokens (Marathi) and 393M tokens (Nepali).

4.3 Evaluation datasets

4.3.1 Word Similarity Dataset

A Word Similarity dataset is a set of word pairs, each annotated by humans according to the de-

[3] Note that we do not lemmatize our data; good-quality lemmatizers are a scarce resource that we cannot assume for the LRL.

[4] See https://www.ldcil.org/resourcesTextCorp.aspx for efforts on collecting data on under-resourced languages such as Bodo, Dogri, Santhali, etc.

[5] We use the following shorthand to refer to our models unless otherwise specified: <language>-<method_label>-<tokens_of_training_data>. There may be two data slots in the case of bilinugal embeddings, containing amount of Marathi/Nepali and Hindi data respectively.

gree of similarity (integers ranging from 1 to 10) between the two words. Evaluation is usually performed by finding the cosine similarity between the two words vectors, and calculating the Spearman's Rank Correlation between the human and model "similarity" judgments for all word pairs. We report this correlation multiplied by 100.

We present results on the Marathi Word Similarity dataset presented by Akhtar et al. (2017), containing 104 word pairs. This dataset is created by translating a subset of the WordSimilarity-353 English dataset into Marathi by native Marathi speakers, and re-evaluating the similarity scores by 8 native speaker annotators.[6]

4.3.2 WordNet-Based Synonymy Tests

We also perform WordNet-Based Synonymy Tests (WBST) (Piasecki et al., 2018) for Marathi and Nepali. A WBST consists of a set of "questions" consisting of one "query word", and N options, all of which occur MIN times in the corpus. One of the options is a synonym or closely related to the query word, while the rest are "distracters", or randomly selected words. The task is to identify the synonym; we do this by calculating the cosine distances between the query word vector and each of the options and selecting the closest. The reported score is the percentage of correctly answered questions. We use the IndoWordNet,[7] built by Sinha et al. (2006); Debasri et al. (2002), for generating the WBST.

5 Segmentation

5.1 Motivation

Due to the fusional/agglutinative nature of the languages, as well as the morphological and tokenization differences as discussed in Section 3, we apply unsupervised morphemic segmentation to both source and target side data. This is motivated by the need to handle data scarcity on the LRL side, since fully inflected tokens are much rarer than their constituent subwords; we see that the unsegmented Marathi and Nepali data have 100K and 140K distinct tokens respectively, but only 20K and 40K distinct "morphemes", respectively, post-segmentation.

The morphemic segmentation is also an attempt to isolate the morphs in the language data since, according to our hypothesis, it is easier to find correspondences between the two languages at this level rather than at the token level. This is clear in the fact that 50% of the subwords in the Marathi segmented data also occur in the Hindi corpus, whereas for the unsegmented data, this is only 20% of tokens. For Nepali, the difference is lower, in particular, 40% and 20% respectively. See Figures 2 and 3 for a visualisation of the frequency range of the common subwords over that of all subwords in the Hindi and Marathi corpora respectively. Finally, we see that while the mean length of subwords in the Marathi and Hindi corpora are 5.02 and 4.72 respectively, the mean length of common subwords is 3.95; this indicates that shorter subwords are (naturally) more likely to be common than longer counterparts. We see similar numbers for Nepali.

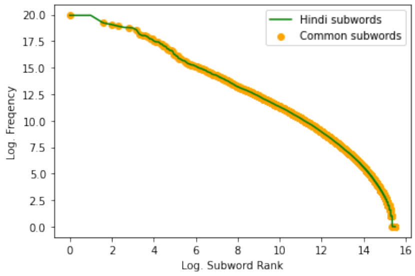

Figure 2: Shared subwords in Hindi and Marathi corpora; numbering up to 17.23% of the total # of subwords in the Hindi corpus. Common subwords are well-distributed over the range.

The most obvious fallout to attempting static embedding transfer at the subword level is morphological homonymy i.e. morphs that may have more than one "meaning", and therefore deserve more than one static embedding.[8] There are many examples of such morphs, e.g. /te/ is both the (free) third person plural pronoun, as well as the (bound) first person female present tense morph in Marathi.

5.2 Tools and evaluation

We experimented with BPE and Morfessor and decided to use the latter, since BPE seemed unable to preserve longer morphs regardless of parameter settings. However, this decision may vary according to language type. We perform a manual evaluation

[6] not available for Nepali.

[7] See http://www.cfilt.iitb.ac.in/WordNet/webmwn/

[8] This is of course a general problem with static embeddings; however, it is exacerbated at the level of subwords, especially imperfectly segmented, since they are shorter and more multifunctional, as it were, than longer lexemes.

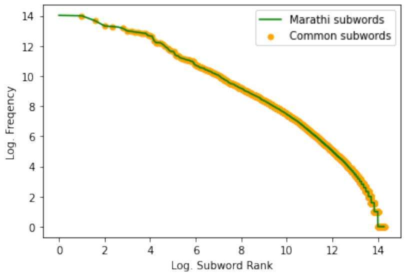

Figure 3: Shared Hindi-Marathi subwords, numbering up to 40.39% of the total # of subwords in the Marathi corpus. As in Figure 2, we see a distribution over the frequency range with the bulk in the mid-to-low frequency range.

of the resulting Marathi segmentation[9] over 100 words sampled by frequency, which shows 72.6% precision and 64.9% recall. True and false positives are counted with respect to morph boundaries rather than at the word level, and each boundary prediction contributes equally to precision/recall. 61% of words are segmented completely correctly.

6 Approach

As baseline, we train fastText models on the available tokenized data (MAR-BASE-0.8M, NEP-BASE-1.4M) for both languages. We work with 300-dimensional embeddings for all experiments.[10]

6.1 Normalized Edit Distance (NED) Approach (Marathi)

Our initial experiments were performed on Hindi-Marathi. The NED approach is based on finding a bilingual subword-level mapping; it takes advantage of the high number of cognates and borrowings between related languages as well as the common script. Its primary intuition is that since the languages share not only lexical items but also syntactic and morphological properties, embedding vectors can essentially be "copied" over to the LRL from the HRL.

For each Marathi morph, we choose the Hindi subword with the minimum NED from it. NED is calculated in the following way:

$$NED(l,h) = \frac{edit\_distance(l,h)}{max(length(l), length(h))}$$

To obtain the embedding of any Marathi word, we first segment it. For each subword, we look for the closest Hindi subword by NED, and retrieve the corresponding Hindi subword embedding. Finally, we compose the subword embeddings, using addition, to give the word embedding. See Algorithm 1 for a depiction.[11]

Algorithm 1: NED Approach

```
l_word ← LRL word;
H_EMB ← HRL embeddings;
l_morphs ← $segment\_lrl$(l_word);
  l_subwords_emb ← empty list;
for l_morph in l_morphs do
    h_closest ← $closest\_HRL\_morph$(l_word);
    append(l_subwords_emb, H_EMB(h_closest));
end
l_emb ← $compose\_subwords$(l_subwords_emb);
return l_emb ;
```

6.2 Iterative approach (Marathi, Nepali)

Although the approach presented in Artetxe et al. (2017) is intended to generate bilingual *word* embeddings for equally well-resourced languages (See Section 2.4), we hypothesize that the algorithm will maintain its quality at the subword level for morphologically rich languages; further, that in our data-asymmetry situation, this approach will serve to "transfer" some of the higher quality of the HRL embedding space to the LRL embeddings, by leveraging a bilingual mapping to induce the relationships already encoded in the HRL embeddings.

We apply this approach to both Marathi and Nepali. As the initial set of LRL embeddings, we use fastText vectors trained on available segmented data (MAR-SEGM-0.8M, NEP-SEGM-1.4M). For the HRL, we can use any available resource. We try using pretrained fastText vectors (HIN-PRETR-2G); we also retrain fastText on the segmented Hindi data (HIN-SEGM-18M). For all runs, we set the initial seed dictionary as identical words[12] in the source and target corpora.[13] See Algorithm 2 for a depiction of OOV handling for this approach. For composing the subword embeddings of a word, we tried

[9] The authors do not speak Nepali and are therefore unable to provide a manual evaluation.

[10] Repeating some experiments for 100 dimensional embeddings spaces, we observe similar trends, with a generally lower performance.

[11] Of course, an NED-based approach is highly limited to related words in the language. However, testing it out gives us an interesting insight about cognates and identical words (see Section 9.1)

[12] This is only possible because the languages share a script.

[13] Note that this approach does not use any parallel data or bilingual lexicons; this aligns with our assumptions about parallel data. However, in the case that parallel data does exist, it can be used to find a good quality bilingual seed lexicon in lieu of using identical words; this has been shown to improve the quality of the resulting bilingual embeddings.

Algorithm 2: Bilingual embeddings with MAR-SEGM-0.8M/NEP-SEGM-1.4M as backoff

```
l_word ← LRL word;
L_EMB ← Bilinual LRL embeddings;
L_EMB_backoff ← Monolingual LRL embeddings;
l_morphs ← segment_lrl(l_word);
l_subwords_emb ← empty list;
for l_morph in l_morphs do
    l_morph_emb ← empty list ;
    if l_morph in L_EMB then
        l_morph_emb ← L_EMB(l_word);
    end
    else
        l_morph_emb←L_EMB_backoff(l_morph);
    end
    append(l_subwords_emb, l_morph_emb);
end
l_emb← compose_subwords(l_subwords_emb);
return l_emb ;
```

Approach	Score
MAR-BASE-0.8M	24.64
MAR-SEGM-0.8M	**43.23**
BI-MAR-JOINT-0.8M-18M	35.48

Table 1: Marathi monolingual and Marathi-Hindi Joint results on Marathi WordSim task. Notation of models explained in Section 4.2.

addition, averaging, and picking the first subword embedding while discarding the rest. The idea behind the last method is that this approximates the word stem, and also reduces the noise created by summing different subword embeddings.

7 Results: Word Similarity (Marathi)

7.1 Baseline and Comparison Models

In Table 1, we show the performance of MAR-BASE-0.8M and MAR-SEGM-0.8M. taking motivation from Chaudhary et al. (2018), we also try a joint approach i.e. we train bilingual embeddings jointly on the segmented Hindi and Marathi data (BI-MAR-JOINT-0.8M-18M). We observe that simple segmentation of the data causes an improvement of over 20 points, outdoing not only MAR-BASE-0.8M but MAR-SKIPGR-27M (See Table 2). Surprisingly, the joint model BI-MAR-JOINT-0.8M-18M dips in performance in comparison to the MAR-SEGM-0.8M. We discuss this effect of the Hindi data on the bilingual embeddings in Section 9.1.

In Table 2, we show the performance of pretrained fastText Marathi embeddings mentioned in Section 4.2 (MAR-PRETR-334M), as well as the best performing model score from Akhtar et al. (2017) on this evaluation dataset. Akhtar et al. (2017) test different sets of embeddings including Skip-gram, CBOW (Mikolov et al., 2013) and fastText (Bojanowski et al., 2017) algorithms, all trained on a corpus with 27M tokens, of which the Skip-Gram (MAR-SKIPGR-27M) performed best.

Embeddings	Score
MAR-PRETR-334M	**54.89**
MAR-SKIPGR-27M	41.12
HIN-PRETR-2G	39.94

Table 2: Scores of high-resource Marathi and Hindi models on Marathi WordSim task for comparison.

Embeddings	Identical Word Score
HIN-PRETR-2G	41.17
MAR-PRETR-334M	**50.38**

Table 3: Scores of pretrained embeddings on word pairs from the Marathi WordSim dataset that are identical in both languages

Finally, Table 3 shows the performance of the MAR-PRETR-334M and HIN-PRETR-2G on certain word pairs in the Marathi WordSim dataset such that both words are also used identically in Hindi.[14] These word pairs were manually identified from the Marathi evaluation dataset; we found that there were 64 such word pairs.[15] Surprisingly, we see a significant dip in the performance of HIN-PRETR-2G on these word pairs as compared to MAR-PRETR-334M, indicating that while the word pairs appear identical in both languages to a native speaker, their usage in the corpora or interaction with other words from the language is different.[16]

7.2 Normalized Edit Distance (NED)

Our NED models use only Hindi embeddings, and project Marathi morphs onto Hindi morphs as shown in Algorithm 1. For further simplicity, we also tried a self-mapping; i.e. we simply calculate the (Hindi) embeddings of the Marathi morphs obtained by segmentation, as they are. Note that this

[14] That is, both of the words in the word pair must be both Hindi and Marathi words with the same spelling, and near-identical senses.

[15] Many of these are transliterations of English words. 24 of the total 135 unique words are transliterations, and they occur 40 times i.e. 19.6% times in the 104 word pairs.

[16] Note that HIN-PRETR-2G performs very well on the Hindi WordSim dataset; its monolingual quality is not the problem.

Approach	Score
Bi-Mar-Self-Segm-0.8M-18M	**43.62**
Bi-Mar-Self-Pretr-0.8M-2G	42.72
Bi-Mar-NED-Pretr-0.8M-2G	41.85
Bi-Mar-NED-Segm-0.8M-18M	39.37

Table 4: Scores on Marathi WordSim for self-mapping and NED strategies, using different Hindi embeddings. Notation: Bi-<lrl>-<mapping_method>-<hin_embs>-<lrl_tokens>-<hin_tokens>.

is only possible because Marathi and Hindi share a common script. The resulting embeddings are composed by addition unless otherwise mentioned. See Table 4 for the results on different combinations of embeddings and mappings.

Firstly, we observe that the self-mapping performs better than NED in general.[17] This is unsurprising; NED would only perform better for Marathi words that are cognates with Hindi words and show a slight difference in spelling; it will perform competitively with self-mapping for identical words in Hindi and Marathi. As we discuss in Section 7.1, such words form a large part of the evaluation dataset. As for the remaining words, it seems that the Hindi embeddings are able to capture the meaning of the unknown Marathi morphs, perhaps due to similarities at a subword level. Applying the NED mapping, however, can result in Marathi words being mapped to arbitrary Hindi words that may share no semantics with the Marathi word.

Another interesting observation is that the Bi-Mar-Self-Segm-0.8M-18M performs a little better than Bi-Mar-Self-Pretr-0.8M-2G. This affirms our intuition in Section 5 that segmentation on the Hindi side may facilitate the correspondence between commmon subwords, leading to better performance on a Marathi evaluation set despite orders of magnitude less (Hindi) data.

7.3 Iterative Approach

There are several points of interest in the results, given in Table 5. Firstly, we see that the Bi-Mar-Iter-Segm-0.8M-18M outperforms Bi-Mar-Iter-Pretr-0.8M-2G; i.e. once again, we find that it is better to use embeddings trained on segmented Hindi data for the transfer, even though Hin-Segm-18M is trained on two orders of magnitude fewer data than Hin-Pretr-2G. Since this approach is explicitly bilingual and attempts to project the Marathi and Hindi embeddings into a shared space, this is a much more direct affirmation that the similarities between Hindi and Marathi are best exploited at the subword level from *both* sides. Secondly, we see that the “first-morph” manner of composition does slightly better than summing or averaging[18] the subword embeddings.[19] Finally, note that doubling the amount of Hindi data used to train the initial Hindi embeddings does not help. This indicates that the Hindi data is only useful up to a point.

Approach	Comp.	Score
(Mar-Base-0.8M	-	24.64)
Bi-Mar-Iter-Pretr-0.8M-2G	Sum	44.28
Bi-Mar-Iter-Segm-0.8M-9M	Sum	49.49
Bi-Mar-Iter-Segm-0.8M-18M	Sum	49.21
Bi-Mar-Iter-Segm-0.8M-18M	FM	50.06
Bi-Mar-Iter-Segm-0.8M-36M	FM	**50.10**

Table 5: Iterative approach results on Marathi WordSim task using different sets of Hindi embeddings for the crosslingual transfer. Format of the approach name: Bi-<lrl>-Iter-<hin_embs>-<lrl_tokens>-<hin_tokens>. **Comp.**: Composition function. FM (first morph) refers to the strategy of simply using the embedding of the first morph

8 Results: WordNet-Based Synonymy Tests (Marathi, Nepali)

See Table 6 and Table 7 for the Marathi and Nepali scores respectively. These results confirm some of the findings from the WordSim results for Marathi, while showing similar trends for Nepali. We see once more that segmentation helps: Mar-Segm-0.8M and Nep-Segm-1.4M consistently outperform the baselines; further, the iterative method is the best among the low-resource embeddings. We also note that doubling the Hindi data for the iterative approach (e.g. with Bi-Mar-Iter-0.8M-36M) seems not to have much effect on the resulting embeddings for both Marathi and Nepali. It is interesting to observe that Nepali is slightly less respon-

[17] Note that there is a difference between the self-mapping model and directly applying Hin-Pretr-2G as in Table 2 In the former, we segment the Marathi word ourselves and apply Hindi embeddings to the resulting subwords; in the latter, we leave it up to fastText. We note that the former does better.

[18] We do not report averaging scores since they are almost identical to the summing scores.

[19] This could be for several reasons; for example, if the first subword approximates the root of the word, then it may capture most of the meaning, whereas the remaining information may be irrelevant or add noise.

(MIN, N)	**Test size**	**Mar-Base -0.8M**	**Mar-Segm -0.8M**	**Bi-Mar-Iter -Segm-0.8M-18M**	**Bi-Mar-Iter -Segm-0.8M-36M**	**Mar-Pretr -334M**
(10,6)	1183	51.23	58.92	**61.62**	57.06	**84.70**
(10,5)	1183	51.90	54.78	58.66	**61.54**	**84.87**
(20,6)	684	48.98	53.65	**59.94**	58.19	**84.50**
(20,5)	684	57.89	59.94	**64.47**	64.33	**87.57**
(50,5)	293	58.02	63.14	67.24	**68.94**	**81.23**

Table 6: WBST Results. MIN: min. freq. of the question and options in the corpus, N: number of total options, Test size: number of questions. The two best-performing models have been bolded.

(MIN, N)	**Test size**	**Nep-Base -1.4M**	**Nep-Segm -1.4M**	**Bi-Nep-Iter -Segm-1.4M-18M**	**Bi-Nep-Iter -Segm-1.4M-36M**	**Nep-Pretr -393M**
(10,6)	1414	58.20	63.93	**65.28**	65.06	**74.11**
(10,5)	1414	61.10	67.75	**69.17**	69.10	**76.37**
(20,6)	974	62.32	69.30	**69.71**	69.10	**76.38**
(20,5)	974	63.86	69.51	**70.74**	70.12	**78.23**
(50,5)	451	66.29	70.29	71.62	**71.84**	**77.16**

Table 7: WBST Results for Nepali. Formatted similarly to Table 6.

sive to the iterative approach than Marathi; this can perhaps be explained by its lower shared subword vocabulary with Hindi (approximately 40% as compared to 50% for Marathi-Hindi). Finally, as MIN increases, the performance of the low-resource methods generally increases; they naturally perform better on words seen more frequently in the corpus.

9 Discussion

Some of the clearer findings of our experiments are as regards segmentation and the benefits of a non-trivial bilingual embeddings transfer.

We see repeatedly that segmentation on both sides of the transfer helps the quality of the LRL embeddings. Segmenting the Marathi data causes a large boost in monolingual performance (Table 1); furthermore, when transferring from Hindi embeddings, Bi-Mar-Iter-Segm-0.8M-18M outperforms Bi-Mar-Iter-Pretr-0.8M-2G (Table 5); the Hindi embeddings used in the latter are trained on 2 orders of magnitude higher (unsegmented) data.[20] This suggests that the interaction between the two languages is indeed facilitated at a subword level, validating our bilingual native speaker intuition about the same. We also see that the iterative approach consistently outperforms both monolingual models Mar-Base-0.8M and Mar-Segm-0.8M, indicating that bilingual interaction between the related languages is indeed beneficial. In general, this is a good sign for the project of building NLP tools for low-resource languages, although it invites exploration of the impact of different typologies on the observed bilingual effect.

Finally, we find that, in agreement with the findings of the papers that investigate subword composition functions (Zhu et al., 2019a,b), the best-performing composition function for subword embeddings seems to be task and data dependent; even discarding everything except the first subword seems to work better sometimes than aggregating all subword embeddings.

9.1 Using Hindi data

To the best of our knowledge, this is the first work that clearly demonstrates that a trivial "copy-and-paste" transfer approach, such as our NED models, is not adequate, even when working with two culturally related languages that share a very high percentage of vocabulary as well as morphosyntactic properties. Our experiments with identical words pairs in Table 3 especially show that even identical words that are not false friends may behave dif-

[20] Note that we are talking about performance in terms of the resultant Marathi bilingual embeddings rather than the direct evaluation of the Hindi embeddings.

ferently depending on the language;[21] using Hindi embeddings *directly*, even for identical words, is problematic. We believe that this is an important insight into embeddings transfer that rejects relying on trivial or simplistic approaches.

Many of our experiments are intended to indicate how useful the Hindi data and embeddings are to the LRL; e.g. we evaluate HIN-PRETR-2G directly on the Marathi WordSim task (Table 2), we experiment with different amounts of Hindi data for both tasks (Tables 5 and 6), and we try a self-mapping with the NED model (see Table 4). We see that doubling the amount of Hindi data sometimes even harms performance;[22] we also see that BI-MAR-JOINT-0.8M-18M performs worse than MAR-SEGM-0.8M (see Table 1). In conjunction, these results imply that under the current transfer paradigm, adding more Hindi data may sometimes hurt rather than benefit; too much Hindi data for the purpose of training bilingual embeddings may actually "conceal" Marathi word interactions. We also applied the iterative approach on Konkani-Hindi, with a mere 100K tokens of Konkani data and 18M tokens of Hindi data as before; however, the bilingual effect was less clearly visible with this setup, supporting the need for investigation into the optimal balance of LRL-HRL data. We invite further investigation of this effect.

10 Future Work

This work is intended to be the pilot in a series of similar studies. We hypothesize that we can obtain similar results for other genealogically related LRL-HRL pairs. We intend to repeat these experiments for language pairs (simulating LRL environments) such as Punjabi-Hindi, Assamese-Bengali, Konkani-Marathi, and others. Some of the issues we will be working against are different scripts, morphemic segmentation of typologically different languages, and the lack of evaluation data. We would also like to experiment with the integration of parallel data into this approach. Finally, we also think it would be interesting to extend our solution from a bilingual to a multilingual one, with multiple sources for a target language. This would be highly pertinent in the case of Indic languages, where even major Indic languages may be interconnected, and regional languages may benefit from the resources of more than one HRL.

11 Conclusion

Embeddings transfer from a high-resource language to a low-resource related language is an important task in today's scenario of data inequality across languages. We target a family of geographically and genealogically related languages, including some high-resource languages and other low-resource languages, possibly undergoing digitization and data collection. We take two Indic language pairs, Hindi-Marathi/Nepali, simulating a low-resource scenario for Marathi and Nepali, and present an approach to embeddings transfer that uses very little monolingual data on the LRL side, and no parallel data. We demonstrate the benefits of unsupervised morphemic segmentation on both source and target sides for subword-level embeddings transfer. Our final approach improves substantially over monolingual fastText baselines for the Marathi WordSim task, and the WBST task for Marathi and Nepali. Further, we show that a "copy-and-paste" embeddings transfer fails even with a perfect bilingual dictionary for a closely related language pair, establishing the need for more sophisticated methods of low-resource bilingual transfer.

Acknowledgements

This work has been using data, tools and services provided by the LINDAT/CLARIAH-CZ Research Infrastructure (https://lindat.cz), supported by the Ministry of Education, Youth and Sports of the Czech Republic (Project No. LM2018101).

References

Syed Sarfaraz Akhtar, Arihant Gupta, Avijit Vajpayee, Arjit Srivastava, and Manish Shrivastava. 2017. Word similarity datasets for Indian languages: Annotation and baseline systems. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 91–94, Valencia, Spain. Association for Computational Linguistics.

Simran Arora, Avner May, Jian Zhang, and Christopher

[21] This is to say even if words a and b occur identically and with the same senses in both languages, the word pair (a, b) may have a different relationship depending on the language.

[22] Our particular "doubled" dataset actually shows roughly the percentage of shared subwords as before doubling; it is possible that data introducing new subwords will perform better. However, in any case, it is interesting to note that the transfer is not improved by having more HRL data for the same subwords which we might intuitively hope would help the quality of the HRL embeddings and therefore the transfer.

Ré. 2020. Contextual embeddings: When are they worth it? *CoRR*, abs/2005.09117.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ondřej Bojar, Vojtěch Diatka, Pavel Rychlý, Pavel Straňák, Vít Suchomel, Aleš Tamchyna, and Daniel Zeman. 2014. HindMonoCorp 0.5. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Aditi Chaudhary, Chunting Zhou, Lori Levin, Graham Neubig, David R Mortensen, and Jaime G Carbonell. 2018. Adapting word embeddings to new languages with morphological and phonological subword representations. *arXiv preprint arXiv:1808.09500*.

Chakrabarti Debasri, Narayan Dipak Kumar, Pandey Prabhakar, and Bhattacharyya Pushpak. 2002. Experiences in building the Indo-Wordnet: A Wordnet for Hindi. In *Proceedings of the First Global WordNet Conference*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 759–765.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. *arXiv preprint arXiv:1802.06893*.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: Hlt*, pages 771–779.

Bradley Hauer, Garrett Nicolai, and Grzegorz Kondrak. 2017. Bootstrapping unsupervised bilingual lexicon induction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 619–624.

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 the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4948–4961.

Efsun Sarioglu Kayi, Vishal Anand, and Smaranda Muresan. 2020. Multiseg: Parallel data and subword information for learning bilingual embeddings in low resource scenarios. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 97–105.

Philipp Koehn and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 9–16.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Bofang Li, Aleksandr Drozd, Tao Liu, and Xiaoyong Du. 2018. Subword-level composition functions for learning word embeddings. In *Proceedings of the Second Workshop on Subword/Character LEvel Models*, pages 38–48, New Orleans. Association for Computational Linguistics.

Alessio Miaschi and Felice Dell'Orletta. 2020. Contextual and non-contextual word embeddings: an in-depth linguistic investigation. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 110–119, Online. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. Unks everywhere: Adapting multilingual language models to new scripts. *arXiv preprint arXiv:2012.15562*.

Maciej Piasecki, Gabriela Czachor, Arkadiusz Janz, Dominik Kaszewski, and Paweł Kędzia. 2018. Wordnet-based evaluation of large distributional models for

Polish. In *Proceedings of the 9th Global Wordnet Conference*, pages 229–238, Nanyang Technological University (NTU), Singapore. Global Wordnet Association.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Manish Sinha, Mahesh Reddy, and Pushpak Bhattacharyya. 2006. An approach towards construction and application of multilingual Indo-Wordnet. In *3rd Global Wordnet Conference (GWC 06), Jeju Island, Korea*. Citeseer.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Zihan Wang, Stephen Mayhew, Dan Roth, et al. 2020. Extending multilingual BERT to low-resource languages. *arXiv preprint arXiv:2004.13640*.

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.

Yi Zhu, Benjamin Heinzerling, Ivan Vulić, Michael Strube, Roi Reichart, and Anna Korhonen. 2019a. On the importance of subword information for morphological tasks in truly low-resource languages. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 216–226.

Yi Zhu, Ivan Vulić, and Anna Korhonen. 2019b. A systematic study of leveraging subword information for learning word representations. 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 912–932.