Don't Forget Cheap Training Signals Before Building Unsupervised Bilingual Word Embeddings

Anonymous ACL submission

Abstract

Bilingual Word Embeddings (BWEs) are one of the cornerstones of cross-lingual transfer of NLP models. They can be built using only monolingual corpora without supervision leading to numerous works focusing on unsupervised BWEs. However, most of the current approaches to build unsupervised BWEs do not compare their results with methods based on easy-to-access cross-lingual signals. In this paper, we argue that such signals should always be considered when developing unsupervised BWE methods. The two approaches we find most effective are: 1) using identical words as seed lexicons (which unsupervised approaches incorrectly assume are not available for orthographically distinct language pairs) and 2) combining such lexicons with pairs extracted by matching romanized versions of words with an edit distance threshold. We experiment on thirteen non-Latin languages (and English) and show that such cheap signals work well and that they outperform using more complex unsupervised methods on distant language pairs such as Chinese, Japanese, Kannada, Tamil, and Thai. In addition, we show that our signals are even competitive with the use of high-quality lexicons in supervised approaches. Our results show that these training signals should not be neglected when building BWEs, even for distant languages.

1 Introduction

Bilingual Word Embeddings (BWEs) are useful for many cross-lingual tasks. They can be built effectively even when only a small seed lexicon is available by mapping monolingual embeddings into a shared space. This makes them particularly valuable for low-resources settings (Mikolov et al., 2013). In addition, unsupervised mapping approaches can build BWEs for some languages when no seed lexicon is available. Various unsupervised methods have been proposed (Zhang et al., 2017; Lample et al., 2018; Artetxe et al., 2018; Alvarez-Melis and Jaakkola, 2018; Chen and Cardie, 2018; Hoshen and Wolf, 2018; Mohiuddin and Joty, 2019; Alaux et al., 2019; Dou et al., 2020; Grave et al., 2019; Li et al., 2020) relying on the assumption that embedding spaces are isomorphic. However, with one exception, none of them compare their results with the widely available baseline of using identical words as seed lexicons for bootstrapping with semi-supervised approaches. In this paper however, we argue that such signals should be used as a cheap and effective baseline in the development of future unsupervised methods.

We study two approaches for extracting the initial seed lexicons to build BWEs without relying on expensive dictionaries: (1) relying on identical pairs as proposed by Smith et al. (2017). Previous work assumed such pairs not to be available for language pairs with distinct scripts, hence the development of various unsupervised mapping approaches. We show that, surprisingly, they do appear in large quantities in the monolingual corpora that we use, even for distinct-script pairs. Although identical pairs are noisy, we show that they are sufficient to create good BWEs. In addition, we propose to (2) strengthen identical pairs by extending them with further easily accessible pairs based on romanization and edit distance, which exploits implicit links between languages in the form of approximate word transliteration pairs. We focus on distant language pairs having distinct scripts for many of which unsupervised approaches have failed or had very poor performance so far. For instance, English to Chinese, Japanese, Kannada, Tamil, and Thai, which all obtain a score close to 0 on the Bilingual Dictionary Induction (BDI) task (Vulić et al., 2019). We evaluate the two approaches on thirteen different non-Latin[1] languages paired with English on BDI. We compare our lexicons' performance with unsupervised mapping and the frequently used

[1] We use *(non-)Latin language* here as a short form for *language standardly written in a (non-)Latin script*.

MUSE training lexicons (Lample et al., 2018) and show that our noisy word pairs make it possible to build BWEs for language pairs where unsupervised approaches failed before and give accuracy scores similar to high quality lexicons. Our work calls into question – at least for BDI – the strong trend toward unsupervised approaches in recent literature, similarly to Vulić et al. (2019), given that cheap signals are (i) available and easy to exploit, (ii) sufficient to obtain performance similar to high-quality dictionaries like MUSE and (iii) able to make up for the failure of unsupervised methods. Finally, we analyze which lexicon properties impact performance and show that our lexicon outperform unsupervised methods also for non-English language pairs. Our paper calls for the need to use easily accessible bilingual signals, such as identical and/or transliteration word pairs, as baselines when developing unsupervised BWE approaches.

2 Unsupervised pair extraction

We show that we can extract the seed lexicon needed for mapping systems without the need for labeled data, making up for the failure of unsupervised methods. First, we show that identical pairs do appear in corpora of distant languages and can be exploited. Secondly, we propose a novel method to boost the identical pairs sets by extracting the initial seed lexicon without the need for any bilingual knowledge, starting from monolingual corpora, and using romanization and edit distance.

2.1 Identical pair approach

When dealing with languages with different scripts, identical pairs would seem to be unlikely to occur, which is assumed by unsupervised mapping methods. Smith et al. (2017) form dictionaries from identical strings which appear in both languages but limit their approach to similar languages sharing a common alphabet, such as European ones. Similarly, Lample et al. (2018) refrain from using such identical word pairs assuming they were not available for distant languages. However, there are domains where these pairs are actually available in large quantity; an example is Wikipedia: see the fastText wiki embedding (Bojanowski et al., 2017) statistics in Table 1. Most of these identical pairs are punctuation marks and digits, non-transliterated named entities written in the Latin script or English words (assumingly words of a title) which were not translated in the non-English languages. This is also true for language pairs not including English. In this paper, we build BWEs based on these pairs and show they are sufficient for good BDI results on distant language pairs with distinct scripts.

Lang	ID	Lang	ID	Lang	ID
Ko-Th	17K	Ko-He	11K	He-Th	15K
En-Zh	62K	En-Bn	31K	En-Ar	19K
En-Th	46K	En-Hi	30K	En-Ru	18K
En-Ja	43K	En-Ta	23K	En-He	17K
En-El	35K	En-Kn	21K	En-Ko	15K
En-Fa	32K				

Table 1: # identical pairs per language pair.

2.2 Romanization based augmentation (ID++)

Identical pairs are noisy and may appear in smaller quantities for certain corpora and language pairs (e.g., He-Ko). We proposed our romanization approach that builds the seed lexicon completely automatically and can augment the identical pairs set. We exploit the concept of transliteration and orthographic similarity to find a cheap signal between languages (cf. (Riley and Gildea, 2018; Severini et al., 2020)) and to take advantage of cognates (Chakravarthi et al., 2019). It consists of 3 steps at the end of which we add the identical pairs and run VecMap in a semi-supervised setting.

1. Source candidates First, we generate a list of source language words, which are the candidates to be matched with a word on the target side. We use the English Wikipedia dumps[2] as our monolingual corpus and apply Flair (Akbik et al., 2018) to extract Universal Part-of-Speech (UPOS) tags. We collect all English proper nouns (PROPN) since names are often transliterated between languages. The resulting English proper noun set consists of ≈800K words.

2. Target candidates The language-specific target data are extracted from the vocabulary of the pre-trained Wikipedia fastText embeddings (Bojanowski et al., 2017). The sets are not preprocessed with a POS tagger assuming that such a tool is missing or perform poorly for low-resource languages. Compared to the English proper noun set, the vocabularies are smaller: between 40K and 500K. Then, we romanize the corpora to obtain equivalent words but with only Latin characters – this supports distance-based metrics in step (3). We use Uroman (Hermjakob et al., 2018) for romanization. Examples of romanization are карл

[2] https://dumps.wikimedia.org/ (01.04.2020)

	En-Th	En-Ja	En-Kn	En-Ta	En-Zh
Unsupervised					
1.	0.00	0.96	0.00	0.07	0.07
2.	0.00	0.48	0.00	0.07	0.00
3.	0.00	0.00	0.00	$0.00^{\diamond}$	0.00
Semi-supervised (Artetxe et al., 2018)					
ID	24.40	48.87	22.03	17.93	37.00
Rom.	23.33	48.46	22.90	18.00	0.27
ID++	23.47	49.14	24.23	18.20	35.00
MUSE	24.33	48.73	23.78	18.80	36.53

Table 2: acc@1 on BDI for unsupervised (1: Artetxe et al. (2018), 2: Grave et al. (2019), 3: Mohiuddin and Joty (2019)) and semi-supervised approaches for 5 languages for which unsupervised methods fail. The semi-supervised results are obtained using VecMap with three different initial lexicons: the identical pair set (ID), ID extended with romanization based pairs (ID++) and the MUSE dictionary. We show an ablation study as well, i.e., the romanized pairs only (Rom.). Scores from Mohiuddin et al. (2020) are marked with $^{\diamond}$.

(Russian)→ carl and βαβυλών (Greek) → babylon. Uroman mainly covers 1-1 character correspondences and does not vocalize words for Arabic and Hebrew. In general, its romanization is not as accurate as the transliteration of a neural model. However, neural models need a training corpus of labeled pairs to work well while Uroman only uses the character descriptions from the Unicode table,[3] manually created tables and some heuristics, supporting a large number of languages.

3. Candidate matching For an English noun to find the corresponding target word, the noun is compared with each (romanized) target word based on their orthography. The similarity of two words w_1 and w_2 is defined as $1 - \mathrm{NL}(w_1, w_2)$, where NL is Levenshtein distance (Levenshtein, 1966) divided by the length of the longer string. We select a pair of words if the similarity is ≥ 0.8; that ensures a trade off between number of pairs and quality, based on some manual investigation. We use the Symmetric Delete algorithm to speed up computation, similar to (Riley and Gildea, 2018). It takes the lists of source and target words, and a constant k and identifies all the source-target pairs that are identical after k insertion or deletions.[4]

[3] http://unicode.org/Public/UNIDATA/UnicodeData.txt

[4] We used minimum frequency and minimum length equal to 1, k equals to 2.

The final step is to look up, for each romanized target word, its original non-romanized form.

3 Evaluation

We evaluate our seed lexicons on BDI to show the quality of the BWEs obtained with them. Recent papers (Marchisio et al., 2020) show that there is a direct relationship between BDI accuracy and downstream BLEU for machine translation. Moreover, Sabet et al. (2020) show that good-quality word embeddings directly reflect the performance also for extrinsic tasks like word alignment. We use the VecMap tool to build BWEs since it supports both unsupervised, semi-supervised and supervised techniques (Artetxe et al., 2018). The semi-supervised approach is of particular interest to us since it performs well with small and noisy seed lexicons by iteratively refining them. We use pre-trained Wikipedia fastText embeddings (Bojanowski et al., 2017) as the input monolingual vectors, taking only the 200K most frequent words and using default parameters otherwise. We compare the performance of VecMap using our lexicons with MUSE, a high-quality dictionary. MUSE contains dictionaries for many languages and it was created using a Facebook internal translation tool (Lample et al., 2018). Since Kannada is not supported by MUSE, we use the dictionary provided by Author (2020). We show $acc@1$ scores based on CSLS vector similarity calculated by the MUSE evaluation tool (Lample et al., 2018).[5]

Tables 2 and 3 show accuracy for all language pairs considering English as the source; see Table 7 in Appendix B for the full table containing results in both directions. Table 2 gives scores for language pairs for which unsupervised methods completely diverge (acc@$1 < 1$). We report results for three unsupervised methods (Artetxe et al., 2018; Mohiuddin and Joty, 2019; Grave et al., 2019). In contrast, using identical word pairs as lexicon (ID) or its extension with the romanization based pairs (ID++) with VecMap leads to successful BWEs without any parallel data or manually created lexicons. In addition, scores are even comparable to high-quality dictionaries like MUSE. Looking at results for all language pairs in Table 2 and 3, our sets always obtain results comparable to MUSE (baseline dictionaries), with improvements

[5] We follow Artetxe et al. (2018) work for comparison reasons and did not remove identical pairs from the test sets. However, overlaps between train romanized lexicons and test lexicons correspond to less than 1%.

	Unsup.	ID	Rom.	ID++	MUSE
En-Ar	36.30	40.27	39.33	40.20	39.87
En-Hi	40.20	40.47	39.60	40.20	40.33
En-Ru	44.80	49.13	48.87	49.53	48.80
En-El	47.90	47.87	48.00	48.27	48.00
En-Fa	36.70	37.67	36.80	37.67	38.00
En-He	44.60	44.47	44.53	44.67	45.00
En-Bn	18.20	19.87	19.80	20.13	21.60
En-Ko	19.80	27.92	28.40	28.81	28.94

Table 3: acc@1 on BDI for (best) unsupervised method and semi-supervised VecMap with different initial lexicons. (full table in Appendix B, Table 7).

	Unsup.	ID	Rom.	ID++	PanLex
Th-Ko	0.00	2.81	3.37	3.09	2.95
Th-He	0.00	9.75	0.00	8.86	10.13
Ko-Th	0.00	15.90	14.23	15.26	14.36
Ko-He	14.62	15.68	16.08	16.00	15.11
He-Th	0.00	16.42	0.00	16.54	17.90
He-Ko	14.30	15.39	15.15	15.09	16.06

Table 4: acc@1 on BDI for unsupervised and semi-supervised VecMap for all combinations of Korean, Hebrew, and Thai. PanLex are results obtained with training lexicons from Vulić et al. (2019) and semi-supervised VecMap.

for Arabic, Chinese, Russian and Greek. In the unsupervised cases (Table 2), both ID and ID++ pair sets lead to an accuracy improvement of at least 17 points. ID++ outperform ID for three of the five low-resource pairs and five out of eight high-resource pairs proving that the romanized pairs can indeed strengthen the identical pairs sets. These results show that good quality BWEs can be built by relying on implicit cross-lingual signals without expensive supervision or fragile unsupervised approaches.

Non-English centric evaluation We analyze the performance of ID and ID++ for language pairs that do not include English. We use the test dictionaries of Vulić et al. (2019) that are derived from PanLex (Baldwin et al., 2010; Kamholz et al., 2014) by automatically translating each source language word into the target languages. We run VecMap for all combinations of Korean, Hebrew, and Thai. Romanized train lexicons are extracted by combining the languages through English (e.g., Th-Ko is obtained using En-Th and En-Ko), i.e., words are paired if their English translation is the same. Table 4 shows results. When Thai is involved, the unsupervised method fails as for English-Thai. Both ID and ID++ always outperform the respective unsupervised scores, and perform similar to higher-quality dictionaries. Additionally, ID++ outperforms ID in 3 out of 6 cases. These results demonstrate further the simplicity and high quality of our methods.

Romanized-only We analyze the performance of romanized pair lexicons on their own. Line Rom. in Table 2 and 3 shows that they obtain competitive results to the other two approaches, with improvements for Japanese, and perform similarly to MUSE dictionaries. The only failure is for for Chinese (En-Zh) – presumably because Chinese has a logographic script that does not represent phonemes directly, so romanization is less effective. These results show that the romanized pairs on their own also represent strong signals that shouldn't be neglected. Moreover, they constitute a good alternative when identical pairs are not available is such quantities (e.g., corpora of religious domain, law field, or cultural-specific documents).

Impact of OOVs We analyze the pairs used for the various sets (Appendix A, Table 5). We define OOVs as words for which there is no embedding available among the pre-trained Wikipedia fastText embeddings. Our romanized sets contain a substantial number of OOVs. (The identical pair sets do not contain OOVs because words are extracted from the top 200K most frequent.) The main reason for OOVs is that the selected English pair of a word is so rare that they do not have embeddings. On the other hand, the high number of OOVs (and resulting reduction of usable pairs) has only limited negative impact on the performance.

4 Conclusion

We presented two approaches to deal with the failure of unsupervised methods for some language pairs, focusing on English paired with non-Latin languages. (i) We exploited identical pairs that surprisingly appear in corpora of distinct scripts. (ii) We combined them with a simple method to extract the initial hypothesis set via romanization and edit distance. With both approaches, we obtained results that are competitive with high-quality dictionaries. Without using explicit cross-lingual signal, we outperformed previous unsupervised work for most languages and in particular for five language pairs for which previous unsupervised work failed. Our results question the strong trend towards unsupervised mapping approaches, and show that cheap cross-lingual signals should always be considered for building BWEs, even for distant languages.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

Jean Alaux, Edouard Grave, Marco Cuturi, and Armand Joulin. 2019. Unsupervised hyperalignment for multilingual word embeddings. In *Proceedings of the 7th International Conference on Learning Representations*.

David Alvarez-Melis and Tommi S Jaakkola. 2018. Gromov-wasserstein alignment of word embedding spaces. In *EMNLP*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798.

Author. 2020. Details withheld to preserve blind review.

Timothy Baldwin, Jonathan Pool, and Susan Colowick. 2010. Panlex and lextract: Translating all words of all languages of the world. In *Coling 2010: Demonstrations*, pages 37–40.

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.

Bharathi Raja Chakravarthi, Mihael Arcan, and John P McCrae. 2019. Comparison of different orthographies for machine translation of under-resourced dravidian languages. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Xilun Chen and Claire Cardie. 2018. Unsupervised multilingual word embeddings. *arXiv preprint arXiv:1808.08933*.

Zi Yi Dou, Zhi Hao Zhou, and Shujian Huang. 2020. Unsupervised bilingual lexicon induction via latent variable models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 621–626.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2019. Unsupervised alignment of embeddings with wasserstein procrustes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1880–1890. PMLR.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. Out-of-the-box universal romanization tool uroman. In *Proceedings of ACL 2018, System Demonstrations*, pages 13–18.

Yedid Hoshen and Lior Wolf. 2018. Non-adversarial unsupervised word translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 469–478.

David Kamholz, Jonathan Pool, and Susan Colowick. 2014. Panlex: Building a resource for panlingual lexical translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3145–3150.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics doklady*, 10(8):707–710.

Yanyang Li, Yingfeng Luo, Ye Lin, Quan Du, Huizhen Wang, Shujian Huang, Tong Xiao, and Jingbo Zhu. 2020. A simple and effective approach to robust unsupervised bilingual dictionary induction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5990–6001.

Kelly Marchisio, Kevin Duh, and Philipp Koehn. 2020. When does unsupervised machine translation work? *arXiv preprint arXiv:2004.05516*.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Muhammad Tasnim Mohiuddin, M Saiful Bari, and Shafiq Joty. 2020. Lnmap: Departures from isomorphic assumption in bilingual lexicon induction through non-linear mapping in latent space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2712–2723.

Muhammad Tasnim Mohiuddin and Shafiq Joty. 2019. Revisiting adversarial autoencoder for unsupervised word translation with cycle consistency and improved training. 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 3857–3867.

Parker Riley and Daniel Gildea. 2018. Orthographic features for bilingual lexicon induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 390–394.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Silvia Severini, Viktor Hangya, Alexander Fraser, and Hinrich Schütze. 2020. Combining word embeddings with bilingual orthography embeddings for bilingual dictionary induction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6044–6055.

Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*.

Ivan Vulić, Goran Glavaš, Roi Reichart, and Anna Korhonen. 2019. Do we really need fully unsupervised cross-lingual embeddings? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4398–4409.

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.

A Statistics

In this section we show statistics on the language pairs analyzed and additional scores. Table 5 presents the number of pairs for each set that are not OOVs in the fastText wiki embeddings (Bojanowski et al., 2017).

	MUSE	ID	Romanized	ID++
En-Th	6,799	46,653	10,721 / 53,804	58779 / 101066
En-Ja	7,135	43,556	11,488 / 118,626	54970 / 161848
En-Kn	1,552	21,090	12,888 / 59,207	33843 / 80032
En-Ta	8,091	23,538	5,987 / 120,836	29472 / 143990
En-Zh	8,728	62,289	6,360 / 41,829	68597 / 103971
En-Ar	11,571	19,275	4,773 / 61,031	24019 / 80115
En-Hi	8,704	30,502	16,180 / 73,553	46557 / 103791
En-Ru	10,887	18,663	9,913 / 301,698	28520 / 319688
En-El	10,662	35,270	20,740 / 150,472	55841 / 185244
En-Fa	8,869	32,866	10,226 / 85,210	43019 / 117817
En-He	9,634	17,012	4,005 / 40,258	20977 / 57059
En-Bn	8,467	31,954	10,721 / 53,804	42573 / 85532
En-Ko	7,999	15,518	9956 / 134156	25344 / 149031

Table 5: Number of pairs used that are not OOVs in the fastText wiki embeddings compared to the full size of the sets. For MUSE full and identical pairs sets there are no OOVs.

Size of seed set and word frequency We analyze the impact of the size of the initial romanized seed set and of word frequency. Table 6 displays accuracy scores for MUSE and Romanized lexicons containing $n \in \{25, 1000\}$ least and most frequent word pairs. Performance of VecMap applied to seed sets of size 25 is close to 0. The only exception is Russian where the unsupervised approach already works well. Next, we investigate seed sets of size 1000 consisting of either the least frequent or the most frequent words. High-frequency seed sets give better results as expected. The effect is particularly strong for Tamil: the high-frequency set has performance close to the full set whereas the low-frequency set is at $\leq$0.07. Performance of MUSE seed sets of size 25 and romanized seed sets of size 1000 is similar, demonstrating the higher quality of MUSE. However, obtaining the romanized pairs is much cheaper.

B Main results

In Table 7 there are the accuracy scores based on CSLS vector similarity calculated by the MUSE evaluation tool (Lample et al., 2018). We show the scores for thirteen language pairs in both directions. The first five pairs are the ones for which unsupervised methods fail. We show both unsupervised and semi-supervised VecMap performance with baselines dictionaries and our three sets.

C Reproducibility

We run our method on up to 48 cores of Intel(R) Xeon(R) CPU E7-8857 v2 with 1TB memory and a single GeForce GTX 1080 GPU with 8GB memory. The training of semi-suprised BWEs using VecMap took approximately 1 hour per language pair. For VecMap, as well as for all others methods we analyzed, we used the latest code available in their git repositories with default parameters. ID++ is implemented in Python.

		MUSE				Rom.			
		25L	25H	1000L	1000H	25L	25H	1000L	1000H
En-Ta	→	14.73	16.27	17.33	17.40	0.00	0.00	0.07	17.80
	←	16.48	18.35	22.44	23.44	0.00	0.00	0.00	21.57
En-Fa	→	35.33	34.20	38.07	37.20	0.00	0.20	37.47	37.47
	←	41.73	42.60	44.14	44.21	0.07	0.13	42.40	43.40
En-Zh	→	39.00	39.40	38.20	37.67	0.00	0.00	0.07	0.40
	←	32.93	34.47	34.33	34.40	0.00	0.00	0.07	0.60
En-Ru	→	49.07	43.07	49.07	49.27	49.33	47.73	49.40	49.00
	←	65.93	60.60	65.93	66.13	65.80	64.47	65.60	66.40

Table 6: acc@1 using 25 or 1000 pairs lower-frequency (L) and higher-frequency (H) sets for MUSE and our romanized only (Rom.) set.

			Baselines				Our		
			Unsupervised			Semi-sup.	Semi-supervised		
			1	2	3	MUSE	ID	Rom.	ID++
1	En-Th	→	0.00	0.00	0.00	**24.33**	**24.40**	23.33	23.47
		←	0.00	0.00	0.00	**19.04**	**19.92**	17.96	19.85
2	En-Ja	→	0.96	0.48	0.00	**48.73**	48.87	48.46	**49.14**
		←	0.96	0.00	0.00	**32.87**	33.22	**34.80**	33.43
3	En-Kn	→	0.00	0.00	0.00	**23.78***	22.03	22.90	**24.23**
		←	0.00	0.00	0.00	**41.25***	**43.04**	42.50	41.79
4	En-Ta	→	0.07	0.07	0.00$^{\diamond}$	**18.80**	17.93	18.00	**18.20**
		←	0.07	0.00	0.00$^{\diamond}$	**24.38**	**24.78**	23.51	**24.78**
5	En-Zh	→	0.07	0.00	0.00	**36.53**	**37.00**	0.27	35.00
		←	0.00	0.00	0.00	**32.80**	**34.33**	0.07	32.67
6	En-Ar	→	33.60	7.67	36.30$^{\diamond}$	**39.87**	**40.27**	39.33	40.20
		←	47.72	12.92	52.60$^{\diamond}$	**54.48**	54.42	54.42	**54.62**
7	En-Hi	→	40.20	0.00	0.00$^{\diamond}$	**40.33**	**40.47**	39.60	40.20
		←	**50.57**	0.07	0.00$^{\diamond}$	50.50	49.77	49.90	**50.10**
8	En-Ru	→	**48.80**	37.33	46.90$^{\diamond}$	**48.80**	49.13	48.87	**49.53**
		←	**66.13**	52.73	64.70$^{\diamond}$	65.67	**66.13**	65.73	66.07
9	En-El	→	47.67	34.67	47.90$^{\diamond}$	**48.00**	47.87	48.00	**48.27**
		←	63.40	49.20	**63.50**$^{\diamond}$	63.33	63.27	**64.40**	63.47
10	En-Fa	→	33.27	0.53	36.70$^{\diamond}$	**38.00**	**37.67**	36.80	**37.67**
		←	39.99	0.40	**44.50**$^{\diamond}$	43.47	**43.67**	42.93	43.60
11	En-He	→	44.60	37.13	44.00$^{\diamond}$	**45.00**	44.47	44.53	**44.67**
		←	57.88	50.01	57.10$^{\diamond}$	**57.94**	**58.14**	57.81	57.94
12	En-Bn	→	18.20	0.00	0.00$^{\diamond}$	**21.60**	19.87	19.80	**20.13**
		←	22.19	0.00	0.00$^{\diamond}$	**28.46**	28.88	28.67	**29.41**
13	En-Ko	→	19.80	9.62	0.00	**28.94**	27.92	28.40	**28.81**
		←	24.37	13.83	0.00	**34.09**	33.40	33.74	**33.95**

Table 7: acc@1 for unsupervised methods (1: Artetxe et al. (2018), 2: Grave et al. (2019), 3: Mohiuddin and Joty (2019)) and semi-supervised VecMap with different initial lexicons: MUSE set, identical pairs dataset (ID), our romanized only sets (Rom.), and the union of identical and romanized pairs (ID++). We show both forward (→) and backward (←) directions. In bold the best result for each pair of languages, for "Baselines" and "Our".
Scores from Mohiuddin et al. (2020) are marked with $^{\diamond}$.
*Kannada is not supported by MUSE, so we use the dictionary provided by (Author, 2020).