Classification-Based Self-Learning for Weakly Supervised Bilingual Lexicon Induction

Mladen Karan[♠*] **Ivan Vulić**[♢*] **Anna Korhonen**[♢] **Goran Glavaš**[♣]

[♠]TakeLab, Faculty of Electrical Engineering and Computing, University of Zagreb
[♢] Language Technology Lab, TAL, University of Cambridge
[♣] Data and Web Science Group, University of Mannheim
mladen.karan@fer.hr {iv250,alk23}@cam.ac.uk
goran@informatik.uni-mannheim.de

Abstract

Effective projection-based cross-lingual word embedding (CLWE) induction critically relies on the iterative *self-learning* procedure. It gradually expands the initial small seed dictionary to learn improved cross-lingual mappings. In this work, we present CLASSYMAP, a *classification-based approach to self-learning*, yielding a more robust and a more effective induction of projection-based CLWEs. Unlike prior self-learning methods, our approach allows for integration of diverse features into the iterative process. We show the benefits of CLASSYMAP for bilingual lexicon induction: we report consistent improvements in a weakly supervised setup (500 seed translation pairs) on a benchmark with 28 language pairs.

1 Introduction and Motivation

Cross-lingual word embeddings (CLWEs), that is, representations of words in a shared cross-lingual vector space, enable multilingual modeling of meaning and facilitate cross-lingual transfer for downstream NLP tasks (Ruder et al., 2019). One of their primary use cases is bilingual lexicon induction (BLI), that is, learning translation correspondences across languages which benefit the development of core language technology also for resource-poor languages and domains (Adams et al., 2017; Smith et al., 2017; Heyman et al., 2018; Hangya et al., 2018; Vulić et al., 2019).

Earlier work focused on joint CLWE induction from bilingual corpora, relying on word- (Klementiev et al., 2012; Gouws and Søgaard, 2015), sentence- (Zou et al., 2013; Hermann and Blunsom, 2014; Coulmance et al., 2015; Levy et al., 2017), or document-level supervision (Søgaard et al., 2015; Vulić and Moens, 2016). However, recent focus is predominantly on post-hoc alignment of independently trained monolingual word embeddings: the so-called *projection-based* or *mapping* approaches (Mikolov et al., 2013; Conneau et al., 2018; Joulin et al., 2018; Artetxe et al., 2018b; Patra et al., 2019). Such methods are particularly suitable for *weakly supervised* learning setups: they support CLWE induction with only as much as few thousand word translation pairs as the bilingual supervision.[1]

One critical component of weakly supervised projection-based CLWEs is a *self-learning* procedure that iteratively refines the initial seed dictionary to learn projections of increasingly higher quality. This process leads to substantial improvements of the initially mapped space, especially with smaller seed dictionaries (Artetxe et al., 2017; Vulić et al., 2019). However, current self-learning procedures are still rather basic, typically relying only on direct extraction of (mutual) nearest neighbors from the current shared space (Conneau et al., 2018; Artetxe et al., 2018b; Glavaš et al., 2019). In this work, we propose a more sophisticated self-learning procedure for weakly supervised projection-based CLWE methods, and show its benefits for a wide range of language pairs.

We frame self-learning as iterative *classification-based* process, which yields several benefits over the previously used self-learning mechanisms. **1)** It enables integration of a variety of heterogeneous features at different levels of granularity (e.g., word-level vs. orthographic features); some trans-

[*]Equal contribution.

[1]In the extreme, *fully unsupervised* projection-based CLWEs extract such seed bilingual lexicons from scratch on the basis of monolingual data only (Conneau et al., 2018; Artetxe et al., 2018b; Hoshen and Wolf, 2018; Alvarez-Melis and Jaakkola, 2018; Chen and Cardie, 2018; Mohiuddin and Joty, 2019, *inter alia*). However, as shown in recent comparative empirical analyses (Glavaš et al., 2019; Vulić et al., 2019), using seed sets of only 500-1,000 translation pairs, with all other components equal, always outperforms fully unsupervised methods. Therefore, we focus on a more natural weakly supervised setup (Artetxe et al., 2020) instead, i.e., we assume the existence of at least 500 seed translations for each language pair in consideration.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6915–6922
July 5 - 10, 2020.

lation cues (e.g., subword-level overlap) have been ignored by previous self-learning approaches. **2)** It allows us to control for the reliability of translation pairs considered as candidates for the dictionary updates in the current iteration. Effectively, this helps reduce noise in the process as the training dictionary grows. **3)** As suggested by prior work on classification-based BLI (Irvine and Callison-Burch, 2017; Heyman et al., 2017), framing the actual BLI task as a classification problem results in further gains in the final BLI performance.

We extensively evaluate our classification-based self-learning procedure, termed CLASSYMAP, on the standard BLI data set (Glavaš et al., 2019) spanning 28 pairs of diverse languages. The integration of the proposed self-learning method into VECMAP (Artetxe et al., 2018b), a state-of-the-art projection-based CLWE framework, yields substantial gains over previous self-learning procedures.[2] We demonstrate that the improvements are indeed achieved through the synergy of diverse features used by the classifier. We also demonstrate further BLI improvements when we treat BLI as a supervised classification-based task.

2 Classification-Based Self-Learning

Projection-Based CLWE Methods (linearly) align independently trained monolingual word embeddings $\boldsymbol{X}_1$ of the source language L_1 and $\boldsymbol{X}_2$ (target language L_2), using a seed word translation dictionary D (Mikolov et al., 2013; Artetxe et al., 2018a). Working in weakly supervised setups, we assume the existence of some translation pairs ($\approx$ 500 pairs) in D. Let $\boldsymbol{X}_{1,D} \subset \boldsymbol{X_1}$ and $\boldsymbol{X}_{2,D} \subset \boldsymbol{X_2}$ refer to the row-aligned subsets of monolingual embedding spaces containing vectors of translation pairs from D. Those are used to learn orthogonal transformations $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ that define the final shared cross-lingual space $\boldsymbol{W}_{cl} = \boldsymbol{W}_1 \cup \boldsymbol{W}_2$, where $\boldsymbol{W}_1 = \boldsymbol{X}_1\boldsymbol{T}_1$ and $\boldsymbol{W}_2 = \boldsymbol{X}_2\boldsymbol{T}_2$.

Our departure point is a standard self-learning setup from related work (Artetxe et al., 2018b; Conneau et al., 2018), outlined in the following. At each iteration k, the dictionary $D^{(k)}$ is first used to learn the joint space $\boldsymbol{W}_{cl}^{(k)} = \boldsymbol{W}_1^{(k)} \cup \boldsymbol{W}_2^{(k)}$.

[2]We use VECMAP due to its very competitive and robust BLI performance according to the recent comparative studies (Glavaš et al., 2019; Vulić et al., 2019; Doval et al., 2019). We note that our methodology is equally applicable to other projection-based methods that employ self-learning e.g., (Conneau et al., 2018; Mohiuddin and Joty, 2019), and our preliminary results with other methods suggest the similar benefits stemming from the classification-based approach.

Algorithm 1: Classification-based self-learning

$\mathbf{X}_1, \mathbf{X}_2 \leftarrow$ monolingual embeddings of L_1 and L_2
$D \leftarrow$ initial word translation dictionary
$C \leftarrow$ `TrainClassifier`(D)
$\mathbf{W}_1, \mathbf{W}_2 \leftarrow$ `AlignEmbeddings`$(\mathbf{X_1}, \mathbf{X_2}, D)$
for *each of* n *iterations* **do**
  $D_{1,2} \leftarrow$ nn$(\mathbf{W_1}, \mathbf{W_2})$; $D_{2,1} \leftarrow$ nn$(\mathbf{W_2}, \mathbf{W_1})$
  $D' \leftarrow (D_{1,2} \cap D_{2,1}) \setminus D$
  Sort D' descending by frequency
  $D'' \leftarrow$ first P elements of D'
  Generate scores for each pair in D'' using C
  Sort D'' descending by score
  Add first K elements of D'' to D
  $C \leftarrow$ `TrainClassifier`(D)
  $\mathbf{W}_1, \mathbf{W}_2 \leftarrow$
   `AlignEmbeddings`$(\mathbf{X_1}, \mathbf{X_2}, D)$
return: $\mathbf{W}_1$ (and/or $\mathbf{W}_2$) and C

The nearest neighbours in $\boldsymbol{W}_{cl}^{(k)}$ are then used to extract the new dictionary $D^{(k+1)}$. Previous work typically relies on a variant of mutual nearest neighbours in the aligned embedding space of the current iteration to select likely translation candidates for the next. However, as hinted by Lubin et al. (2019), that procedure still results in many noisy candidates inserted in the extended seed sets, and the error may get amplified over subsequent iterations.

New Self-Learning Procedure. Therefore, we propose a more versatile self-learning process. We train a *supervised classifier* in each iteration: given a word pair, it produces a probability score denoting to which extent the pair is a correct translation pair. The classifier can be fed a wide range of features on the character, subword, and word level.

We apply the classifier in two ways. First, at iteration k the classification scores are used to select likely translation candidates which are added to the dictionary $D^{(k+1)}$ for iteration $k+1$. Second, similar to Heyman et al. (2017), at test time we use the classifier scores to *rerank* translation candidates produced by 1) finding nearest neighbours in the final aligned embedding space and 2) considering orthographically similar candidates.[3] A high-level overview of the proposed classification-based self-learning procedure is outlined in Algorithm 1.

Self-Learning: Components. For implementing the `AlignEmbeddings` operation (see Algorithm 1) we rely on the VECMAP[4] system (Artetxe et al., 2018b) in its *supervised* variant. The *nn*

[3]We later show in §3 that both usages are beneficial for BLI. The former yields improved CLWEs directly. We plan to probe the usefulness of the CLWEs in other tasks beyond BLI in future work. The latter (reranking) step, on the other hand, is tied to the BLI task in particular. For this reason we later report all BLI results both with and without reranking.

[4]https://github.com/artetxem/vecmap

function returns word pairs that are nearest neighbours in a given aligned embedding space. The `TrainClassifier` functionality can be instantiated using any standard classification framework. In this work, we opt for a simple a multi-layer perceptron with a single hidden layer.

A very important design choice concerns generating negative training examples for the classifier. All word pairs in the dictionary at current iteration $D^{(k)}$ are used as positive examples. For each positive pair (s, t), we generate two negative examples: 1) (s, x), where x is sampled uniformly from N_o target words which are orthographically (measured by edit distance) most similar to s; 2) (s, y), where y is sampled uniformly from N_c target words closest (by cosine) to s in the current space $\boldsymbol{W}_{cl}^{(k)}$.

This strategy performed considerably better than randomly generating negative examples. The intuition is as follows: at test time the classifier must operate on word pairs that are generated using nearest neighbour search. Such word pairs are not random, but are rather very close in the aligned embedding space and are often orthographically similar. Thus, this strategy for generating negative samples makes the train conditions for the classifier better reflect the test conditions.

Features. The classification-based approach allows for the integration of a wide spectrum of diverse features that capture different word translation evidence. We outline the sets of features used in this work, computed for each word pair (s, t).

F1. Edit distance – Levenshtein and Jaro-Winkler distance between s and t (Cohen et al., 2003). Following Heyman et al. (2017) we also include normalized edit distance, log of the rank of t in a list sorted by edit distance with respect to s, as well as a product of these two values.

F2. Cosine similarity of s and t in $\boldsymbol{W}_{cl}^{(k)}$ (at iter k).

F3. Aligned embeddings of s and t, PCA-reduced to 10 dimensions (20 features in total).

F4. Normalized n-gram overlap (Šarić et al., 2012);

F5. Character n-grams – we extract all character n-grams and use χ^2 feature selection to select the 10 most indicative ones. The intuition is to allow the model to recognize indicative prefixes or suffixes.

F6. Subword-level similarity – we use multilingual subword embeddings (SWEs) based on BPEs (Heinzerling and Strube, 2018). We add the following features: i) we average the BPEs of s and t and calculate cosine similarity of the resulting vectors, ii) the pairwise maximum cosine similarity of all pairs of SWEs (one from s and the other from t), and iii) the Earth Mover's distance between the two sets of SWEs (Kusner et al., 2015).

F7. Frequencies – we provide the rank of the word in a list of all words sorted by frequency. The ranks are normalized by the number of words.

At test time, if we use the classifier to perform the final reranking, we take for each source word s a set of candidate target word translations as the union of 1) the top N_{ro} target word neighbours of s by edit distance, and 2) the top N_{rc} target word neighbours of s by cosine in the final aligned $\boldsymbol{W}_{cl}$. We then score the $N_{ro} + N_{rc}$ candidates using the classifier from the last self-learning iteration.

3 Experiments and Results

3.1 Experimental Setup

Monolingual Vectors and BLI Data. Following prior work (Artetxe et al., 2018b; Glavaš et al., 2019), we start from monolingual fastText vectors trained on full Wikipedias for each language (Bojanowski et al., 2017); vocabularies are trimmed to the 200K most frequent words. We evaluate on the standard BLI dataset from Glavaš et al. (2019): it comprises 28 language pairs with a good balance of typologically similar and distant languages: English (EN), German (DE), Italian (IT), French (FR), Russian (RU), Croatian (HR), Turkish (TR), and Finnish (FI). As our focus is on weakly supervised setups, we use only 500 translation pairs as our initial seed dictionary. We report BLI performance using the standard *Precision@1* (P@1) measure.

Classifier Details. We use the Adam optimizer (Kingma and Ba, 2015) and regularize the model via ℓ_2-penalty on the weights and early stopping on 10% of held-out data. Early stopping is performed for each language pair separately, while other hyperparameter values are found by grid search[5] maximizing a three-fold cross-validation score on the training data for a randomly selected language pair (EN–HR), and reused in all other experiments.

Hyperparameters. We find values for other hyperparameters on held-out data for a randomly chosen language pair: EN–HR. Unless otherwise stated, we fix them to the following values for all other experiments and language pairs. In Algo-

[5]Hidden layer sizes explored are 3, 5, 10, 20, 25 and regularization strengths are 0.0001, 0.01, and 1. The values selected by grid search were 25 and 1, respectively.

rithm 1, $P = 1000$, $K = 500$, $n = 30$. Further, we sample 2 negative examples per each positive example from the sets of size $N_o = N_c = 5$. $N_{ro} = N_{rc} = 3$ when doing the final reranking. We note that more careful tuning of these values could lead to further improvements in results.

Baselines. We compare to the VECMAP system (Artetxe et al., 2018b) in its *semi-supervised* variant as a robust and highly competitive self-learning framework (Glavaš et al., 2019; Vulić et al., 2019).

3.2 Results and Discussion

The main results over a representative selection of language pairs and setups are provided in Table 1. Full results over all 28 pairs are provided in Appendix A. The results indicate several important findings. First, classification-based self-learning is more powerful than the standard VECMAP self-learning: we observe gains on 22/28 pairs using CLASSYMAP without the final reranking step, even without language pair-dependent fine-tuning. Second, framing BLI as a classification task leads to further gains: we report improvements on 25/28 pairs using CLASSYMAP with the final reranking step over both supervised and semi-supervised VECMAP variants. Using reranking with CLASSYMAP seems useful across the board.[6]

As a side finding, our results also revalidate the evident usefulness of the self-learning procedure for weakly supervised setups in general (Vulić et al., 2019): the average P@1 score across *All* languages of a supervised VECMAP method based on the same initial dictionary, but without any self-learning, is only 0.111, while we report the average of 0.365 (with final reranking) in Table 1.

Importantly, the gains seem more pronounced for more "difficult", typologically dissimilar, and morphologically rich language pairs such as TR–RU or DE–TR, than for similar languages such as IT–FR, with more isomorphic monolingual spaces (Søgaard et al., 2018). To analyze this further, we have run additional experiments on the BLI evaluation sets of Vulić et al. (2019) comprising more typologically distant language pairs[7], with similar conclusions. For instance, with 500 seed pairs CLASSYMAP with reranking scores 24.6 P@1 for Estonian-Esperanto and 16.6 for Hungarian-Basque. The strongest baselines achieve P@1 of 20.0 and 13.8, respectively. In sum, our classification-based approach holds promise to guide future work especially on distant pairs.

	VECMAP (sup)	VECMAP	CLASSYMAP
TR-HR	.030	.160/.171	.200/**.227**
DE-TR	.050	.207/.203	.221/**.268**
TR-FI	.034	.200/.176	.217/**.235**
TR-RU	.028	.123/.152	.162/**203**
FI-HR	.049	.249/.195	.252/**.278**
DE-HR	.058	.229/.206	.246/**.268**
DE-RU	.111	.193/.208	.212/**.239**
EN-X	.177	.357/.325	.375/**.401**
No EN	.089	.310/.286	.322/**.353**
All	.111	.321/.296	.334/**.365**

Table 1: P@1 BLI scores for a selection of language pairs. We also perform the average scores over pairs that include English (EN-X) and those that do not (No EN), as well as the averages for all pairs (All). The a/b score format denotes a score without (a), and with the final reranking step (b). All improvements of CLASSYMAP with reranking over the strongest baseline (i.e., VECMAP with self-learning) are significant ($p<0.05$) according to the non-parametric shuffling test (Yeh, 2000) with the Bonferroni correction.

Step Size and the Number of Iterations. We now analyze how two vital components of self-learning impact the final BLI scores: 1) the number of added dictionary entries per iteration (i.e., step size, see Table 3), and 2) the number of iterations (Figure 1). For brevity, we run the analyses on several "difficult" language pairs: DE–RU, TR–FI, HR–FR, and EN–FI. The results suggest that the step size has only moderate impact on the final scores, and is language pair-dependent. However, all three options improve over the baseline self-learning method, and final reranking is again useful across the board. According to Figure 1, the optimal number of iterations is also pair-dependent: TR–FI performance steadily increases over time, while DE–RU hits the peak after only 5 iterations and steadily declines afterwards. This finding calls for a more careful tuning of this parameter in future work.

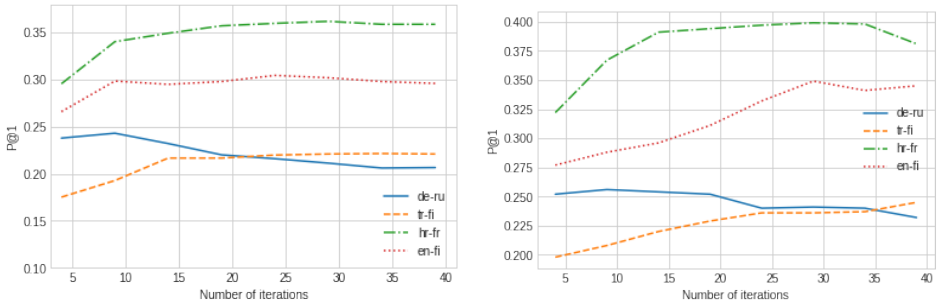
Feature Ablation Analysis. We also perform an ablation analysis, reported in Table 4. Overall, the results suggest that different features contribute to the final performance. This corroborates our hypothesis that one of the main advantages of the classification-based approach is its ability to fuse different translation evidence. However, there are cases (e.g., using *BPE* for DE–RU or TR–FI) where

[6] We have also probed a variant where we learn a classifier for the final reranking step on top of VECMAP's output after its self-learning procedure. However, as suggested by the results in Table 1, this leads to drops in performance compared to standard semi-supervised VECMAP. We speculate that this is due to higher levels of noise in the final VECMAP dictionary.

[7] github.com/cambridgeltl/panlex-bli

	500	1k	3k	5k
DE–RU	.111 / .193 / .212 / .239	.232 / .191 / .224 / .249	.301 / .194 / .244 / .277	.303 / .192 / .262 / .290
EN–FI	.081 / .238 / .299 / .350	.219 / .238 / .313 / .363	.320 / .238 / .318 / .362	.352 / .240 / .330 / .370
HR–FR	.053 / .352 / .363 / .411	.178 / .351 / .368 / .406	.325 / .352 / .376 / .420	.353 / .359 / .372 / .417
TR–FI	.034 / .200 / .217 / .235	.111 / .197 / .234 / .249	.213 / .197 / .246 / .266	.242 / .198 / .258 / .274

Table 2: Performance for varying initial dictionary sizes (500, 1k, 3k, 5k seed translation pairs). The numbers in each entry delimited with '/' are P@1 scores of 1) supervised VECMAP, 2) VECMAP with self-learning, 3) CLASSYMAP without reranking, and 4) CLASSYMAP with reranking, respectively.



(a) Without reranking (b) With reranking

Figure 1: BLI performance (P@1) of CLASSYMAP for varying numbers of self learning iterations.

Entries added	DE–RU	TR–FI	HR–FR	EN–FI
18x500	.219/.242	.215/.228	.362/.391	.298/.313
36x250	.220/.244	.217/.227	.363/.416	.295/.312
60x150	.220/.242	.220/.225	.352/.403	.314/.318

Table 3: P@1 BLI scores when varying the number of new dictionary entries added per iteration (i.e., *iterations* $\times$ *entries*). The a/b score format denotes a score without (a), and with the final reranking step (b).

Feature Sets	DE–RU	TR–FI	HR–FR	EN–FI
F1 + F7	.232/.182	.202/.147	.335/.223	.268/.172
+ F2	.231/.231	.194/.195	.350/.366	.280/.281
+ F3	.247/.260	.199/.204	.333/.365	.284/.292
+ F6	.244/.249	.191/.186	.348/.377	.280/.292
+ F4 + F5	.258/.255	.205/.211	.344/.376	.306/.301

Table 4: Feature ablation. $n = 10$. We experiment with Edit dist. (F1), frequencies (F7), cosine (F2), PCA (F3), BPE (F6), and n-grams (F4 + F5) .

a feature set can negatively affect performance. In sum, this small ablation study warrants finer-grained and language pair-dependent feature selection in future work.

Seed Dictionary Size. We also provide additional results when varying the size of the initial seed dictionary in Table 2. The main finding is that, while the absolute BLI scores are naturally higher with larger seed dictionaries, CLASSYMAP remains useful even with much larger dictionary sizes (check the results with 3k and 5k seed pairs). CLASSYMAP with reranking remains the strongest BLI method, corroborating our previous findings.

4 Conclusion and Future Work

We introduced CLASSYMAP, a novel classification-based approach to self-learning, which is a crucial component of projection-based cross-lingual word embedding induction models in low-data regimes. We reported its usefulness and robustness across a wide spectrum of diverse language pairs in the BLI task, confirming the usefulness of learning classifiers both as part of the self-learning procedure as well as for the final word retrieval in the BLI task.

This proof-of-concept work opens up a wide spectrum of interesting avenues for future research, including the use of more powerful classifiers, more sophisticated features (e.g., character-level transformers), and fine-grained linguistic analyses on the importance of disparate features over different language pairs. One particularly exciting direction is the application of our classification-based self-learning framework on top of the most recent methods that induce bilingual spaces via non-linear alignments (Glavaš and Vulić, 2020; Mohiuddin and Joty, 2020). The code is available online at: https://github.com/mladenk42/ClassyMap.

Acknowledgments

IV and AK are supported by the ERC Consolidator Grant LEXICAL (no 648909) awarded to AK. GG is supported by the Eliteprogramm of the Baden-Württemberg Stiftung (AGREE grant). We thank the reviewers for their insightful suggestions.

References

Oliver Adams, Adam Makarucha, Graham Neubig, Steven Bird, and Trevor Cohn. 2017. Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of EACL*, pages 937–947.

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-Wasserstein alignment of word embedding spaces. In *Proceedings of EMNLP*, pages 1881–1890.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of ACL*, pages 451–462.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Proceedings of AAAI*, pages 5012–5019.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of ACL*, pages 789–798.

Mikel Artetxe, Sebastian Ruder, Dani Yogatama, Gorka Labaka, and Eneko Agirre. 2020. A call for more rigor in unsupervised cross-lingual learning. In *Proceedings of ACL*.

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

Xilun Chen and Claire Cardie. 2018. Unsupervised multilingual word embeddings. In *Proceedings of EMNLP*, pages 261–270.

William W. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg. 2003. A comparison of string distance metrics for name-matching tasks. In *Proceedings of the International Conference on Information Integration on the Web*, pages 73–78.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *Proceedings of ICLR*.

Jocelyn Coulmance, Jean-Marc Marty, Guillaume Wenzek, and Amine Benhalloum. 2015. Trans-gram, fast cross-lingual word-embeddings. In *Proceedings of EMNLP*, pages 1109–1113.

Yerai Doval, Jose Camacho-Collados, Luis Espinosa-Anke, and Steven Schockaert. 2019. On the robustness of unsupervised and semi-supervised cross-lingual word embedding learning. *CoRR*, abs/1908.07742.

Goran Glavaš, Robert Litschko, Sebastian Ruder, and Ivan Vulić. 2019. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. In *Proceedings of ACL*, pages 710–721.

Goran Glavaš and Ivan Vulić. 2020. Non-linear instance-based cross-lingual mapping for non-isomorphic embedding spaces. In *Proceedings of ACL*.

Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *Proceedings of NAACL-HLT*, pages 1386–1390.

Viktor Hangya, Fabienne Braune, Alexander M. Fraser, and Hinrich Schütze. 2018. Two methods for domain adaptation of bilingual tasks: Delightfully simple and broadly applicable. In *Proceedings of ACL*, pages 810–820.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of LREC*, pages 2989–2993.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. In *Proceedings of ACL*, pages 58–68.

Geert Heyman, Ivan Vulić, and Marie-Francine Moens. 2017. Bilingual lexicon induction by learning to combine word-level and character-level representations. In *Proceedings of EACL*, pages 1085–1095.

Geert Heyman, Ivan Vulić, and Marie-Francine Moens. 2018. A deep learning approach to bilingual lexicon induction in the biomedical domain. *BMC Bioinformatics*, 19(1):259:1–259:15.

Yedid Hoshen and Lior Wolf. 2018. Non-adversarial unsupervised word translation. In *Proceedings of EMNLP*, pages 469–478.

Ann Irvine and Chris Callison-Burch. 2017. A comprehensive analysis of bilingual lexicon induction. *Computational Linguistics*, 43(2):273–310.

Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 2018. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of EMNLP*, pages 2979–2984.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *Proceedings of COLING*, pages 1459–1474.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *Proceedings of ICML*, pages 957–966.

Omer Levy, Anders Søgaard, and Yoav Goldberg. 2017. A strong baseline for learning cross-lingual word embeddings from sentence alignments. In *Proceedings of EACL*, pages 765–774.

Noa Lubin, Jacob Goldberger, and Yoav Goldberg. 2019. Aligning vector-spaces with noisy supervised lexicon. In *Proceedings of NAACL-HLT*, pages 460–465.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *CoRR, abs/1309.4168*.

Bari Saiful M Mohiuddin, Tasnim and Shafiq Joty. 2020. Lnmap: Departures from isomorphic assumption in bilingual lexicon induction through non-linear mapping in latent space. *CoRR, abs/1309.4168*.

Tasnim Mohiuddin and Shafiq Joty. 2019. Revisiting adversarial autoencoder for unsupervised word translation with cycle consistency and improved training. In *Proceedings of NAACL-HLT*, pages 3857–3867.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R. Gormley, and Graham Neubig. 2019. Bilingual lexicon induction with semi-supervision in non-isometric embedding spaces. In *Proceedings of ACL*, pages 184–193.

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

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity. In *Proceedings of STARSEM*, pages 441–448.

Samuel L. Smith, David H.P. Turban, Steven Hamblin, and Nils Y. Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In *Proceedings of ICLR*.

Anders Søgaard, Željko Agić, Héctor Martínez Alonso, Barbara Plank, Bernd Bohnet, and Anders Johannsen. 2015. Inverted indexing for cross-lingual NLP. In *Proceedings of ACL*, pages 1713–1722.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the limitations of unsupervised bilingual dictionary induction. In *Proceedings of ACL*, pages 778–788.

Ivan Vulić, Goran Glavaš, Roi Reichart, and Anna Korhonen. 2019. Do we really need fully unsupervised cross-lingual embeddings? In *Proceedings of EMNLP*, pages 4406–4417.

Ivan Vulić and Marie-Francine Moens. 2016. Bilingual distributed word representations from document-aligned comparable data. *Journal of Artificial Intelligence Research*, 55:953–994.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING*, pages 947–953.

Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of EMNLP*, pages 1393–1398.

A BLI Results for All 28 Language Pairs

	VECMAP (supervised)	VECMAP (SL)	VECMAP (SL+R)	CLASSYMAP (SL)	CLASSYMAP (SL+R)
EN–DE	.238	**.466**	.392	.451	.460
EN–TR	.076	.247	.253	.273	**.333**
EN–FI	.081	.238	.203	.299	**.350**
EN–HR	.072	.213	.189	.238	**.271**
EN–RU	.135	.203	.222	.230	**.266**
EN–IT	.325	**.552**	.480	.542	.546
EN–FR	.314	**.582**	.536	.573	.580
DE–TR	.050	.207	.203	.221	**.268**
DE–FI	.070	.240	.194	.265	**.297**
DE–HR	.058	.229	.206	.246	**.268**
DE–RU	.111	.193	.208	.212	**.239**
DE–IT	.196	.464	.397	**.475**	.466
DE–FR	.143	.465	.426	.461	**.484**
TR–FI	.034	.200	.176	.217	**.235**
TR–HR	.030	.160	.171	.200	**.227**
TR–RU	.028	.123	.152	.162	**.203**
TR–IT	.061	.296	.290	.297	**.334**
TR–FR	.047	.307	.323	.316	**.369**
FI–HR	.049	.249	.195	.252	**.278**
FI–RU	.064	.263	.217	.280	**.302**
FI–IT	.066	.318	.317	.328	**.376**
FI–FR	.059	.322	.315	.330	**.384**
HR–RU	.076	.305	.265	.312	**.347**
HR–IT	.078	.366	.332	.361	**.415**
HR–FR	.053	.352	.325	.363	**.411**
RU–IT	.130	.402	.343	.409	**.438**
RU–FR	.106	.407	.370	.417	**.442**
IT–FR	.367	.633	.583	.630	**.633**
EN–X	.177	.357	.325	.372	**.401**
No EN	.089	.310	.286	.322	**.353**
All	.111	.321	.296	.334	**.365**

Table 5: P@1 BLI scores for all 28 language pairs. We report scores of 1) VECMAP in the supervised setting without self learning, 2) VECMAP and CLASSYMAP with only self learning but without reranking (SL), and 3) VECMAP and CLASSYMAP with both self learning and reranking (SL+R). All models start with the same seed set of 500 word translation pairs.