Learning Unsupervised Multilingual Word Embeddings with Incremental Multilingual Hubs

Geert Heyman[†], **Bregt Verreet**[†], **Ivan Vulić**[‡], **Marie-Francine Moens**[†]
[†] LIIR, Department of Computer Science, KU Leuven
[‡] Language Technology Lab, TAL, University of Cambridge
{geert.heyman,bregt.verreet,sien.moens}@kuleuven.be
iv250@cam.ac.uk

Abstract

Recent research has discovered that a shared bilingual word embedding space can be induced by projecting monolingual word embedding spaces from two languages using a self-learning paradigm without any bilingual supervision. However, it has also been shown that for distant language pairs such fully unsupervised self-learning methods are unstable and often get stuck in poor local optima due to reduced isomorphism between starting monolingual spaces. In this work, we propose a new robust framework for learning unsupervised multilingual word embeddings that mitigates the instability issues. We learn a shared multilingual embedding space for a variable number of languages by incrementally adding new languages one by one to the current multilingual space. Through the gradual language addition our method can leverage the interdependencies between the new language and all other languages in the current multilingual hub/space. We find that it is beneficial to project more distant languages later in the iterative process. Our fully unsupervised multilingual embedding spaces yield results that are on par with the state-of-the-art methods in the bilingual lexicon induction (BLI) task, and simultaneously obtain state-of-the-art scores on two downstream tasks: multilingual document classification and multilingual dependency parsing, outperforming even supervised baselines. This finding also accentuates the need to establish evaluation protocols for cross-lingual word embeddings beyond the omnipresent intrinsic BLI task in future work.

1 Introduction

The ubiquitous use and success of word embeddings in monolingual tasks inspired further research on inducing cross-lingual word embeddings for two or more languages in the same vector space. Embeddings of translations and words with similar meaning are geometrically close in the *shared cross-lingual vector space*. This property makes them effective features for cross-lingual NLP tasks such as cross-lingual document classification (Klementiev et al., 2012), cross-lingual information retrieval (Vulić and Moens, 2015), bilingual lexicon induction (Mikolov et al., 2013b; Gouws et al., 2015; Heyman et al., 2017), and (unsupervised) machine translation (Artetxe et al., 2017b; Lample et al., 2018; Artetxe et al., 2018c).

Most prior work has focused on methods for constructing *bilingual* word embeddings (BWEs), yielding word representations for exactly two languages. For problems such as multilingual document classification, however, it is highly-desirable to represent words in a *multilingual* space. A favourable property is that it enables fitting a single classifier on the union of training datasets in many languages, which results in 1) knowledge transfer across languages that may lead to better classification performance, and 2) a setup that is easier to maintain as it is no longer required to train many different monolingual or bilingual classifiers.

Multilingual word embedding (MWE) methods typically generalize existing BWE methods by mapping multiple source language spaces to the space of one target language (Ammar et al., 2016), which is used as a pivot/hub language. This approach may lead to suboptimal solutions as it does not account for interdependencies between the source languages. Most BWE and MWE methods rely on *cross-lingual supervision* to some extent: e.g., bilingual lexicons (Mikolov et al., 2013a), parallel corpora (Gouws et al., 2015), or subject-aligned document pairs (Vulić and Moens, 2016). In such paradigms, modeling dependencies between all languages is impractical as it requires supervision for all language pair combinations.

Recent research has shown that BWEs can also be learned *without* cross-lingual supervision and can even outperform supervised BWE variants

on bilingual lexicon induction benchmarks (Conneau et al., 2018; Artetxe et al., 2018a). Chen and Cardie (2018) took a first step towards learning multilingual spaces without supervision while incorporating dependencies between all languages but their approach extends the work of Conneau et al. (2018), which has known limitations concerning optimization stability with distant language pairs (Søgaard et al., 2018). In this work, we investigate robust methods to induce MWEs without any cross-lingual supervision. The robustness of our approach is illustrated in good performance for distant languages such as Finnish and Bulgarian. This paper makes the following contributions.

First, based on a reformulation of the BWE method of Artetxe et al. (2018a), we propose two novel methods for inducing MWEs: 1) the single hub space model (SHS) uses the classical idea of mapping source languages to a single hub language; 2) the incremental hub space model (IHS) incorporates dependencies between all languages by incrementally expanding the multilingual space by one language in each step. IHS results in mappings that are more robust and coherent across languages. Both SHS and IHS only require monolingual data.

Second, we evaluate our method on benchmarks for bilingual lexicon induction (BLI), multilingual document classification, and dependency parsing. We find that the IHS method is competitive with state-of-the-art BWE methods on the bilingual lexicon induction benchmarks, while yielding the highest scores on the multilingual document classification and dependency parsing benchmarks.

Third, unlike the majority of prior work (Conneau et al., 2018; Artetxe et al., 2018a; Chen and Cardie, 2018, *inter alia*), we do not limit our evaluation to the intrinsic BLI task only. Consequently, we investigate if embedding reweighting, a recently proposed best practice for BWEs, is useful for extrinsic tasks such as document classification and dependency parsing in multilingual settings.

2 Related Work

Cross-lingual word embeddings have received a lot of attention in recent years. Most methods construct a space shared between two languages using cross-lingual supervision in the form of bilingual lexicons (Mikolov et al., 2013a; Artetxe et al., 2016; Smith et al., 2017), parallel corpora (Klementiev et al., 2012; Faruqui and Dyer, 2014; Gouws et al., 2015; Luong et al., 2015) or subject-aligned document pairs (Vulić and Moens, 2016). See Ruder et al. (2018) for a full overview of BWE model typology in relation to the required supervision.

To enable knowledge transfer across an arbitrary number of languages, multilingual methods have been introduced. Huang et al. (2015), propose decomposing a matrix with multilingual co-occurrence counts weighted by probabilistic dictionaries. Ammar et al. (2016) compare this method to three other MWE models: MultiCluster, MultiCCA, and MultiSkip. MultiCluster uses bilingual dictionaries to cluster translations and then train the monolingual Skip-gram model (SG) (Mikolov et al., 2013a) on a union of monolingual corpora where they replace words with their cluster id such that words in the same cluster get the same representation. MultiCCA is the multilingual extension of the method of Faruqui and Dyer (2014): Using canonical correlation analysis (CCA) and dictionaries with English as the target language, monolingual embeddings are projected to the English vector space. MultiSkip is a straightforward extension of the BiSkip method (Luong et al., 2015) which generalizes the monolingual SG objective to account for word alignments in parallel corpora. Similarly, Duong et al. (2017), extend CBOW to multiple languages. All these methods learn multilingual embeddings using bilingual dictionaries of parallel corpora: This limits their applicability for many languages.

More recently, Conneau et al. (2018); Artetxe et al. (2018a) showed that BWEs can be effectively induced without any cross-lingual supervision. The approaches are based on the assumption that monolingual embedding spaces are approximately isomorphic.[1] Improving on earlier attempts (Cao et al., 2016; Zhang et al., 2017), Conneau et al. (2018) propose a two-step framework to map two monolingual spaces to the shared space. First, they use an adversarial objective to get an initial bilingual space in which the discriminator can no longer distinguish to which language a given word embedding belongs. They then fine-tune the initial solution. An important limitation is that the adversarial objective is prone to converge to degenerate solutions. Furthermore, Søgaard et al. (2018) empirically prove that the method typically fails for distant language pairs such as English-Finnish.

In parallel, Artetxe et al. (2018a) proposed an-

[1] One of the necessary conditions for this assumption to hold is that the monolingual corpora on which the embeddings are trained are comparable (Søgaard et al., 2018).

other framework with the same goal. Expanding on their earlier work (Artetxe et al., 2017a, 2018b), they use an unsupervised heuristic to obtain a noisy initial seed lexicon which is used to obtain an initial bilingual space. This solution is iteratively improved similar to Artetxe et al. (2017a) and Conneau et al. (2018) while using value dropping regularization to escape early convergence to local minima. Their method is the starting point for this work, and it is discussed in detail in §3.

The two unsupervised approaches are limited to finding mappings between a pair of languages. To the best of our knowledge, Chen and Cardie (2018) is the only unsupervised method that constructs a multilingual embedding space. Their method extends the adversarial pre-training and iterative refinement steps of Conneau et al. (2018) to the multilingual setting. Their work does not investigate the limitations of Conneau et al. (2018)'s method: Less stable optimization and difficulties with mapping spaces with reduced isomorphism. Furthermore, their generalization turns the iterative refinement into a non-convex optimization problem. In contrast, our multilingual methods proposed in this work are applicable to distant language pairs and decompose every iteration in the refinement step in multiple convex optimization problems, making them very robust and widely applicable.

3 Unsupervised Bilingual Embeddings

We now summarize mapping-based approaches to learning BWEs, which serve as the backbone of our multilingual approach. These methods rely on a *mapping procedure*, that is, a way to transform two monolingual spaces such that translations and similar words obtain similar representations. Supervised approaches take translations from readily available seed training dictionaries. Unsupervised approaches construct a seed lexicon from scratch, and use an iterative procedure to refine the seed lexicon and the mapped bilingual space.

Mapping Procedure. Various mapping procedures have been proposed in the literature (Mikolov et al., 2013b; Dinu et al., 2015; Lazaridou et al., 2015; Vulić and Korhonen, 2016). These methods can be seen as variants of a a single framework (Artetxe et al., 2018b), summarized here.

At its core, each mapping procedure learns the orthogonal transformations $\boldsymbol{W}_x$ and $\boldsymbol{W}_z$ for the monolingual embedding spaces $\boldsymbol{X}$ and $\boldsymbol{Z}$ that minimize the distance between embeddings of translations in the mapped spaces $\boldsymbol{X}\boldsymbol{W}_x$ and $\boldsymbol{Z}\boldsymbol{W}_z$. The orthogonality constraint ensures that the transformations preserve the monolingual constellation of embeddings. Formally, let $\boldsymbol{D}$ be a matrix representing a bilingual dictionary s.t. $D_{ij} = 1$ if the i^{th} source word is translated by the j^{th} target word and $D_{ij} = 0$ otherwise, then $\boldsymbol{W}_x$ and $\boldsymbol{W}_z$ are found by solving the following optimization problem:

$$\begin{aligned}
&\arg\max_{\boldsymbol{W}_x,\boldsymbol{W}_z} \sum_i \sum_j D_{ij}(\boldsymbol{X}_{i,:}\boldsymbol{W}_x) \cdot (\boldsymbol{Z}_{j,:}\boldsymbol{W}_z) \\
&= \arg\max_{\boldsymbol{W}_x,\boldsymbol{W}_z} tr(\boldsymbol{X}\boldsymbol{W}_x(\boldsymbol{D}\boldsymbol{Z}\boldsymbol{W}_z)^\intercal) \qquad (1) \\
&\text{subject to } \boldsymbol{W}_x\boldsymbol{W}_x^\intercal = \boldsymbol{I}\ ,\ \boldsymbol{W}_z\boldsymbol{W}_z^\intercal = \boldsymbol{I} \\
&\text{where } tr(\cdot) \text{ denotes the trace operator.}
\end{aligned}$$

Eq. (1) has a closed-form solution based on the singular vectors of $\boldsymbol{X}^\intercal\boldsymbol{D}\boldsymbol{Z}$: $\boldsymbol{W}_x = \boldsymbol{U}, \boldsymbol{W}_z = \boldsymbol{V}$ with $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\intercal = \text{SVD}(\boldsymbol{X}^\intercal\boldsymbol{D}\boldsymbol{Z})$. In addition to the transformation, there are several optional pre-processing (S1-S2) and post-processing (S3-S5) steps:

S1. *Normalization*: apply length normalization (normalizing $\boldsymbol{X}$ and $\boldsymbol{Z}$ such that all embeddings have a unit Euclidean norm), or mean centering, or a combination of both;

S2. *Whitening*: apply ZCA whitening (Bell and Sejnowski, 1997) on $\boldsymbol{X}$ and $\boldsymbol{Z}$ which transforms the monolingual embedding matrices such that each dimension/component has unit variance and such that the dimensions are uncorrelated (see Eq. (2) later). The intuition is that it is easier to align the vector spaces along directions of high variance;

S3. *Re-weighting* the components according to the singular value matrix $\boldsymbol{S}$ of $\boldsymbol{X}^\intercal\boldsymbol{D}\boldsymbol{Z}$: This is an attempt to further align the embeddings in the multilingual space as each singular value measures how well a dimension in the multilingual space correlates across languages for the given dictionary;

S4. *De-whitening*, the inverse transformation of S2: After the mapping, it was shown as important to restore the variance information in case whitening was applied (Artetxe et al., 2018c);

S5. *Dimensionality reduction* truncates the embedding vectors such that only components with the highest singular values are kept.

Refinement Procedure. The refinement aims at iteratively improving the seed dictionary and the bilingual space with Expectation Maximization (Dempster et al., 1977). In each iteration, the mapping procedure is executed using the dictionary

from the previous iteration to obtain a new bilingual space, and then a new dictionary is induced using nearest neighbor retrieval in the *cross-lingual similarity matrix* $\boldsymbol{M}$. This is repeated until the (unsupervised) training objective $\sum_i \sum_j D_{ij}\big((\boldsymbol{X}_{i,:}\boldsymbol{W}_x) \cdot (\boldsymbol{Z}_{j,:}\boldsymbol{W}_z)\big)$ stops increasing.

The matrix $\boldsymbol{M}$ is calculated using cross-domain similarity local scaling (CSLS; Conneau et al. (2018)), an extended variant of cosine similarity that avoids the hubness problem (Radovanović et al., 2010; Dinu et al., 2015).[2] In particular, the element m_{ij} at row i and column j of $\boldsymbol{M}$ corresponds to the CSLS value between the cross-lingual vectors $\boldsymbol{x}_i^{CL}$ and $\boldsymbol{z}_j^{CL}$ of the i^{th} source word and the j^{th} target word respectively: $m_{ij} = \text{CSLS}(\boldsymbol{x}_i^{CL}, \boldsymbol{z}_j^{CL})$; $\text{CSLS}(\boldsymbol{x}, \boldsymbol{z}) = 2cos(\boldsymbol{x}, \boldsymbol{z}) - r_{Zk}(\boldsymbol{x}) - r_{Xk}(\boldsymbol{z})$. $r_{Xk}(\boldsymbol{x})$ and $r_{Zk}(\boldsymbol{z})$ calculate the average cosine similarity of a vector with its k nearest neighbors (measured by cosine) in the mapped spaces of $\boldsymbol{X}$, $\boldsymbol{Z}$, respectively. It is also beneficial to jointly infer dictionaries source-to-target and target-to-source (Artetxe et al., 2018a).[3] The mapping is then learned from the concatenation of these two dictionaries.[4]

To avoid suboptimal local minima, Artetxe et al. (2018a) propose to randomly drop values from the matrix $\boldsymbol{M}$ with probability $1 - p$ (further *value dropping*). The value of p is exponentially increased as training progresses. p is initialized to a small value (e.g., 0.1). Whenever the objective stops improving for $N_{patience}$ refinement steps, p is multiplied with a given factor (e.g., 2) until $p \geq 1$ after which all values in $\boldsymbol{M}$ are kept. Value dropping was shown to be crucial when constructing bilingual spaces between distant language pairs. We later analyze its impact on the proposed multilingual methods.

Inducing a Seed Lexicon. Artetxe et al. (2018a) obtain a seed lexicon based on the assumption that for a translation pair w_i^X, w_j^Z, the monolingual similarity vectors, $\sqrt{\boldsymbol{X}_i\boldsymbol{X}^\intercal}$ and $\sqrt{\boldsymbol{Z}_j\boldsymbol{Z}^\intercal}$ of translations i, j are (approximately) equal up to a permutation.[5] Therefore, seed translations for a source word i are generated by finding the nearest neighbor based on similarity of monolingual similarities. This heuristic yields a very noisy seed lexicon, but it was proven to contain a sufficiently strong bilingual signal to bootstrap the refinement procedure. The seed lexicon is inferred symmetrically (i.e., by concatenating respective source-to-target and target-to-source seed lexicons) and the vocabularies are truncated to the C_{seed} most frequent words.

4 Unsupervised Multilingual Embeddings: Methodology

We now present two models for learning unsupervised multilingual word embedding spaces: the single hub space model (SHS) and the incremental hub space model (IHS). The methods generalize the bilingual framework described in §3, and rely on (a subset of) preprocessing and postprocessing steps S1-S5 in the multilingual setting.

Single Hub Space (SHS). The SHS model defines one language as the hub language L_0 and projects the embedding spaces $\boldsymbol{Z}_1, ..., \boldsymbol{Z}_N$ of all other languages $L_1, ..., L_N$ (further *secondary languages*) to the hub space $\boldsymbol{X}$. Hence, we reduce the construction of a multilingual space of N languages to the alignment of $N - 1$ vector spaces. Learning these projections is similar to the bilingual case: We use the unsupervised iterative refinement procedure and seed lexicon heuristic from §3. However, we require the orthogonal mapping to be asymmetric: The hub language space should either remain unchanged or it should be transformed with the same operation for each of the $N - 1$ language pairs. We therefore derive an asymmetric version of the mapping framework from §3 that yields the exact same solution as the original.

Let $\boldsymbol{X}$ be the embedding matrix of the, $\boldsymbol{Z}_1$,. . ., $\boldsymbol{Z}_N$ the embedding matrices of the secondary languages, and $\boldsymbol{D}^{k,l}$ the dictionary between languages L_k and L_l. We induce a multilingual space $\boldsymbol{X}^m$, $\boldsymbol{Z}_1^m$, ... , $\boldsymbol{Z}_N^m$ in three main steps. First, the embeddings of each language are preprocessed by normalizing and whitening the embeddings, as described by Eqs. (2)-(6). Normalization consists of subsequently performing length normalization, mean

[2] Hubness is the phenomenon observed in a high-dimensional vector space where there are vectors, called hubs, which are the nearest neighbors to many vectors in the space.

[3] Note that $\boldsymbol{z}_j^{CL}$ being the nearest neighbor of $\boldsymbol{x}_i^{CL}$ does not imply that the inverse is also true.

[4] Due to the large search space, limiting the search space by truncating both vocabularies and their corresponding embedding matrices to the $C_{refinement}$ most frequent words results in better solutions and speeds up computation.

[5] The square root in the formulas is empirically motivated.

centering, and then again length normalization.

$$\text{ZCAwhiten}(\boldsymbol{W}) = \boldsymbol{W}(\boldsymbol{W}^\intercal\boldsymbol{W})^{-0.5} \quad (2)$$

$$\boldsymbol{X}^{'} = \text{normalize}(\boldsymbol{X}) \quad (3)$$

$$\boldsymbol{X}^{''} = \text{ZCAwhiten}(\boldsymbol{X}^{'}) \quad (4)$$

$$\boldsymbol{Z}_l^{'} = \text{normalize}(\boldsymbol{Z}_l) \quad (5)$$

$$\boldsymbol{Z}_l^{''} = \text{ZCAwhiten}(\boldsymbol{Z}_l^{'}) \quad (6)$$

After preprocessing, we rotate each secondary language L_l to a bilingual space between the hub language space and its own embedding space, as described by Eqs. (7)-(10). The calculations are analogous to the bilingual mapping procedure: the left and right singular vectors $\boldsymbol{U}_l$ and $\boldsymbol{V}_l$ of $\boldsymbol{X}^{''}\boldsymbol{D}^{k,l}\boldsymbol{Z}_l^{''\intercal}$ are the rotation matrices that project the preprocessed matrices $\boldsymbol{X}^{''}$ and $\boldsymbol{Z}_l^{''}$ to their bilingual space; this is formulated in Eqs. (7)-(8). The bilingual projection of $\boldsymbol{Z}_l^{''}$ can be reweighted by multiplying it with a given power q of the singular values matrix $\boldsymbol{S}_l$ of $\boldsymbol{X}^{''\intercal}\boldsymbol{D}^{0,l}\boldsymbol{Z}_l^{''}$, Eq. (9). Intuitively, the reweighting operation makes the dimensions that correlate better across languages more important. Next, we restore the variance information of $\boldsymbol{Z}_l^{'}$ by performing a dewhitening operation: we project back to the monolingual space, multiply with the inverse of the whitening matrix, and then project back to the bilingual space (Eq. (10)).[6]

$$\boldsymbol{U}_l\boldsymbol{S}_l\boldsymbol{V}_l^\intercal = \text{SVD}(\boldsymbol{X}^{''\intercal}\boldsymbol{D}^{0,l}\boldsymbol{Z}_l^{''}) \quad (7)$$

$$\boldsymbol{Z}_{l,bi(l)} = \boldsymbol{Z}_l^{''}\boldsymbol{V}_l \quad (8)$$

$$\boldsymbol{Z}_{l,bi(l)}^{'} = \boldsymbol{Z}_{l,bi(l)}\boldsymbol{S}_l^q \quad (9)$$

$$\boldsymbol{Z}_{l,bi(l)}^{''} = \boldsymbol{Z}_{l,bi(l)}^{'}\boldsymbol{V}_l^\intercal(\boldsymbol{Z}_l^{'\intercal}\boldsymbol{Z}_l^{'})^{0.5}\boldsymbol{V}_l \quad (10)$$

Finally, we project $\boldsymbol{Z}_{l,bi(l)}^{''}$ to the space of the hub language in Eq. (11)). The multilingual space for the hub language is simply the monolingual embedding space after preprocessing, see Eq. (12).

$$\boldsymbol{Z}_l^m = \boldsymbol{Z}_{l,bi(l)}^{''}\boldsymbol{U}_l^\intercal \quad (11)$$

$$\boldsymbol{X}^m = \boldsymbol{X}^{'} \quad (12)$$

For the bilingual case this formulation is equivalent to the symmetric mapping introduced in §3: one can easily verify that the dot products between the mapped spaces simplify to the same formula.

[6]Note that the projection matrices that map from the bilingual to the monolingual spaces are given by the inverses of $\boldsymbol{U}_l$ and $\boldsymbol{V}_l$. Since the matrices are orthogonal their inverses are equal to their transposes.

Incremental Hub Space (IHS). SHS enables language interactions only indirectly through the hub language. Ideally, a multilingual method should incorporate interdependencies between all languages. We hypothesize that, especially when mapping a language distant to the hub language, it is beneficial to incorporate the structural similarities with all other languages as a regularization mechanism to find a more robust mapping.

We therefore propose the incremental hub space (IHS) model. It incrementally expands the multilingual space $\boldsymbol{X}^m$ and takes into account all languages in the current multilingual space when adding a new language. First, we define a language order and initialize the space to the preprocessed embedding space of language L_0. Next, following the order, we gradually add new languages to the space: in each iteration we rotate the preprocessed embedding space $\boldsymbol{Z}_l^{''}$ of language l to the multilingual space by minimizing the dot product between embeddings of the translations between language l and all the languages in the multilingual space. The recipe to calculate the cross-lingual embedding $\boldsymbol{Z}_l^m$ is similar to the SHS model: the preprocessing and postprocessing steps are the same, but the projection matrices are calculated with Eq. (14) instead of Eq. (7) and conform with the new objective from Eq. (13). After convergence, $\boldsymbol{Z}_l^m$ is added to the multilingual space $\boldsymbol{X}^m$: $\boldsymbol{X}_l^m = \boldsymbol{Z}_l^m$.

$$\underset{\boldsymbol{W}_{xl},\boldsymbol{W}_{zl}}{\arg\max} \sum_{k=0}^{l-1} tr(\boldsymbol{X}_k^m\boldsymbol{W}_{xl}(\boldsymbol{D}^{k,l}\boldsymbol{Z}_l^{''}\boldsymbol{W}_{zl})^\intercal) \quad (13)$$

$$\text{subject to } \boldsymbol{W}_{xl}\boldsymbol{W}_{xl}^\intercal = \boldsymbol{I}, \boldsymbol{W}_{zl}\boldsymbol{W}_{zl}^\intercal = \boldsymbol{I}$$

$$\boldsymbol{U}_l\boldsymbol{S}_l\boldsymbol{V}_l^\intercal = \text{SVD}(\boldsymbol{C}) \quad (14)$$

$$\boldsymbol{C} = (\boldsymbol{X}_0^m)^\intercal\boldsymbol{D}^{0,l}\boldsymbol{Z}_l^{''}|| \ldots ||(\boldsymbol{X}_{l-1}^m)^\intercal\boldsymbol{D}^{l-1,l}\boldsymbol{Z}_l^{''}$$

where $||$ denotes concatenation along the row axis

In supervised settings this approach would be impractical as it requires bilingual dictionaries $\boldsymbol{D}^{k,l}$ for all language pairs k, l, and not only with the hub language. However, within an unsupervised framework this constraint is lifted.

5 Experimental Setup

Tasks and Datasets. The induced embeddings are evaluated in three tasks: bilingual lexicon induction (BLI), multilingual dependency parsing, and multilingual document classification. BLI is currently the most widely used method to evaluate bilingual embedding spaces. Although BLI performance is not the primary goal of our multilingual

embedding spaces, it provides a fast means to address the following questions: **1)** Is the incremental construction of multilingual embedding spaces indeed an effective regularization method? Is it still necessary to perform value dropping in this case?[7]; **2)** Is the reweighting of embedding spaces also beneficial for BLI in multilingual settings?; **3)** Does multilingual training improve bilingual lexicon induction performance? How do our multilingual models compare to each other and to the state-of-the-art unsupervised BLI methods?

We report *Precision@1 (P@1)* BLI performance on two standard BLI datasets. 1) DINUARTETXE is the extended version of Dinu et al. (2015)'s dataset, used by Artetxe et al. (2018a).[8] It consists of bilingual dictionaries for English-German, English-Italian, English-Spanish and English-Finnish. Monolingual embeddings are provided, based on the CBOW model trained on the WaCKy corpora for English, Italian and German (Baroni et al., 2009), the monolingual WMT Common Crawl corpus for Finnish, and the WMT News Crawl for Spanish (Bojar et al., 2015). The test dictionary sizes are between 1.869 and 1,993 word pairs for each language pair. As our methods are unsupervised, we do not use the provided training dictionaries. 2) EURMUSEWIKI is the dataset compiled from dictionaries for all combinations of the following European languages: English, German, Spanish, French, Italian, and Portuguese. The test set sizes range between 1,513 and 3,660 word pairs. We rely on publicly available monolingual fastText embeddings (Bojanowski et al., 2016).[9] All monolingual word embeddings are 300-dimensional and represent the 200k most frequent words as in prior work (Dinu et al., 2015; Conneau et al., 2018).

Multilingual dependency parsing and multilingual document classification tasks assess the embeddings w.r.t. their actual goal: enabling transfer learning across multiple languages. The word embeddings are used as feature vectors for classifiers in the respective downstream tasks. We address the following research questions: **4)** Is reweighting of embedding spaces also beneficial in downstream tasks?; **5)** How do our methods compare against supervised multilingual embedding models?

We rely on the evaluation platform of Ammar et al. (2016) for the downstream tasks[10]: the users submit their multilingual embeddings and obtain the final scores, which ensures that the classifiers we use are identical to the ones used in prior work (Ammar et al., 2016; Duong et al., 2017).

REUTERSMLDC is a multilingual document classification dataset covering seven languages: English, German, French, Italian, Spanish, Danish, and Swedish. The final performance is reported as the average accuracy across all languages. The respective training and test set consist of 7,000 and 13,058 documents. The dataset is well balanced in the number of documents per language.[11] The architecture of the document classifier is the average perceptron used by Klementiev et al. (2012).

MLPARSING is a multilingual dependency parsing dataset sampled from the Universal Dependencies 1.1 corpus (Agić et al., 2015)[12]. It contains 12 languages: English, German, French, Spanish, Italian, Bulgarian, Czech, Danish, Swedish, Greek, Finnish, and Hungarian. The respective training and test set contain 6,748 and 1,200 sentences. The test set contains 100 sentences for each language, while for the training set the number of sentences for a language ranges between 98 and 6,694. The parser used is the stack-LSTM parser by Dyer et al. (2015). The parser is not allowed to use any part-of-speech and morphology features, and keeps the input word embeddings fixed to isolate the effect of the evaluated embeddings on the parsing performance (Ammar et al., 2016). The reported scores are UAS scores averaged across languages.

For comparison with related work, we train 512-dimensional monolingual embeddings on the text collections used by Ammar et al. (2016) and Duong et al. (2017). The monolingual embeddings are again trained using fastText.

Training Setup. In all experiments, we set the following hyper-parameters to values that were used in prior research (Conneau et al., 2018; Artetxe et al., 2018a). When constructing the seed lexicon the 4,000 most frequent words of each language are considered ($C_{seed} = 4,000$), and during the refinement step the 20,000 most frequent words

[7] Value dropping significantly slows down training time and leads to non-deterministic outcomes. However, it has been shown to be crucial in the bilingual setting to obtain good results when mapping distant language pairs in previous work (Artetxe et al., 2018a).

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

[9] https://fasttext.cc/docs/en/pretrained-vectors.html

[10] https://github.com/wammar/multilingual-embeddings-eval-portal

[11] As the dataset is not publicly available this information was provided by the first author of Ammar et al. (2016).

[12] http://hdl.handle.net/11234/LRT-1478

Model	Drop	EN-DE	EN-IT	EN-ES	EN-FI	Avg
SHS	no	48.00	45.93	36.53	0.14	32.65
SHS	yes	47.51	45.60	36.33	31.92	**40.34**
IHS	no	47.93	45.93	36.07	31.04	**40.24**
IHS	yes	47.45	45.48	36.43	30.97	40.08

Table 1: Comparison of *P@1* scores for SHS and IHS models with and without value dropping (*Drop*) on the DINUARTETXE BLI dataset.

of each language are used ($C_{refinement} = 20,000$). When using value dropping, the keep probability p is initialized is 0.1, $N_{patience}$ is set to 50, and the stochastic multiplier is set to 2. Dictionaries are constructed symmetrically: from hub language(s) to the secondary language and from the secondary language to the hub language(s): during refinement each dictionary consists of $2 \times 20,000$ translation pairs. We use CSLS with $k = 10$ nearest neighbors following the setup of Conneau et al. (2018).

6 Results and Discussion

Experiment 1: Value Dropping. In the first experiment, we investigate if the expensive value dropping procedure is a necessary condition for mapping between distant language pairs in our multilingual framework. Table 1 provides results on the DINUARTETXE dataset for SHS and IHS models. For IHS we process the languages in the following order: English, German, Italian, Spanish, Finnish. When using value dropping we report the average and best results across five runs.

We observe that value dropping is crucial for SHS to succeed for English-Finnish. However, it is not necessary for IHS. This supports our hypothesis that mapping a language to a multilingual hub space serves a type of regularization that can substitute value dropping. As validated later, it is still important to avoid adding distant languages early with the incremental IHS procedure.[13,14]

Experiment 2: Comparative BLI Performance and Reweighting. In this experiment, we test if reweighting embedding spaces is beneficial for BLI in our multilingual setup, and also compare our methods against state-of-the-art BLI methods. Table 2 and Table 3 show the results for SHS and IHS with reweighting coefficients q of 0, 0.5 and 1 on DINUARTETXE and EURMUSEWIKI, respectively. We also include the state-of-the-art results of Artetxe et al. (2018a) and Chen and Cardie (2018) for reference. The EURMUSEWIKI benchmark evaluates BLI performance on all language pair combinations of its six languages and does this in both directions (EN-DE, DE-EN, EN-ES, ... IT-PT, PT-IT) yielding 28 $P@1$ scores per model. For clarity, we report the average $P@1$ scores per language as well as the global $P@1$ average. Following Experiment 1, all results for SHS are obtained using value dropping (again averaged across 5 different runs), while we do not use it with IHS. The SHS hub language is English, and the language orders for IHS are EN, DE, IT, ES, FI for DINUARTETXE, and EN, DE, ES, FR, IT, PT for EURMUSEWIKI.

The scores reveal that reweighting the embedding spaces is indeed still beneficial for BLI when mapping to a multilingual space. Both SHS and IHS obtain best results with the reweighting coefficient $q = 0.5$. When comparing SHS and IHS, we see that for language pairs involving English (the SHS hub language) SHS obtains slightly better results, but for the other language pairs IHS outperforms SHS slightly. This is no surprise as IHS by design incorporates dependencies between all languages when learning the projection matrices, though it is striking that mapping to a single hub language is still a strong BLI baseline. For both datasets IHS obtains BLI performance on par with the state-of-the-art: on DINUARTETXE, SHS and IHS ($q = 0.5$) obtain scores similar to (Artetxe et al., 2018a); on EURMUSEWIKI IHS ($q = 0.5$) slightly outperforms Chen and Cardie (2018) for all languages except Spanish. Although optimizing BLI performance is not the main goal of this work, these results verify the soundness of our methods.

Experiment 3: Language Order. Next, we investigate 1) the influence of the hub language choice for SHS, and 2) the impact of the language order for IHS. We run both SHS and IHS with reweighting 0.5 on DINUARTETXE. SHS is with value dropping (results are again averaged over 5 runs), and for IHS we do not use value dropping.

We find that the SHS model is sensitive to the hub language: the best average scores are obtained when using English (41.6%) or German (41.0%).

[13] For instance, when using IHS with a language order that starts with English and Finnish, value dropping still prevents bad performance for Finnish. However, this is not a problem in practice as the language order can be easily predetermined according to various language similarity heuristics.

[14] We further validated the robustness of our approach on other distant language pairs but moved this experiment to the appendix due to space constraints.

Model	q	EN-DE	EN-IT	EN-ES	EN-FI	All
Artetxe	2×0.5	48.13	**48.19**	37.33	**32.63**	**41.57**
SHS	0	47.51	45.60	36.33	31.92	40.34
IHS	0	47.93	45.93	36.07	31.04	40.24
SHS	0.5	**48.69**	47.67	37.51	32.40	**41.57**
IHS	0.5	48.60	47.73	37.53	31.74	41.40
SHS	1	47.77	47.91	37.00	31.82	41.13
IHS	1	48.00	**48.00**	**37.93**	31.46	41.35
IHS order	0.5	48.60	47.81	**38.24**	**33.22**	**41.96**

Table 2: BLI $P@1$ scores on DINUARTETXE: SHS and IHS are evaluated for different values of the reweighting parameter q. The state-of-the-art results (Artetxe et al., 2018a) are added as a reference. The result with the highest average score obtained when trying different language orders is in the bottom row.

Model	q	EN	DE	ES	FR	IT	PT	All
UME		79.57	70.46	**82.88**	82.01	80.69	80.13	79.29
SHS	0	80.04	68.24	80.95	80.10	78.71	77.96	77.67
IHS	0	79.61	69.52	82.35	81.26	80.00	79.02	78.63
SHS	0.5	**80.34**	70.16	82.15	81.65	80.31	79.78	79.07
IHS	0.5	79.91	**70.77**	82.68	**82.08**	**81.08**	**80.47**	**79.50**
SHS	1	79.61	69.59	81.53	81.10	79.74	79.47	78.51
IHS	1	79.05	69.99	81.63	81.23	80.13	79.68	78.62

Table 3: BLI $P@1$ scores on EURMUSEWIKI averaged per language: SHS and IHS are tested for different values of the reweighting parameter q. The results of Chen and Cardie (2018) (UME) are added as a reference.

With Italian, the average score drops to 40.4%, mainly due to worse performance on English and German. With Spanish, the average score further drops to 31.6%, as Spanish and Finnish completely fail to align even when using value dropping. With Finnish, EN-ES alignment becomes unstable, while $P@1$ for EN-DE and EN-IT drops 5.5% and 3.7% compared to the case with English as the hub. These results indicate that the hub language has to be chosen carefully to avoid instability issues.

For IHS, we evaluate all 120 order permutations on DINUARTETXE. The best performing order (EN-DE-ES-FI-IT) achieves an average accuracy of 41.96%, see the last row in Table 2. The full results, not reported due to space constraints[15] confirm our hypothesis that distant languages (Finnish) should be mapped at the end: when using Finnish as one of the first two languages performance drops significantly. EN-FI scores drop below 1% and the results for all other language pairs are also suboptimal.

Experiment 4: Downstream Tasks. In this experiment, we investigate the effect of reweighting input word embeddings on downstream model performance, and compare SHS and IHS to several supervised methods that use cross-lingual supervision. Table 4 reports the results for SHS and IHS with q set to 0 and 0.5 on the REUTERSMLDC and MLPARSING benchmarks,[16] along with the results from related work. For SHS the hub language is English and for IHS the language order is English, German, Spanish, Italian, French, Bulgarian, Czech, Danish, Finnish, Greek, Hungarian, and Swedish. We again use SHS with value dropping and IHS without it. The results in Table 4 are comparable: all methods were trained on the same text corpora (i.e., the collections of Ammar et al. (2016)), but our methods do not use parallel corpora nor bilingual dictionaries.

Model	q	PARSING	MLDC
Invariance (Huang et al., 2015)		59.80	91.10
MultiSkip (Luong et al., 2015)		57.70	90.40
MultiCluster (Ammar et al., 2016)		61.00	92.10
MultiCCA (Ammar et al., 2016)		58.70	92.10
(Duong et al., 2017)		61.20	90.80
SHS	0	63.48	92.59
IHS	0	**65.77**	**92.72**
SHS	0.5	62.23	92.63
IHS	0.5	63.42	92.56

Table 4: Results on the MLPARSING (dependency parsing) and REUTERSMLDC (document classification) benchmarks: SHS and IHS are compared with and without reweighting and we show the state-of-the-art results of supervised embedding mapping methods as a reference. The results for Invariance, MultiSkip, MultiCluster, MultiCCA are from (Ammar et al., 2016).

A first interesting result is that, contrary to the BLI task, reweighting the embeddings is not beneficial for multilingual dependency parsing and document classification. This can be explained by the fact that the reweighted embedding spaces are no longer isomorphic to the original monolingual embedding spaces, hence important patterns in the embedding space could be distorted. Further, we notice that both SHS and IHS improve over the best reported results on the REUTERSMLDC and MLPARSING benchmarks. This result is surprising given that all the reported baselines require supervision to induce the multilingual embedding spaces. Further, we again find that the best results are obtained with IHS, most notably for dependency parsing for which the difference in UAS scores between

[15] The full results can be found in Appendix A.2.

[16] Since the languages covered in MLPARSING is a superset of the languages in REUTERSMLDC, we use the same multilingual embedding space for both tasks.

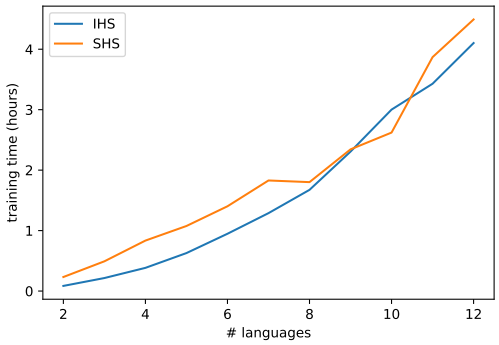

Figure 1: Training times for SHS and IHS for an increasing number of languages.

the best IHS and SHS models is 2.29%.

Experiment 5: Time Complexity. In this experiment, we study how training time of SHS and IHS behaves as a function of the number of languages that are mapped. The singular value decompositions are more expensive for IHS as the matrices grow linearly with the number of languages (see Eq. (14)) whereas for SHS they are constant. On the other hand, IHS does not require the use of value dropping. Figure 1 plots training time for IHS and SHS conditioned on the number of languages. The estimates are based on training on a single Nvidia Titan Xp GPU. We find that SHS with value dropping and IHS without value dropping have similar training times, IHS being a bit more efficient when mapping 7 languages or less. Mapping 12 languages takes approximately four hours for both methods.

7 Conclusion

We proposed two novel methods for learning multilingual word embeddings (MWEs) without any cross-lingual supervision. The better-performing incremental hub space model (IHS) is the first unsupervised MWE method that combines three desirable properties: 1) It incorporates interdependencies between all targeted languages; 2) It works for distant language pairs; and 3) It is both deterministic and robust, that is, it does not produce degenerate solutions. Our evaluation on standard benchmarks has proven that the IHS method induces multilingual word embeddings that are competitive with the state of the art in bilingual lexicon induction. Moreover, we have shown that IHS outperforms even supervised models on downstream tasks of multilingual dependency parsing and document classification, and this anomaly requires further investigation in future work. Furthermore, we looked at the influence of reweighting the dimensions of the embedding spaces according to their cross-correlations with the hub language space(s) and found that, while it improves performance for the BLI task, it is harmful to downstream cross-lingual transfer tasks. These empirical observations stress the requirement to include comprehensive evaluation protocols for cross-lingual word embedding models in future research.

Acknowledgments

We thank the reviewers for their insightful comments. IV would like to thank Goran Glavaš and Anders Søgaard for interesting discussions, and the ERC Consolidator Grant LEXICAL (no 648909) for the support. GH, BV and MFM would like to thank Stijn Jaques, Evelyn Reynders and Evelien Verbaenen for the fruitful collaboration and Flanders Innovation & Entrepreneurship (VLAIO) for funding TELLMI within the ITEA 3 project PAPUD.

References

Željko Agić, Maria Jesus Aranzabe, Aitziber Atutxa, Cristina Bosco, Jinho Choi, Marie-Catherine de Marneffe, Timothy Dozat, Richrd Farkas, Jennifer Foster, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, Jan Hajič, Anders Trrup Johannsen, Jenna Kanerva, Juha Kuokkala, Veronika Laippala, Alessandro Lenci, Krister Lindén, Nikola Ljubešić, Teresa Lynn, Christopher Manning, Hctor Alonso Martínez, Ryan McDonald, Anna Missilä, Simonetta Montemagni, Joakim Nivre, Hanna Nurmi, Petya Osenova, Slav Petrov, Jussi Piitulainen, Barbara Plank, Prokopis Prokopidis, Sampo Pyysalo, Wolfgang Seeker, Mojgan Seraji, Natalia Silveira, Maria Simi, Kiril Simov, Aaron Smith, Reut Tsarfaty, Veronika Vincze, and Daniel Zeman. 2015. Universal Dependencies 1.1.

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A. Smith. 2016. Massively Multilingual Word Embeddings. *CoRR abs/1602.01925*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning Principled Bilingual Mappings of Word Embeddings while Preserving Monolingual Invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2289–2294.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017a. Learning Bilingual Word Embeddings with (Almost) no Bilingual Data. In *Proceedings of the*

55th Annual Meeting of the Association for Computational Linguistics (ACL), pages 451–462.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. 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 (ACL)*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. Generalizing and Improving Bilingual Word Embedding Mappings with a Multi-Step Framework of Linear Transformations. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pages 5012–5019.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018c. Unsupervised Statistical Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017b. Unsupervised Neural Machine Translation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–11.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky Wide Web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Anthony Bell and Terrence Sejnowski. 1997. The 'Independent Components' of Natural Scenes are Edge Filters. *Vision Research*, pages 3327–3338.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. *CoRR abs/1607.04606*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the 10th Workshop on Statistical Machine Translation (WMT)*, pages 1–46, Lisbon, Portugal. Association for Computational Linguistics.

Hailong Cao, Tiejun Zhao, Shu Zhang, and Yao Meng. 2016. A Distribution-based Model to Learn Bilingual Word Embeddings. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*.

Xilun Chen and Claire Cardie. 2018. Unsupervised Multilingual Word Embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 261–270.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Herv Jégou. 2018. Word Translation Without Parallel Data. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–14.

Arthur Dempster, Nan Laird, and Donald Rubin. 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2015. Improving Zero-shot Learning by Mitigating the Hubness Problem. In *Proceedings of the International Conference on Learning Representations (ICLR), workshop track*.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2017. Multilingual Training of Crosslingual Word Embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 894–904.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-Based Dependency Parsing with Stack Long Short-Term Memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL)*, pages 334–343.

Manaal Faruqui and Chris Dyer. 2014. Improving Vector Space Word Representations Using Multilingual Correlation. *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 462–471.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BilBOWA: Fast Bilingual Distributed Representations without Word Alignments. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 748–756.

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 the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 1085–1095.

Kejun Huang, Matt Gardner, Evangelos Papalexakis, Christos Faloutsos, Nikos Sidiropoulos, Tom Mitchell, Partha P Talukdar, and Xiao Fu. 2015. Translation Invariant Word Embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1084–1088.

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

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing Crosslingual Distributed Representations of Words. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 1459–1474.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised Machine Translation Using Monolingual Corpora Only. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–12.

Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. 2015. Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL)*, pages 270–280.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual Word Representations with Monolingual Quality in Mind. In *Proceedings Workshop on Vector Modeling for NLP, NAACL 2015*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 27th Annual Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013b. Exploiting Similarities among Languages for Machine Translation. In *CoRR, abs/1309.4168*.

Milo Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. 2010. Hubs in Space: Popular Nearest Neighbors in High-dimensional Data. *Journal of Machine Learning Research*, 11:2487–2531.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2018. A Survey of Cross-lingual Embedding Models. *Journal of Artificial Intelligence Research (JAIR)*.

Samuel Smith, David Turban, Steven Hamblin, and Nils Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–10.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the Limitations of Unsupervised Bilingual Dictionary Induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ivan Vulić and Anna Korhonen. 2016. On the Role of Seed Lexicons in Learning Bilingual Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 247–257.

Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and Cross-Lingual Information Retrieval Models Based on (Bilingual) Word Embeddings. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 363–372.

Ivan Vulić and Marie-Francine Moens. 2016. Bilingual Distributed Word Representations from Document-Aligned Comparable Data. *Journal of Artifical Intelligence Research*, 55:953–994.

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 (ACL)*, pages 1959–1970.

A Appendices

A.1 Experiment with Distant Language Pairs

In this section, we report an additional bilingual lexicon induction experiment that further supports our claim that the IHS model is still applicable when mapping languages with different characteristics, we performed an additional bilingual lexicon induction experiment on the following language pairs: Dutch-Turkish, Spanish-Hungarian, and Finnish-Bulgarian. We induced multilingual spaces with the IHS model and the model of Chen and Cardie (2018) using the publicly available monolingual embeddings trained with fastText. IHS is run with reweighting ($q = 0.5$), without value dropping, and with the following language order Dutch, Spanish, Bulgarian, Finnish, Hungarian, Turkish. The model of Chen and Cardie (2018)[17] is run with Dutch as the target language and with the recommended hyper-parameters. For evaluation, we obtained dictionaries for each language pair using Panlex (Kamholz et al., 2014).

The results are reported in Table 5. We find average BLI accuracy scores of 28.24% (IHS) and 28.20% (Chen and Cardie, 2018). The fact that both models are robust to distant languages without using value dropping further supports our hypothesis that mapping multiple languages simultaneously is an effective regularization mechanism.

A.2 Results Experiment 3

In this section, we report all the results of Experiment 3 of the paper. In Table 6 we report the results for the SHS model with different hub languages and in Table 7 we report the performance for IHS

[17] https://github.com/ccsasuke/umwe

Model	NL-TR	ES-HU	FI-BG	Avg
IHS	22.67	30.21	31.84	28.24
UME	21.51	31.26	31.84	28.20

Table 5: Precision@1 scores for a bilingual lexicon experiment on three distant language pairs.

Hub language	EN-DE	EN-IT	EN-ES	EN-FI	Average
EN	48.69	47.67	37.51	32.40	41.57
DE	47.75	46.65	38.15	31.47	41.01
IT	45.88	47.15	37.88	30.65	40.39
ES	43.77	45.27	36.63	0.57	31.56
FI	43.15	43.92	6.71	33.20	31.74

Table 6: Influence of the hub language of the SHS model on precision@1 scores evaluated on the DINUARTETXE BLI dataset.

with all possible language orders (sorted by precision@1). Both the SHS and IHS models are run with reweighting ($q = 0.5$), SHS is run with value dropping and IHS is run without value dropping.

Order	EN-DE	EN-ES	EN-FI	EN-IT	Avg
EN DE ES FI IT	48.60	38.20	33.22	47.80	41.96
EN DE FI ES IT	48.60	37.73	33.50	47.93	41.94
DE IT EN FI ES	48.40	38.07	33.22	47.93	41.91
EN DE ES IT FI	48.60	38.20	32.72	47.60	41.78
EN IT DE ES FI	48.33	38.73	32.51	47.53	41.78
EN IT ES DE FI	48.07	38.47	33.01	47.53	41.77
EN DE IT FI ES	48.60	37.53	33.22	47.73	41.77
EN DE FI IT ES	48.60	37.13	33.50	47.67	41.73
EN IT DE FI ES	48.33	38.20	32.79	47.53	41.71
DE IT EN ES FI	48.40	38.00	32.51	47.93	41.71
DE EN FI IT ES	48.40	37.60	33.29	47.53	41.71
DE EN FI ES IT	48.40	37.40	33.29	47.67	41.69
DE EN IT FI ES	48.40	37.73	33.22	47.27	41.66
DE EN ES IT FI	48.40	38.20	32.65	47.33	41.65
EN IT FI DE ES	48.53	37.87	32.65	47.53	41.65
DE IT ES EN FI	48.27	38.07	32.30	47.73	41.59
DE EN ES FI IT	48.40	38.20	32.37	47.40	41.59
DE IT FI ES EN	47.80	37.93	33.22	47.40	41.59
DE EN IT ES FI	48.40	37.53	33.08	47.27	41.57
EN IT FI ES DE	47.80	38.07	32.65	47.53	41.51
EN DE IT ES FI	48.60	37.67	32.02	47.73	41.51
DE ES IT FI EN	47.27	38.33	32.87	47.40	41.47
IT EN DE FI ES	47.40	38.47	32.65	47.07	41.40
DE ES FI IT EN	47.33	38.53	32.30	47.33	41.37
DE ES IT EN FI	47.27	38.40	31.88	47.80	41.34
DE ES EN FI IT	47.67	37.87	32.02	47.67	41.31
DE IT ES FI EN	47.40	37.87	32.65	47.27	41.30
EN ES DE FI IT	47.93	37.53	32.09	47.53	41.27
EN IT ES FI DE	47.87	38.47	31.18	47.53	41.26
DE IT FI EN ES	47.60	38.07	32.37	46.87	41.23
DE ES EN IT FI	47.67	37.87	31.67	47.53	41.19
IT DE EN FI ES	47.00	38.00	33.15	46.53	41.17
IT ES DE EN FI	47.53	38.47	32.37	46.27	41.16
DE ES FI EN IT	47.47	37.60	32.44	46.87	41.10
IT EN DE ES FI	47.40	38.40	31.39	47.07	41.07
IT EN FI DE ES	47.00	38.27	31.88	47.07	41.06
EN ES DE IT FI	47.93	37.53	30.90	47.60	40.99
ES DE IT EN FI	46.67	38.27	31.95	47.07	40.99
IT EN ES DE FI	47.20	37.93	31.74	47.07	40.99
ES IT DE FI EN	47.20	37.40	32.23	46.73	40.89
IT EN FI ES DE	46.47	38.00	31.88	47.07	40.86
EN ES IT FI DE	47.73	37.53	30.76	47.20	40.81
IT DE EN ES FI	47.00	37.93	31.74	46.53	40.80
EN ES IT DE FI	47.33	37.53	31.04	47.20	40.78
IT DE ES EN FI	46.67	38.07	32.02	46.33	40.77
ES IT DE EN FI	46.60	37.73	31.46	47.27	40.77
IT DE FI ES EN	46.80	37.73	32.51	45.87	40.73
ES DE IT FI EN	46.47	37.27	32.23	46.87	40.71
ES DE EN FI IT	46.67	37.67	31.32	47.13	40.70
IT EN ES FI DE	46.40	37.93	31.11	47.07	40.63
ES DE EN IT FI	46.67	37.67	30.69	47.27	40.58
IT DE ES FI EN	46.53	37.40	32.02	46.20	40.54
IT ES DE FI EN	46.53	38.27	31.32	45.93	40.51
IT DE FI EN ES	46.27	37.60	32.02	45.93	40.46
ES IT EN DE FI	46.20	37.20	31.18	46.93	40.38
ES EN IT FI DE	46.87	36.80	30.83	46.67	40.29
ES EN DE FI IT	46.40	36.80	31.32	46.53	40.26
ES EN DE IT FI	46.40	36.80	31.32	46.47	40.25
ES EN IT DE FI	46.13	36.80	31.11	46.67	40.18
ES IT EN FI DE	45.67	37.20	30.55	46.93	40.09
ES DE FI IT EN	45.67	36.80	30.83	46.87	40.04
IT ES EN DE FI	45.33	37.60	30.27	46.20	39.85
IT ES EN FI DE	45.20	37.60	30.34	46.20	39.84
ES DE FI EN IT	45.20	37.07	30.27	46.60	39.79
EN ES FI DE IT	47.27	37.53	0.21	47.33	33.09
DE FI IT EN ES	48.07	37.33	0.07	46.80	33.07
EN FI DE IT ES	46.93	37.40	0.14	47.47	32.99
EN FI DE ES IT	46.93	37.73	0.14	47.07	32.97
EN ES FI IT DE	47.33	37.53	0.21	46.80	32.97
DE FI IT ES EN	47.20	37.73	0.07	46.73	32.93
DE FI ES EN IT	47.47	37.40	0.07	46.40	32.84
DE FI ES IT EN	47.80	37.20	0.07	46.07	32.79
DE FI EN IT ES	46.73	37.27	0.14	46.60	32.69
DE FI EN ES IT	46.73	37.40	0.14	46.33	32.65
IT FI DE ES EN	46.13	37.67	0.14	46.20	32.54
ES EN FI DE IT	46.60	36.80	0.35	46.20	32.49
ES FI DE IT EN	44.67	38.13	0.07	46.93	32.45
ES EN FI IT DE	46.60	36.80	0.35	45.47	32.31
IT FI ES DE EN	45.60	36.93	0.14	46.47	32.29
ES FI IT DE EN	45.60	36.93	0.00	46.53	32.27
IT ES FI DE EN	45.00	37.60	0.21	45.67	32.12
IT FI DE EN ES	45.80	37.73	0.14	44.80	32.12
EN FI IT DE ES	45.13	36.60	0.14	46.13	32.00
EN FI IT ES DE	45.07	36.33	0.14	46.13	31.92
ES FI DE EN IT	44.07	37.13	0.07	46.07	31.84
ES IT FI DE EN	44.13	36.60	0.07	46.33	31.78
ES FI IT EN DE	44.80	36.73	0.00	45.47	31.75
IT FI EN ES DE	45.07	36.20	0.14	45.47	31.72
IT FI ES EN DE	44.60	36.53	0.14	45.33	31.65
IT FI EN DE ES	44.93	35.80	0.14	45.47	31.59
IT ES FI EN DE	44.20	36.93	0.21	44.93	31.57
ES IT FI EN DE	43.33	36.13	0.07	45.93	31.37
FI DE IT EN ES	42.93	37.13	0.42	44.73	31.30
FI DE IT ES EN	43.20	36.67	0.28	45.00	31.29
FI IT DE EN ES	44.40	36.67	0.07	43.67	31.20
ES FI EN DE IT	43.73	35.27	0.14	45.00	31.04
ES FI EN IT DE	43.60	35.27	0.14	44.67	30.92

Order	EN-DE	EN-ES	EN-FI	EN-IT	Avg
FI DE ES IT EN	42.27	35.07	0.14	45.33	30.70
FI IT ES DE EN	44.07	35.07	0.07	43.33	30.64
FI DE EN IT ES	41.27	36.40	0.14	44.33	30.54
FI IT DE ES EN	43.33	35.40	0.07	43.13	30.48
FI ES IT DE EN	43.60	33.60	0.21	44.40	30.45
FI DE EN ES IT	41.27	35.27	0.14	44.60	30.32
FI IT ES EN DE	43.07	34.87	0.14	42.47	30.14
FI ES DE IT EN	42.13	32.67	0.14	45.00	29.99
FI DE ES EN IT	42.73	31.47	0.14	45.47	29.95
FI EN IT DE ES	42.60	32.67	0.28	41.53	29.27
FI ES IT EN DE	42.07	31.93	0.21	42.27	29.12
FI ES DE EN IT	42.00	29.13	0.14	44.60	28.97
FI EN IT ES DE	40.80	31.33	0.28	41.53	28.49
FI EN DE IT ES	35.27	30.33	0.28	40.27	26.54
FI EN DE ES IT	35.27	29.40	0.28	40.07	26.26
EN FI ES DE IT	46.53	0.73	0.14	47.13	23.63
EN FI ES IT DE	47.00	0.73	0.14	44.87	23.19
FI EN ES DE IT	17.27	0.07	0.28	20.33	9.49
FI IT EN DE ES	9.13	6.60	0.00	2.53	4.57
FI IT EN ES DE	5.20	3.93	0.00	2.53	2.92
FI ES EN IT DE	0.33	0.13	0.35	0.67	0.37
FI ES EN DE IT	0.27	0.13	0.35	0.47	0.31
FI EN ES IT DE	0.33	0.07	0.28	0.13	0.20

Table 7: Influence of language order of the IHS model on precision@1 scores evaluated on the DINUARTETXE BLI dataset.