# On a Novel Application of Wasserstein-Procrustes for Unsupervised Cross-Lingual Alignment of Embeddings

**Guillem Ramírez*[1], Rumen Dangovski*[1], Preslav Nakov[2], Marin Soljačić[1]**
Massachusetts Institute of Technology (MIT)[1]
Mohamed bin Zayed University of Artificial Intelligence (MBZUAI)[2]
gramirez@ed.ac.uk

## Abstract

Unsupervised word embeddings, pre-trained on vast monolingual text corpora, have driven the neural revolution in Natural Language Processing (NLP). Initially developed for English, these embeddings soon expanded to other languages, spurring efforts to align embedding spaces for cross-lingual NLP applications. Unsupervised cross-lingual alignment of embeddings (UCAE) is particularly appealing due to its minimal data requirements and competitive performance against supervised and semi-supervised approaches. In this work, we scrutinize prevalent UCAE methods and discover their objectives inherently resemble the Wasserstein-Procrustes problem. Consequently, we propose a direct solution for Wasserstein-Procrustes, enhancing popular UCAE techniques such as iterative closest point (ICP), multilingual unsupervised and supervised embeddings (MUSE), and supervised Procrustes methods. Evaluation on benchmark datasets demonstrates significant improvements over existing approaches. Our reexamination of the Wasserstein-Procrustes problem fosters further research, paving the way for more effective algorithms to align word embeddings across languages.

**Keywords:** Wasserstein-Procrustes, cross-lingual embeddings, unsupervised alignment

## 1. Introduction

Pre-trained word embeddings, which map words to dense vectors of low dimensionality, have been the key enabler of the ongoing neural revolution, and today they serve as the basic building blocks of contemporary Natural Language Processing (NLP) models. While initially introduced for English (Mikolov et al., 2013a; Pennington et al., 2014; Bojanowski et al., 2017; Joulin et al., 2017), pre-trained embeddings quickly emerged for a number of other languages (Heinzerling and Strube, 2018), and the idea of cross-language embedding spaces was born. In a cross-language embedding space, two semantically similar (or dissimilar) words would be close to (or far from) each other regardless of whether they are from the same or from different languages. Using such a space is attractive, as for a number of NLP tasks, it enables the application of an NLP model trained for one language on input from another language.

Ideally, such spaces could be trained on parallel bilingual datasets, but such resources are of limited size, e.g., compared to the large-scale monolingual resources typically used to pre-train monolingual word embeddings. Thus, it has been more attractive to train monolingual word embeddings for different languages independently, and then to try to align the corresponding embedding spaces in what is commonly known as bilingual lexicon induction. This has been attempted in a supervised (Mikolov et al., 2013b; Irvine and Callison-Burch, 2013), in a semi-supervised (Artetxe et al., 2017), and in an unsupervised setting (Lample et al. (2017); Lample and Conneau (2019); Alipour et al. (2022); Feng et al. (2022); Tian et al. (2022); Liang et al. (2023); Li et al. (2023); Liu and Piccardi (2023); Ghayoomi (2023); Ghazvininejad et al. (2023)).

Initial space alignment efforts used word translation pairs as anchors, inferring transformations between languages in a supervised setup (Mikolov et al., 2013b). The alignment employs an orthogonal transformation minimizing the Frobenius norm in the Procrustes problem, with a closed-form solution obtainable via SVD. For the translation of word embeddings, $W$ is taken to be an orthogonal matrix due to a self-similarity argument (Smith et al., 2017). The convenience of using an orthogonal matrix has also been supported empirically (Xing et al., 2015; Zhang et al., 2016; Artetxe et al., 2016). The orthogonal Procrustes problem has a closed-form solution $W = UV^\top$, where $U\Sigma V^\top$ is the singular value decomposition (SVD) of $X^\top Y$ as shown by Schönemann (1966).

**Procrustes** Given two ordered clouds of points $X, Y \in \mathbb{R}^{N\times d}$, each with $N$ points of dimension $d$, the orthogonal Procrustes problem finds the orthogonal matrix $W \in \mathbb{R}^{d\times d}$ that minimizes the following Frobenius norm:

$$\arg\min_{W\in O(d)} \|XW - Y\|_2^2 \tag{1}$$

A popular unsupervised formulation of the problem is known as the Wasserstein-Procrustes (Grave et al., 2019; Alaux et al., 2019), which is more challenging as it needs to optimize a generalization

[*] Equal contribution.

*BUCC 2024: The 17th Workshop on Building and Using Comparable Corpora*, pages 1–11
20 May, 2024.

of the Procrustes objective. One-to-one maps are encouraged through a permutation matrix $P$.

The convenience of one-to-one maps is justified for different reasons. First, the hubness problem (Dinu and Baroni, 2014) occurs in high-dimensional vector spaces where certain vectors are the nearest neighbor to a disproportionate number of other vectors, thus reducing the quality of the embedding space (Radovanovic et al., 2010). Second, one-to-one maps can be linked to Wasserstein distance and computational optimal transport.

**Wasserstein-Procrustes** Given two clouds of points $X, Y \in \mathbb{R}^{N\times d}$, each with $N$ points of dimension $d$, the Wasserstein-Procrustes problem finds an orthogonal matrix $W \in \mathbb{R}^{d\times d}$ and a permutation matrix $P \in \mathbb{R}^{N\times N}$ that minimize the Frobenius norm:

$$\operatorname*{arg\,min}_{P\in\pi(N), W\in O(d)} \|XW - PY\|_2^2 \tag{2}$$

where $\pi(N)$ is the set of $N$-dimensional permutation matrices and $O(d)$ is the set of $d$-dimensional orthogonal matrices.

In practice, most approaches modify the objective yet achieve good accuracy in synthetic dictionary induction tasks. We ask: Can we find approximate Wasserstein-Procrustes solutions (Equation 2) with high accuracy in dictionary tasks? Can we enhance existing methods using refinements to optimize Equation 2? Can we identify scenarios with good solutions? We address these questions by analyzing different objective functions in the literature, adhering to Artetxe et al. (2020)'s call for fair model comparison.

## 2. Background: Towards a Unifying Framework

There have been attempts to compare different methods proposed for the Unsupervised Cross-Lingual Alignment of Embeddings, or UCAE (Hartmann et al., 2019), and there have been papers that have tried to generalise the different possibilities one approach could possibly have. Artetxe et al. (2018a) proposed a framework based on different steps and showed how existent methods would fit in it. Ruder et al. (2019) described the most general framework for UCAE. However, we are not aware of a unified description of the existing methods from the point of view of what is being optimized, namely the loss function. We start by analyzing methods based on optimal transport methods, as they are most relevant to our approach.

### 2.1. Optimal Transport Methods

There have been some approaches framing the problem of unsupervised dictionary induction as an optimal transport problem, and this is the approach we will adopt in the following sections. Haghighi et al. (2008) proposed a self-learning method for bilingual lexicon induction, representing words with orthographic and contextual features and using the Hungarian algorithm (Tomizawa, 1971) to find an optimal one-to-one matching.

With the emergence of word embeddings (Mikolov et al., 2013a), words were interpreted as vectors in high-dimensional spaces, and concepts such as distance between words started to gain attention. Ruder et al. (2018) presented Viterbi EM, where words were mapped following a one-to-one map between subsets $X'$ and $Y'$ of $X$ and $Y$, respectively, and the isometry was induced by an orthogonal matrix. They deviated from the Wasserstein-Procrustes objective by including a penalization term for unmatched words $Y'_\perp = Y - Y'$. They did not consider all possible matches, instead imposing a restriction on the $k$ nearest neighbors when running the Jonker-Volgenant algorithm for optimal transport (Jonker and Volgenant, 1987).

Zhang et al. (2017) proposed two different methods: WGAN (an adversarial network that optimizes the Wasserstein distance) and EMDOT (an iterative procedure that uses Procrustes and solves a linear transport problem). Both methods are inspired by the Earth Mover's Distance (EMD), which defines a distance between probability distributions, which they applied to frequencies of words. They found that, although EMDOT could converge to bad local minima, it improved the results when used as a refinement tool after first optimizing with WGAN. Alvarez-Melis and Jaakkola (2018) used the concept of Gromov-Wasserstein distance to provide an alternative to Wasserstein-Procrustes. This distance does not operate on points but on pairs of points, turning the problem of finding optimal matching $\Gamma^*$ from a linear into a quadratic one. This new loss function can be optimized efficiently with first-order methods, whereby each iteration involves solving a traditional optimal transport problem. Artetxe et al. (2018b) achieved better results by combining this idea with a refinement method called stochastic dictionary induction, i.e., randomly dropping dimensions out of the similarity matrix when extracting a seed dictionary for the next iteration of the Procrustes analysis.

### 2.2. Other Methods

Wasserstein-Procrustes is one of the recurring loss functions in the literature, but there have been also deviations from the original problem. Grave et al. (2019) suggested an iterative procedure whose initial condition minimizes the convex relaxation $\left\|X^\top PY\right\|_2^2$ instead of the original problem. This relaxation is known as the Gold-Rangarajan relaxation and can be solved using the Frank-Wolfe algorithm (Gold and Rangarajan, 1996; Frank and Wolfe, 1956). The solution to this relaxation is then used as the initial condition for a gradient-based iterative procedure that stochastically samples different subsets of words for which there is not necessarily a direct translation.

This deviates strongly from Objective 2: not only the initial condition does not optimize the Wasserstein-Procrustes objective, but also the iterative procedure does not optimize it, as it translates words that are not necessarily the optimal matches. Alaux et al. (2019) were also inspired by Objective 2 for aligning multiple languages in a common vector space. However, they minimized a loss function based on the CSLS metric from Lample et al. (2018). In a similar fashion, the entropy regularization of the Gromov-Wasserstein problem (Mémoli, 2011) has been used for bilingual lexicon induction.

Generative Adversarial Network (GAN) optimization was first introduced for bilingual lexicon induction by Barone (2016), but its canonical implementation was given by Lample et al. (2018), who presented *multilingual unsupervised and supervised embeddings* (MUSE), an adversarial method in which the transformation matrix $W$ is considered as a generator, and thus is trained by a generative adversarial network, so that the mapped word embeddings $XW$ cannot be distinguished from the set $Y$ via a discriminator (Goodfellow et al., 2014). However, a simple thought experiment can convince us that this approach does not minimize distances. We elaborate on that experiment in the Appendix.

Hoshen and Wolf (2018) were inspired by the Iterative Closest Point (ICP) method used in 3D point cloud alignment. Although their transformation matrix is not necessarily orthogonal, this property is enforced using the regularization $L(X, Y, W; \lambda) := \lambda\|XWW^\top - X\|_2^2 + \lambda\|YW^\top W - Y\|_2^2$. Another fundamental difference to Objective 2 is that they do not use a one-to-one mapping for $P$.

This list is not exhaustive, as there have been successful methods that do not rely on loss functions, and such that go beyond the geometry of the trained word embeddings. For example, Artetxe et al. (2019) used both the word embeddings and the monolingual corpus used to train them.

To sum up, in Table 1, we list the relevant objectives from above using our formalism from Equation 2. In the table, $\Gamma^*$ is the optimal Gromov-Wasserstein matching, $X'$ and $Y'$ are subsets of the corresponding $X$ and $Y$, $Y'_\perp$ is the complement of $Y'$ in $Y$, and $\overline{Y'_\perp}$ is the average of the complements.

## 3. Properties of the Wasserstein-Procrustes Problem

We begin by simplifying Objective 2 to arrive at some essential properties, described below.

**Proposition 1 (Grave et al. (2019))** *The Wasserstein-Procrustes problem is equivalent to maximizing the trace norm on the permutation matrix $X^\top PY$ over $P$, described as follows:*

$$\operatorname*{arg\,min}_{P\in\pi(N), W\in O(d)} \|XW - PY\|_2^2 = \operatorname*{arg\,max}_{P\in\pi(N)} \left\|X^\top PY\right\|_* \tag{3}$$

*where $\|\cdot\|_*$ denotes the nuclear norm and $W$ is selected, so that it fulfills that $U^\top WV = \mathbb{I}_d$, where both $U(P)$ and $V(P)$ are evaluated at a matrix $P^*$ that achieves the optimum of Equation 3.*

**Hungarian algorithm** Given two clouds of points $X, Y \in \mathbb{R}^{N\times d}$, each with $N$ points of $d$ dimensions, the Hungarian algorithm finds the permutation matrix $P$ that gives the correspondence between the different points by solving the following problem:

$$\operatorname*{arg\,min}_{P\in\pi(N)} \|X - PY\|_2^2. \tag{4}$$

Replacing $W$ in Proposition 1 with the identity matrix $\mathbb{I}_d$ and noting that $\langle \mathbb{I}_d, X^\top PY\rangle_2 = \operatorname{Tr}\left(X^\top PY\right)$ holds for the Frobenius inner product, we obtain the following:

**Corollary 1** *Problem 4 is equivalent to maximizing the trace of $X^\top PY$ over $P$:*

$$\operatorname*{arg\,min}_{P\in\pi(N)} \|X - PY\|_2^2 = \operatorname*{arg\,max}_{P\in\pi(N)} \operatorname{Tr}\left(X^\top PY\right), \tag{5}$$

*which is the maximum weight matching problem. The latter can be solved using the Hungarian algorithm, which has a complexity of $O(N^3)$ (Tomizawa, 1971).*

Even though the Hungarian algorithm has cubic complexity, we could still run it feasibly for $N = 45,000$. In principle, our refinement methods work well by using a subset of the full vocabulary, which typically has $N = 200,000$ words. Speedups of the Hungarian algorithm and approximations could be pursued in future work.

**Equivalent problems** One useful property of the trace norm is that $\|UA\|_* = \|AV\|_* = \|A\|_*$, where $U$ and $V$ are orthogonal matrices. Knowing this, and writing $U_X\Sigma_X V_X^\top$ and $U_Y\Sigma_Y V_Y^\top$ as the SVD decompositions for $X$ and $Y$, respectively, we obtain the following:

$$\left\|X^\top PY\right\|_* = \left\|V_X\Sigma_X U_X^\top P U_Y \Sigma_Y V_Y^\top\right\|_* \tag{6}$$

| Method | Objective |
|---|---|
| Grave et al. (2019) and Ours | $\min_{W\in O(d),P\in\pi(N)} \lVert XW - PY\rVert_2^2$ |
| Alvarez-Melis and Jaakkola (2018) | $\min_{\Gamma^* \text{ best coupling}, W\in O(N)} \lVert X\Gamma^* - WY\rVert_2^2$ |
| Hoshen and Wolf (2018) | $\min_{W\in O(d)} \lVert XW - Y\rVert_2^2 + \lVert YW^\top - X\rVert_2^2 + L(X,Y,W;\lambda)$ |
| Ruder et al. (2018) | $\min_{W\in O(d),P'\in\pi(N')} \lVert X'W - P'Y'\rVert_2^2 + \left\lVert Y'_\perp - \overline{Y'_\perp}\right\rVert_2^2$ |
| Lample et al. (2017) | $\min_W \max_{\theta_D} \mathbb{P}_{\theta_D}(\text{source}\mid WX)\mathbb{P}_{\theta_D}(\text{target}\mid Y)$ |
| Zhang et al. (2017) | $\min_{W\in O(d),P\in\pi(N)} \sum_{i=1,j=1}^{N,N} P_{i,j} \left((X_iW)_j - Y_i\right)^2$ |

Table 1: Objective functions of relevant existing methods in the language of our formalism.

which yields

$$\underset{P\in\pi(N)}{\arg\max} \left\lVert \Sigma_X U_X^\top P U_Y \Sigma_Y \right\rVert_* . \tag{7}$$

Let us define $\widetilde{X} = U_X\Sigma_X$ and $\widetilde{Y} = U_Y\Sigma_Y$. Then, the optimal solution $P$ would be the same for translations involving all of the following pairs of word embeddings: $(X, Y)$, $(\widetilde{X}, Y)$, $(X, \widetilde{Y})$ and $(\widetilde{X}, \widetilde{Y})$. However, the optimal transformation matrix $W^*$ will be different for each of these problems. There is a different, yet interesting way of looking at this: if we follow the iterative procedure that starts from an initial transformation matrix $X_0 = XW_0$ (where $W_0$ is our initial approximation to the transformation matrix), and then we want to solve Problem (5), the equivalent problems will induce a set of *natural initializations* of the transformation $W$, which we formalize below:

> *Given the iterative procedure that tries to minimize the Wasserstein-Procrustes objective by first obtaining the permutation matrix $P_n = \arg\min_{P\in\pi(N)} \operatorname{Tr}(X_n^\top P Y_n)$ and then the transformation matrix $W_n = \arg\min_{W\in\mathbb{R}^{N\times N}} \lVert X_nW - P_nY_n\rVert_2^2$, the procedure aims for the same solution $P$ as the problems with initial conditions $X_0 = XW_0$, $X_0 = XV_XW_0$, $X_0 = XW_0V_Y^\top$, $X_0 = XV_XW_0V_Y^\top$.*

The significance of the different natural initialization is that it gives us a starting point for different problems that have the same solution $P$. It must be noted, however, that these transformations of $X_0$ are not the unique ones that will have the same original solution, as the trace norm is invariant to any orthogonal transformation; however, they help to avoid bad local minima as we will show in Section 5 below. Another way of looking at these initialization is that we are performing PCA to the embedding matrices without a dimensionality reduction. Hoshen and Wolf (2018) proposed using PCA in a similar context.

## 4. Approach

Below, we present a general iterative algorithm to solve the Wasserstein-Procrustes problem.

**Joint optimization on $W$ and $P$.** For the Wasserstein-Procrustes problem from Equation 2, a joint iterative procedure involving the Procrustes problem and the Hungarian algorithm (see Algorithm 1) has been dismissed due to its computational cost and convergence to bad local minima (Zhang et al., 2017). However, as we will show below, there are a number of situations where such an approach can be extremely beneficial if we apply some improvements based on the discussion in the previous section.

**Algorithm 1** *Cut Iterative Hungarian (CIH) Algorithm*

```
1. We initialize as follows: $X \leftarrow XW_0$.
2. We find $P \leftarrow$ Hungarian$(X, Y)$ and $W \leftarrow$ Procrustes$(X, PY)$.
3. If the trace norm has increased, update $X_{NEW} \leftarrow XW$ and $Y_{NEW} \leftarrow PY$, repeat Step 2.
```

**Variants of the natural initializations.** The first improvement is to consider the different equivalent problems or the natural initialization transformations, mentioned in the previous section. We observe empirically that apart from the four problems that share the same optimal $P$, it is possible to improve the results by considering the opposite optimization problem: instead of maximizing the costs for the two clouds of points $(X, Y)$, sometimes *minimizing* the costs yields a solution with a higher trace norm, and thus the algorithm eventually converges to a better solution. The matrix $X^\top PY$ is generally not symmetric with non-negative eigenvalues, and thus the trace norm and the trace are not the same. The minimization is achieved by simply considering the cloud $-X$ instead of $X$. Algorithm 2 is the most general iterative procedure that we consider here, and it serves as the backbone for our experiments below:

**Algorithm 2** *Iterative Hungarian (IH) Algorithm. It is the same as Algorithm 1, but in Step 2 we also consider the solutions for four natural initializations:* $X_0 = XW_0$, $X_0 = XV_XW_0$, $X_0 = XW_0V_Y^\top$, $X_0 =$

*$XV_XW_0V_Y^\top$, also considering the cloud $-X$ for the four different initializations.*

**Supervised translation.** Although the scope of this paper is the unsupervised cross-lingual alignment of embeddings, we also decided to run some experiments that involve minimal supervision. There are different ways of doing this, but the procedure that converges the fastest is to fix $n$ pairs of words when calculating the Hungarian map, where typically $n \ll N$. We also consider similar approaches, e.g., deciding how to update Algorithm 2, taking into account the accuracy of the maps on a small subset of the data. Choosing among these methods could be motivated by how trustworthy the initial dictionary is. By *trustworthy* here we mean how many of the corresponding cloud points are correctly matched.

We use a fast implementation of the Hungarian algorithm[1] for dense matrices based on shortest path augmentation (Edmonds and Karp, 1972). Relaxations of the original problem can achieve higher speed ups. Cuturi (2013) showed how smoothing the classical optimal transport problem with an entropic regularization term results in a problem that can be solved using the Sinkhorn-Knopp's matrix scaling algorithm (Sinkhorn and Knopp, 1967) at a speed that is orders of magnitude faster than that of transportation solvers.

**Mapping.** Although our method finds a permutation matrix $P$, this is not necessarily the best possible mapping as the set of word-to-word translations does not have to represent a one-to-one mapping. Nearest neighbor approaches can be used, but they suffer from the so-called hubness problem: in high-dimensional vector spaces, certain vectors are universal nearest neighbors (Radovanovic et al., 2010), and this is a common problem for word-embedding-based bilingual lexicon induction (Dinu and Baroni, 2014). Lample et al. (2018) presented *cross-domain similarity local scaling* (CSLS), which is a method intended to reduce the influence of hubs by expanding high-density areas and condensing low-density ones.

Given a source vector $x_s$, the mean similarity of its transformation $Wx_s$ to its $k$ target nearest neighbors $\mathcal{N}_T^k(Wx_s)$ is defined as

$$\mu_T^k(Wx_s) = \frac{1}{k} \sum_{y_t \in \mathcal{N}_T^k(Wx_s)} \cos(Wx_s, y_t).$$

Likewise is defined $\mu_S^k(y_t)$, i.e., the mean similarity of a target word $y_t$ to its neighborhood of source mapped vectors. Then, the CSLS similarity between a mapped source vector $x_s$ and a target vector $y_t$ is calculated as follows: $\mathrm{CSLS}(Wx_s, y_t) = 2\cos(Wx_s, y_t) - \mu_T^k(Wx_s) - \mu_S^k(y_t)$. Intuitively, this mapping increases the similarity associated with isolated word vectors, and it decreases the one for vectors lying in dense areas. In the following experiments, we use the mapping induced by CSLS with $k = 10$.

## 5. Experiments

Below, we describe our experiments. In our first set of experiments, we deploy our method on top of well-known methods for cross-lingual alignment of embeddings and we show that it improves their accuracy, meaning that it can be used as a refinement tool. In the second set of experiments, we recreate the benchmarks from (Grave et al., 2019), and we show that our method can align word embedding spaces without a good initialization matrix.

### 5.1. The Iterative Hungarian Algorithm as a Refinement Tool

The experiments in this section use the Iterative Hungarian (IH) algorithm starting with the initial condition $W_0$ produced from the following methods:

- The adversarial approach by Lample et al. (2017). This combines the adversarial training described in Section 2 with a refinement step, which consists of creating a dictionary from the best matches and then running the supervised Procrustes algorithm using that dictionary.
- The supervised Procrustes approach.
- The Iterative Closest Point (ICP) method by Hoshen and Wolf (2018).

We used the word embeddings, the dictionaries and the evaluation methods from Lample et al. (2018). We trained the transformation matrix obtained from MUSE (Lample et al., 2018) on 200,000 words. Then we ran the Iterative Hungarian algorithm on a subsample of 45,000 words. Finally, we refined the new transformation matrix following the procedure in Lample et al. (2018). Also, inspired by their work, we induced mappings using CSLS with $k = 10$ nearest neighbors.

We ran the Iterative Hungarian algorithm after normalizing the word embeddings (divide them by their Euclidean norm), which we found to converge faster. It must be noted that, since the adversarial part does not normalize the word embeddings, the $W_0$ matrices do not match exactly and thus not normalizing them should yield better results at a higher computational cost. Hartmann et al. (2019) showed that unit-length normalization makes GAN-based methods more unstable and also deteriorates their performance, but supervised alignments or Procrustes refinement are not affected by this.

[1] http://github.com/cheind/py-lapsolver

| Method | en-es | es-en | en-fr | fr-en | en-it | it-en | en-de | de-en | en-ru | ru-en | mean |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MUSE (1) | **82.6** | 83.7 | 82.5 | 82.0 | 76.8 | 77.6 | **75.1** | 72.5 | 42.5 | 60.1 | 73.5 |
| MUSE (1) + IH | 82.5 | **84.1** | **82.7** | **82.4** | **78.3** | **77.9** | 74.9 | **73.3** | **44.5** | **60.7** | **74.1** |
| MUSE (2) | 81.9 | 83.2 | 82.1 | 82.4 | **77.5** | 77.5 | 74.7 | 72.9 | 37.0 | 61.9 | 73.1 |
| MUSE (2) + IH | **82.5** | **84.1** | **82.7** | 82.4 | 77.3 | **78.1** | 74.7 | **73.3** | **42.3** | **62.5** | **74.0** |
| MUSE (3) | 82.1 | **84.0** | 82.1 | 82.3 | **77.9** | 77.7 | 74.8 | 69.9 | 37.1 | 60.1 | 72.8 |
| MUSE (3) + IH | **82.3** | 83.9 | **82.6** | **82.4** | 77.8 | **77.8** | **75.1** | **72.9** | **38.9** | **62.1** | **73.6** |
| Procrustes | 81.7 | 83.3 | 82.1 | 81.9 | 77.3 | 77.0 | 73.7 | 72.7 | **49.9** | 60.8 | 74.0 |
| Procrustes + IH | **82.5** | **84.2** | **82.2** | **82.6** | **78.1** | **78.0** | **75.0** | **73.5** | 47.9 | **63.9** | **74.8** |
| ICP (1) | 81.9 | 82.7 | 81.9 | 81.5 | 76.0 | 75.5 | 72.3 | 72.3 | **46.4** | 56.6 | 72.7 |
| ICP (1) + IH | **82.5** | **84.1** | **82.1** | **82.7** | **78.1** | **78.0** | **76.6** | **72.7** | 46.2 | **63.2** | **74.6** |
| ICP (2) | 80.8 | 82.5 | 81.3 | 80.4 | 76.3 | 76.3 | 72.3 | 72.4 | 46.5 | 57.5 | 72.6 |
| ICP (2) + IH | **82.2** | **84.1** | **82.4** | **82.3** | **78.2** | **77.9** | **76.4** | **73.3** | **46.6** | **63.1** | **74.7** |
| ICP (3) | 82.0 | 82.6 | 82.0 | 81.8 | 75.7 | 76.6 | 73.1 | 72.6 | 45.1 | 56.2 | 72.8 |
| ICP (3) + IH | **82.5** | **84.2** | 82.0 | **82.4** | **77.7** | **77.7** | **76.9** | **73.5** | **45.2** | **63.1** | **74.5** |

Table 2: The Iterative Hungarian (IH) Algorithm starts with a transformation matrix $W$ from MUSE, Procrustes or ICP and then refines it. The numbers 1, 2 and 3 represent runs over different seeds for non-deterministic methods (MUSE and ICP).

The results can be seen in Table 2. We can see that our Iterative Hungarian algorithm improves the accuracy when used as a refinement tool. We believe that this is because the other methods do not try to optimize the Wasserstein-Procrustes objective directly, even though they achieve very good translations without relying on it. In the Appendix we report the performance of our algorithm on more language pairs.

We also tried Zhang et al. (2019)'s Iterative Normalization: before applying IH, we subtracted the mean of the word embeddings, and we normalized them. We repeated this process three times, and then we applied IH. The results appear in Table 3: although this method improved the initialization produced by MUSE, better results were obtained by simply normalizing the word embeddings (as shown in Table 2).

### 5.2. Aligning Word Embeddings from the Same Data

The second set of experiments justify that the simple iterative procedure displayed in Algorithm 2 works and we explain under what circumstances it can be relaxed or needs some help in the form of either supervision or a natural initialization matrix $W_0$. For the following controlled experiments, we set the initialization matrix to be the identity. We experiment with the following four approaches:

- *Hungarian.* Run the Hungarian algorithm for only one iteration, and then taking the permutation matrix $P$ as the map.
- *Cut Iterative Hungarian (CIH).* Run the Hungarian algorithm to update $Y \leftarrow PY$ and $X \leftarrow XW$ (see Algorithm 1).
- *Iterative Hungarian (IH).* Run the previous iterative procedure but considering the different natural initializations (see Algorithm 2).
- *Supervised Iterative Hungarian (SIH).* Learn the correct mapping from a random 5% subsample of the words, and then we run the IH algorithm for the remaining words.

The experiments from this subsection recreate those by Grave et al. (2019); the idea is that English word embeddings are trained after changing some parameters, and the different spaces of word embeddings are rotated in order to match. We use fastText (Bojanowski et al., 2017; Joulin et al., 2017) to train word embeddings on 100M English tokens from the 2007 News Crawl corpus.[2]

The different experiments in this section consist of changing the different training conditions and correctly mapping the results. We train the models using Skipgram (Mikolov et al., 2013c) unless stated otherwise, using the standard parameters of fastText.[3] We perform four experiments:

- **Seed.** We only change the seed used to generate the word embeddings in our fastText runs. The source and the target are word embeddings trained using the same parameters.

[2] http://statmt.org/wmt14/translation-task.html

[3] https://github.com/facebookresearch/fastText

| Method | en-es | es-en | en-fr | fr-en | en-it | it-en | en-ru | ru-en | mean |
|---|---|---|---|---|---|---|---|---|---|
| MUSE | 81.7 | 83.5 | **82.5** | 81.9 | 77.5 | 77.7 | **45.3** | 61.0 | 73.9 |
| MUSE + IH | **82.3** | **84.0** | 82.3 | **82.5** | **77.9** | **77.9** | 44.9 | **61.9** | **74.2** |

Table 3: The Iterative Hungarian (IH) Algorithm starts with a transformation matrix $W$ from MUSE, applies the iterative normalization from (Zhang et al., 2019) and then it refines the mapping.

| Method | Seed | Window | Algorithm | Data |
|---|---|---|---|---|
| Hungar. | **99%** | 7% | 7% | 1% |
| CIH | **100%** | **100%** | **100%** | 0% |
| IH | **100%** | **100%** | **100%** | 0% |
| SIH | **100%** | **100%** | **100%** | **100%** |

Table 4: Our method correctly aligns the word embeddings. *Hungar.* is short for *Hungarian*.

| Method | Seed | Window | Algorithm | Data |
|---|---|---|---|---|
| $\mathbb{I}$ | **9.49** | **12.59** | **12.45** | **14.11** |
| $V_X$ | 14.13 | 14.14 | 14.18 | 14.19 |
| $V_Y^\top$ | 14.15 | 14.18 | 14.18 | 14.14 |
| $V_X V_Y^\top$ | 13.95 | 14.10 | 14.09 | 14.16 |

Table 5: Distance between the natural initialization and the optimal solution for the four experiments.

- **Window.** We use window sizes of 2 and 10, respectively. The source and the target correspond to word embeddings trained on the same data but with different window sizes.

- **Algorithm.** We train the first algorithm with Skipgram and the second one with CBOW (Mikolov et al., 2013c). The source and the target correspond to word embeddings trained on the same data but using a different method.

- **Data.** We separate the dataset in two different parts of the same length. We train corresponding word embeddings from the two separate parts. The source and the target correspond to word embeddings trained with the same parameters but on different data.

We run the above algorithms on the 10,000 most frequent words. Table 4 shows the results for the different algorithms. We perform the final mapping using the nearest neighbor for CSLS with $k = 10$, and the reported score is the percentage of words correctly mapped. Notice, that since we are *translating English to English*, the correct map is trivial. Some observations follow:

- The supervised approach works well with very little supervision, but all other attempts failed when facing the problem of mapping data from different datasets. This is probably because, by adding some supervision, we improve the initial $W_0$. This effect may be similar (although with less impact) to the help introduced in the IH algorithm with the equivalent problems or the natural initial transformations.

- The first three experiments converged in three iterations or less. The SIH algorithm took around twenty iterations to converge for the *Data* experiment.

- The Hungarian algorithm, which was not designed for the Wasserstein-Procrustes method, correctly finds the mapping for the seed experiment, whereas some other reported iterative experiments failed to achieve good results in this experiment (Grave et al., 2019).

The proposed iterative procedures do converge, but they usually need good initial conditions or the help of supervision to converge to a good minimum. This suggests that Algorithm 1 could work well as long as we start from an initial transformation matrix $W_0$ close enough to the true solution. The importance of the initial condition can be shown by the natural initial conditions. The solution of the four different equivalent problems induce different optimal transformation matrices $W^*$. In the first iteration of the IH algorithm, a branch among these four is chosen. Table 5 shows the Euclidean distance between each of the four natural initializations (assuming $W_0 = \mathbb{I}$) and their respective optimal solution $W^*$ for the four experiments. These distances are different for the four branches, and to choose the best one (the one minimizing this distance) is key for convergence.

The distances that are too big do not converge to a good solution. For the *Seed* experiment, such a small distance explains why a single iteration of the Hungarian algorithm was enough for a strong result. The Window and the Algorithm do not converge when running on a branch different from the first one—also the one that has the smallest distance—and when they run on the first branch, they converge in a few iterations. Hence, being able to provide a good initial transformation matrix $W_0$ and to correctly discriminate what the best branches are is essential for this approach.

In the Appendix we present further experiments on English to Spanish that test whether our method can be used without a good initialization, but with little supervision. We found that our method works well when little supervision is given.

## 6. Conclusion and Future Work

We have underlined some mathematical properties of the Wasserstein-Procrustes problem and hence used the concept of the different natural initialization transformations in an iterative algorithm to achieve improved results for mapping word embeddings between different languages. In particular, we have shown that it is possible to use our algorithm as a refinement tool for UCAE and we have demonstrated improved results after using the transformation of Lample et al. (2018) as the initialization matrix $W_0$. We hope that our rethinking of the Wasserstein-Procrustes problem would enable further research and would eventually help develop better algorithms for aligning word embeddings across languages, especially taking into account that most unsupervised approaches try to minimize loss functions different from Objective 2.

In future work, we plan to study other loss functions. We are further interested to see how well the objectives in Table 1 correlate with CSLS. Finally, we plan combinations with other existing methods.

## 7. Limitations

While our work provides valuable insights and improvements for unsupervised cross-lingual alignment of embeddings, there are some limitations to consider:

- Our analysis primarily focuses on non-contextual unsupervised word embeddings. In future work, it is essential to extend this analysis to contextualized word embeddings, which are prevalent in modern NLP applications and offer additional challenges and opportunities for alignment.

- Our study is more theoretical in nature, and the Wasserstein-Procrustes problem may not always hold true in practice due to factors such as noisy datasets or significant differences among languages. Despite these potential discrepancies, we believe our unified framework can inspire future research for improving word embeddings and contribute to more effective algorithms in aligning them across languages.

Overall, these limitations highlight potential avenues for further research and emphasize the importance of continued exploration in the field of unsupervised cross-lingual alignment of embeddings.

## 8. Ethics Statement

As researchers in the field of natural language processing, we recognize the importance of addressing ethical considerations in our work. In this study, we focused on unsupervised cross-lingual alignment of embeddings, with the aim of improving alignment techniques and fostering further research in this area. Below, we outline some of the ethical aspects that we have considered in this research:

- **Fairness and Bias:** We are aware that word embeddings can unintentionally capture and propagate biases present in the training data. By improving alignment techniques across languages, our work could potentially contribute to the mitigation of biases and the promotion of fairness in multilingual applications. However, we also acknowledge that our methods could inadvertently introduce or amplify biases. Future work should include thorough assessments of potential biases in the embeddings and the development of strategies to address them.

- **Accessibility:** Our research aims to advance unsupervised cross-lingual alignment methods, which can contribute to the democratization of NLP technologies by enabling their application in low-resource languages with minimal data requirements.

- **Privacy:** As our work is based on unsupervised word embeddings pretrained on large text corpora, it is crucial to ensure that the underlying data does not contain sensitive or personally identifiable information. We have made efforts to use publicly available and well-vetted datasets for our experiments and evaluations, minimizing potential privacy concerns.

- **Impact:** The advancements in unsupervised cross-lingual alignment could lead to improved performance in various multilingual NLP tasks, such as machine translation, cross-lingual information retrieval, and sentiment analysis. While these improvements can have positive effects, it is essential to consider potential misuse of such technologies and remain vigilant against unintended consequences.

## Acknowledgements

This research was sponsored in part by the United States Air Force Research Laboratory and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. This material is based upon work supported in part by the U.S. Army Research

Office through the Institute for Soldier Nanotechnologies at MIT, under Collaborative Agreement Number W911NF-18-2-0048. The work was also supported by the Technical University of Catalonia (UPC), the CFIS program and Fundació Cellex. The views and the conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force, the U.S. Government, or the U.S. Army. The U.S. Government is authorized to reproduce and to distribute reprints for Government purposes notwithstanding any copyright notation herein.

## References

Jean Alaux, Edouard Grave, Marco Cuturi, and Armand Joulin. 2019. Unsupervised hyperalignment for multilingual word embeddings. In *Proceedings of the International Conference on Learning Representations*, ICLR '19, New Orleans, LA, USA.

Ghafour Alipour, Jamshid Bagherzadeh Mohasefi, and Mohammad-Reza Feizi-Derakhshi. 2022. Learning bilingual word embedding mappings with similar words in related languages using gan. *Applied Artificial Intelligence*, 36(1):2019885.

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-Wasserstein alignment of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1881–1890, Brussels, Belgium. Association for Computational Linguistics.

M. Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *AAAI*.

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*, pages 2289–2294, Austin, Texas. Association for Computational Linguistics.

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

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 the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019. Bilingual lexicon induction through unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, ACL '19, pages 5002–5007, Florence, Italy.

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 the 58th Annual Meeting of the Association for Computational Linguistics*, ACL '20, pages 7375–7388, Seattle, WA, USA.

Antonio Valerio Miceli Barone. 2016. Towards cross-lingual distributed representations without parallel text trained with adversarial autoencoders. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, RepL4NLP '16, pages 121–126, Berlin, Germany.

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

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26*, NIPS '13, pages 2292–2300.

Georgiana Dinu and Marco Baroni. 2014. Improving zero-shot learning by mitigating the hubness problem. In *Proceedings of the 3rd International Conference on Learning Representations (Workshop Track)*, ICLR '14, San Diego, CA, USA.

Jack Edmonds and Richard M. Karp. 1972. Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM*, 19(2):248–264.

Zihao Feng, Hailong Cao, Tiejun Zhao, Weixuan Wang, and Wei Peng. 2022. Cross-lingual feature extraction from monolingual corpora for low-resource unsupervised bilingual lexicon induction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5278–5287.

Marguerite Frank and Philip Wolfe. 1956. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110.

Masood Ghayoomi. 2023. Training vs post-training cross-lingual word embedding approaches: A comparative study. *International Journal of Information Science and Management (IJISM)*, 21(1):163–182.

Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. 2023. Dictionary-based phrase-level prompting of large language models for machine translation. *arXiv preprint arXiv:2302.07856*.

Steven Gold and Anand Rangarajan. 1996. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS '14, pages 2672–2680, Montreal, Canada.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2019. Unsupervised alignment of embeddings with Wasserstein Procrustes. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, AISTATS '2019, pages 1880–1890, Naha, Japan.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: HLT*, pages 771–779, Columbus, Ohio. Association for Computational Linguistics.

Mareike Hartmann, Yova Kementchedjhieva, and Anders Søgaard. 2019. Comparing unsupervised word translation methods step by step. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 6033–6043, Vancouver, BC, CA.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, LREC '18, Miyazaki, Japan.

Yedid Hoshen and Lior Wolf. 2018. Non-adversarial unsupervised word translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, EMNLP '18, pages 469–478, Brussels, Belgium.

Ann Irvine and Chris Callison-Burch. 2013. Supervised bilingual lexicon induction with multiple monolingual signals. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '13, pages 518–523, Atlanta, GA, USA.

Roy Jonker and A. Volgenant. 1987. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '17, pages 427–431, Valencia, Spain.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 7059–7069.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. In *Proceedings of the 6th International Conference on Learning Representations*, ICLR '18, Vancouver, BC, Canada.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *Proceedings of the 6th International Conference on Learning Representations*, ICLR '18, Vancouver, BC, Canada.

Ziheng Li, Shaohan Huang, Zihan Zhang, Zhi-Hong Deng, Qiang Lou, Haizhen Huang, Jian Jiao, Furu Wei, Weiwei Deng, and Qi Zhang. 2023. Dual-alignment pre-training for cross-lingual sentence embedding. *arXiv preprint arXiv:2305.09148*.

Davis Liang, Hila Gonen, Yuning Mao, Rui Hou, Naman Goyal, Marjan Ghazvininejad, Luke Zettlemoyer, and Madian Khabsa. 2023. Xlm-v: Overcoming the vocabulary bottleneck in multilingual masked language models. *arXiv preprint arXiv:2301.10472*.

Yuzhi Liu and Massimo Piccardi. 2023. Topic-based unsupervised and supervised dictionary induction. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(3):1–21.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations (Workshop Track)*, ICLR '13, Scottsdale, AZ, USA.

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

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013c. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, NIPS '13, page 3111–3119, Red Hook, NY, USA.

Facundo Mémoli. 2011. Gromov-Wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11:417–487.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP '14, pages 1532–1543, Doha, Qatar.

Milo Radovanovic, Alexandros Nanopoulos, and Mirjana Ivanovic;. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(86):2487–2531.

Sebastian Ruder, Ryan Cotterell, Yova Kementchedjhieva, and Anders Søgaard. 2018. A discriminative latent-variable model for bilingual lexicon induction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 458–468, Brussels, Belgium. Association for Computational Linguistics.

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

Peter H. Schönemann. 1966. A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31(1):1–10.

Richard Sinkhorn and Paul Knopp. 1967. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific J. Math.*, 21(2):343–348.

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 the 2017 International Conference on Learning Representations*, ICLR '17, Toulon, France.

Zhoujin Tian, Chaozhuo Li, Shuo Ren, Zhiqiang Zuo, Zengxuan Wen, Xinyue Hu, Xiao Han, Haizhen Huang, Denvy Deng, Qi Zhang, et al. 2022. Rapo: An adaptive ranking paradigm for bilingual lexicon induction. *arXiv preprint arXiv:2210.09926*.

Nobuaki Tomizawa. 1971. On some techniques useful for solution of transportation network problems. *Networks*, 1:173–194.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1934–1945, Copenhagen, Denmark. Association for Computational Linguistics.

Mozhi Zhang, Keyulu Xu, Ken-ichi Kawarabayashi, Stefanie Jegelka, and Jordan L. Boyd-Graber. 2019. Are girls neko or shōjo? cross-lingual alignment of non-isomorphic embeddings with iterative normalization. *CoRR*, abs/1906.01622.

Yuan Zhang, David Gaddy, Regina Barzilay, and Tommi Jaakkola. 2016. Ten pairs to tag – multilingual POS tagging via coarse mapping between embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1307–1317, San Diego, California. Association for Computational Linguistics.