Learning Unsupervised Word Translations Without Adversaries

Tanmoy Mukherjee[1] and **Makoto Yamada**[2] and **Timothy Hospedales**[1]
[1] The University of Edinburgh [2] Kyoto University/ RIKEN AIP/ JST PRESTO
{t.mukherjee@sms.,t.hospedales@}ed.ac.uk, myamada@i.kyoto-u.ac.jp

Abstract

Word translation, or bilingual dictionary induction, is an important capability that impacts many multilingual language processing tasks. Recent research has shown that word translation can be achieved in an unsupervised manner, without parallel seed dictionaries or aligned corpora. However, state of the art methods for unsupervised bilingual dictionary induction are based on generative adversarial models, and as such suffer from their well known problems of instability and hyper-parameter sensitivity. We present a statistical dependency-based approach to bilingual dictionary induction that is unsupervised – no seed dictionary or parallel corpora required; and introduces no adversary – therefore being much easier to train. Our method performs comparably to adversarial alternatives and outperforms prior non-adversarial methods.

1 Introduction

Translating words between languages, or more generally inferring bilingual dictionaries, is a long-studied research direction with applications including machine translation (Lample et al., 2017), multilingual word embeddings (Klementiev et al., 2012), and knowledge transfer to low resource languages (Guo et al., 2016). Research here has a long history under the guise of decipherment (Knight et al., 2006). Current contemporary methods have achieve effective word translation through theme-aligned corpora (Gouws et al., 2015), or seed dictionaries (Mikolov et al., 2013).

Mikolov et al. (2013) showed that monolingual word embeddings exhibit isomorphism across languages, and can be aligned with a simple linear transformation. Given two sets word vectors learned independently from monolingual corpora, and a dictionary of seed pairs to learn a linear transformation for alignment; they were able to estimate a complete bilingual lexicon. Many studies have since followed this approach, proposing various improvements such as orthogonal mappings (Artetxe et al., 2016) and improved objectives (Lazaridou et al., 2015).

Obtaining aligned corpora or bilingual seed dictionaries is nevertheless not straightforward for all language pairs. This has motivated a wave of very recent research into *unsupervised* word translation: inducing bilingual dictionaries given only monolingual word embeddings (Conneau et al., 2018; Zhang et al., 2017b,a; Artetxe et al., 2017). The most successful have leveraged ideas from Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). In this approach the generator provides the cross-modal mapping, taking embeddings of dictionary words in one language and 'generating' their translation in another. The discriminator tries to distinguish between this 'fake' set of translations and the true dictionary of embeddings in the target language. The two play a competitive game, and if the generator learns to fool the discriminator, then its cross-modal mapping should be capable of inducing a complete dictionary, as per Mikolov et al. (2013).

Despite these successes, such adversarial methods have a number of well-known drawbacks (Arjovsky et al., 2017): Due to the nature of their min-max game, adversarial training is very unstable, and they are prone to divergence. It is extremely hyper-parameter sensitive, requiring problem-specific tuning. Convergence is also hard to diagnose and does not correspond well to efficacy of the generator in downstream tasks (Hoshen and Wolf, 2018).

In this paper, we propose an alternative statistical dependency-based approach to unsupervised word translation. Specifically, we propose to search for the cross-lingual word pairing that maximizes statistical dependency in terms of squared

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 627–632
Brussels, Belgium, October 31 - November 4, 2018.

loss mutual information (SMI) (Yamada et al., 2015; Suzuki and Sugiyama, 2010). Compared to prior statistical dependency-based approaches such as Kernelized Sorting (KS) (Quadrianto et al., 2009) we advance: (i) through use of SMI rather than their Hilbert Schmidt Independence Criterion (HSIC) and (ii) through jointly optimising cross-modal pairing with representation learning within each view. In contrast to prior work that uses a fixed representation, by non-linearly projecting monolingual world vectors before matching, we learn a new embedding where statistical dependency is easier to establish. Our method: (i) achieves similar unsupervised translation performance to recent adversarial methods, while being significantly easier to train and (ii) clearly outperforms prior non-adversarial methods.

2 Proposed model

2.1 Deep Distribution Matching

Let dataset $\mathcal{D}$ contain two sets of unpaired monolingual word embeddings from two languages $\mathcal{D} = (\{\boldsymbol{x}_i\}_{i=1}^n, \{\boldsymbol{y}_j\}_{j=1}^n)$ where $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$. Let π be a permutation function over $\{1, 2, \ldots, n\}$, and $\boldsymbol{\Pi}$ the corresponding permutation indicator matrix: $\boldsymbol{\Pi} \in \{0,1\}^{n \times n}, \boldsymbol{\Pi}\mathbf{1}_n = \mathbf{1}_n$, and $\boldsymbol{\Pi}^\top \mathbf{1}_n = \mathbf{1}_n$. Where $\mathbf{1}_n$ is the n-dimensional vector with all ones. We aim to optimize for both the permutation $\boldsymbol{\Pi}$ (bilingual dictionary), and non-linear transformations $\boldsymbol{g_x}(\cdot)$ and $\boldsymbol{g_y}(\cdot)$ of the respective wordvectors, that maximize statistical dependency between the views. While regularising by requiring the original word embedding information is preserved through reconstruction using decoders $\boldsymbol{f_x}(\cdot)$ and $\boldsymbol{f_y}(\cdot)$. Our overall loss function is:

$$\min_{\boldsymbol{\Theta}_x, \boldsymbol{\Theta}_y, \boldsymbol{\Pi}} \underbrace{\Omega(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y})}_{Regularizer} - \underbrace{\lambda D_{\boldsymbol{\Pi}}(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y})}_{Dependency},$$

$$D_{\boldsymbol{\Pi}}(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}) = D_{\boldsymbol{\Pi}}(\{\boldsymbol{g_x}(\boldsymbol{x}_i), \boldsymbol{g_y}(\boldsymbol{y}_{\pi(i)})\}_{i=1}^n),$$

$$\begin{aligned}\Omega(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}) = \sum_{i=1}^n & \|\boldsymbol{x}_i - \boldsymbol{f_x}(\boldsymbol{g_x}(\boldsymbol{x}_i))\|_2^2 \\ & + \|\boldsymbol{y}_i - \boldsymbol{f_y}(\boldsymbol{g_y}(\boldsymbol{y}_i))\|_2^2 \\ & + R(\boldsymbol{\Theta_x}) + R(\boldsymbol{\Theta_y}).\end{aligned} \tag{1}$$

where $\boldsymbol{\Theta}$s parameterize the encoding and reconstruction transformations, $R(\cdot)$ is a regularizer (e.g., ℓ_2-norm and ℓ_1-norm), and $D_{\boldsymbol{\Pi}}(\cdot, \cdot)$ is a statistical dependency measure. Crucially compared to prior methods such as matching CCA (Haghighi et al., 2008), dependency measures such as SMI do not need comparable representations to get started, making the bootstrapping problem less severe.

2.2 Dependence Estimation

Squared-Loss Mutual Information (SMI) The squared loss mutual information between two random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as (Suzuki and Sugiyama, 2010):

$$\text{SMI} = \iint \left(\frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} - 1\right)^2 p(\boldsymbol{x})p(\boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y},$$

which is the Pearson divergence (Pearson, 1900) from $p(\boldsymbol{x}, \boldsymbol{y})$ to $p(\boldsymbol{x})p(\boldsymbol{y})$. The SMI is an f-divergence (Ali and Silvey, 1966). That is, it is a non-negative measure and is zero only if the random variables are independent.

To measure SMI from a set of samples we take a direct density ratio estimation approach (Suzuki and Sugiyama, 2010), which leads (Yamada et al., 2015) to the estimator:

$$\widehat{\text{SMI}}(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) = \frac{1}{2n} \text{tr}\left(\text{diag}\left(\widehat{\boldsymbol{\alpha}}\right) \boldsymbol{K}\boldsymbol{L}\right) - \frac{1}{2},$$

where $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{L} \in \mathbb{R}^{n \times n}$ are the gram matricies for $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively, and

$$\widehat{\boldsymbol{H}} = \frac{1}{n^2}(\boldsymbol{K}\boldsymbol{K}^\top) \circ (\boldsymbol{L}\boldsymbol{L}^\top),$$
$$\widehat{\boldsymbol{h}} = \frac{1}{n}(\boldsymbol{K} \circ \boldsymbol{L})\mathbf{1}_n, \;\; \widehat{\boldsymbol{\alpha}} = \left(\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_n\right)^{-1} \widehat{\boldsymbol{h}},$$

$\lambda > 0$ is a regularizer and $\boldsymbol{I}_n \in \mathbb{R}^{n \times n}$ is the identity matrix.

SMI for Matching SMI computes the dependency between two sets of variables, under an assumption of known correspondence. In our application this corresponds to a measure of dependency between two *aligned* sets of monolingual wordvectors. To exploit SMI for matching, we introduce a permutation variable $\boldsymbol{\Pi}$ by replacing $\boldsymbol{L} \rightarrow \boldsymbol{\Pi}^\top \boldsymbol{L}\boldsymbol{\Pi}$ in the estimator:

$$\widehat{\text{SMI}}(\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_1^n) = \frac{1}{2n} \text{tr}\left(\text{diag}\left(\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Pi}}\right) \boldsymbol{K}\boldsymbol{\Pi}^\top \boldsymbol{L}\boldsymbol{\Pi}\right) - \frac{1}{2},$$

that will enable optimizing $\boldsymbol{\Pi}$ to maximize SMI.

2.3 Optimization of parameters

To initialize $\boldsymbol{\Theta_x}$ and $\boldsymbol{\Theta_y}$, we first independently estimate them using autoencoders. Then we employ an alternative optimization on Eq. (1) for

$(\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y})$ and $\boldsymbol{\Pi}$ until convergence. We use 3 layer MLP neural networks for both $\boldsymbol{f}$ and $\boldsymbol{g}$. Algorithm 1 summarises the steps.

Optimization for $\boldsymbol{\Theta_x}$ and $\boldsymbol{\Theta_y}$ With fixed permutation matrix $\boldsymbol{\Pi}$ (or π), the objective function

$$\min_{\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}} \Omega(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}) - \lambda D_{\boldsymbol{\Pi}}(\mathcal{D}; \boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}) \quad (2)$$

is an autoencoder optimization with regularizer $D_{\boldsymbol{\Pi}}(\cdot)$, and can be solved with backpropagation.

Optimization for $\boldsymbol{\Pi}$ To find the permutation (word matching) $\boldsymbol{\Pi}$ that maximizes SMI given fixed encoding parameters $\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}$, we only need to optimize the dependency term $D_{\boldsymbol{\Pi}}$ in Eq. (1). We employ the LSOM algorithm (Yamada et al., 2015). The estimator of SMI for samples $\{\boldsymbol{g_x}(\boldsymbol{x}_i), \boldsymbol{g_y}(\boldsymbol{y}_{\pi(i)})\}_{i=1}^n$ encoded with $\boldsymbol{g}_x, \boldsymbol{g}_y$ is:

$$\widehat{\mathrm{SMI}} = \frac{1}{2n} \mathrm{tr}\left(\mathrm{diag}\left(\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Theta},\boldsymbol{\Pi}}\right) \boldsymbol{K}_{\boldsymbol{\Theta_x}} \boldsymbol{\Pi}^\top \boldsymbol{L}_{\boldsymbol{\Theta_y}} \boldsymbol{\Pi}\right) - \frac{1}{2}.$$

Which leads to the optimization problem:

$$\max_{\boldsymbol{\Pi} \in \{0,1\}^{n \times n}} \quad \mathrm{tr}\left(\mathrm{diag}\left(\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Theta},\boldsymbol{\Pi}}\right) \boldsymbol{K}_{\boldsymbol{\Theta_x}} \boldsymbol{\Pi}^\top \boldsymbol{L}_{\boldsymbol{\Theta_y}} \boldsymbol{\Pi}\right)$$
$$\text{s.t.} \quad \boldsymbol{\Pi} \mathbf{1}_n = \mathbf{1}_n, \boldsymbol{\Pi}^\top \mathbf{1}_n = \mathbf{1}_n. \quad (3)$$

Since the optimization problem is NP-hard, we iteratively solve the relaxed problem (Yamada et al., 2015):

$$\boldsymbol{\Pi}^{\text{new}} = (1-\eta)\boldsymbol{\Pi}^{\text{old}} +$$
$$\eta \operatorname*{argmax}_{\boldsymbol{\Pi}} \mathrm{tr}\left(\mathrm{diag}\left(\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Theta},\boldsymbol{\Pi}^{\text{old}}}\right) \boldsymbol{K}_{\boldsymbol{\Theta_x}} \boldsymbol{\Pi}^\top \boldsymbol{L}_{\boldsymbol{\Theta_y}} \boldsymbol{\Pi}^{\text{old}}\right),$$

where $0 < \eta \leq 1$ is a step size. The optimization problem is a *linear assignment problem* (LAP). Thus, we can efficiently solve the algorithm by using the *Hungarian method* (Kuhn, 1955). To get discrete $\boldsymbol{\Pi}$, we solve the last step by setting $\eta = 1$.

Intuitively, this can be seen as searching for the permutation $\boldsymbol{\Pi}$ for which the data in the two (initially unsorted views) have a matching within-view affinity (gram) matrix, where matching is defined by maximum SMI.

3 Experiments

In this section, we evaluate the efficacy of our proposed method against various state of the art methods for word translation.

Implementation Details Our autoencoder consists of two layers with dropout and a $tanh$ nonlinearity. We use polynomial kernel to compute the gram matrices $\boldsymbol{K}$ and $\boldsymbol{L}$. For all pairs of languages, we fix the number of training epochs to 20. All the word vectors are ℓ_2 unit normalized. For CSLS we set the number of neighbors to 10. For optimizing $\boldsymbol{\Pi}$ at each epoch, we set the step size $\eta = 0.75$ and use 20 iterations. For the regularization $R(\boldsymbol{\Theta})$, we use the sum of the Frobenius norms of weight matrices. We train $\boldsymbol{\Theta}$ using full batch gradient-descent, with learning rate 0.05.

Algorithm 1 SMI-based unsupervised word translation

Input: Unpaired word embeddings $\mathcal{D} = (\{\boldsymbol{x}_i\}_{i=1}^n, \{\boldsymbol{y}_j\}_{j=1}^n)$.

1: **Init**: weights $\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}$, permutation matrix $\boldsymbol{\Pi}$.
2: **while** not converged **do**
3: Update $\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}$ given $\boldsymbol{\Pi}$: Backprop (2).
4: Update $\boldsymbol{\Pi}$ given $\boldsymbol{\Theta_x}, \boldsymbol{\Theta_y}$: LSOM (3).
5: **end while**

Output: Permutation Matrix $\boldsymbol{\Pi}$. Params $\boldsymbol{\Theta_x}$, $\boldsymbol{\Theta_y}$.

Datasets We performed experiments on the publicly available English-Italian, English-Spanish and English-Chinese datasets released by (Dinu and Baroni, 2015; Zhang et al., 2017b; Vulic and Moens, 2013). We name this collective set of benchmarks BLI. We also conduct further experiments on a much larger recent public benchmark, MUSE (Conneau et al., 2018)[1].

Setting and Metrics We evaluate all methods in terms of Precision@1, following standard practice. We note that while various methods in the literature were initially presented as fully supervised (Mikolov et al., 2013), semi-supervised (using a seed dictionary) (Haghighi et al., 2008), or unsupervised (Zhang et al., 2017b), most of them can be straightforwardly adapted to run in any of these settings. Therefore we evaluate all methods both in the unsupervised setting in which we are primarily interested, and also the commonly evaluated semi-supervised setting with 500 seed pairs.

Competitors: Non-Adversarial In terms of competitors that, like us, do not make use of GANs, we evaluate: **Translation Matrix** (Mikolov et al., 2013), which alternates between estimating a linear transformation by least squares and matching by nearest neighbour (NN). **Multilingual Correlation** (Faruqui and Dyer, 2014), and **Matching CCA** (Haghighi et al., 2008), which alternates between matching and estimat-

[1] https://github.com/facebookresearch/MUSE/

	MUSE Dataset						BLI Datasets					
Methods	es-en	en-es	it-en	en-it	zh-en	en-zh	es-en	en-es	it-en	en-it	zh-en	en-zh
TM (Mikolov et al., 2013)	5.6	4.8	5.2	4.8	2.6	1.8	3.2	2.9	4.6	4.2	3.2	2.0
CCA (Faruqui and Dyer, 2014)	6.1	5.6	5.8	5.2	3.1	2.3	5.3	5.0	4.6	4.1	3.2	2.9
MCCA (Haghighi et al., 2008)	5.7	5.1	5.4	4.8	3.0	2.2	2.9	2.5	4.2	4.1	2.8	1.9
KS (Quadrianto et al., 2009)	8.3	7.4	6.3	5.7	4.8	3.2	9.6	8.9	8.2	7.3	3.7	3.5
Self-Training (Artetxe et al., 2017)	12.4	12.2	10.7	10.2	5.8	5.6	15.8	14.5	13.7	12.7	14.8	13.4
EMDOT (Zhang et al., 2017b)	72.4	71.8	72.8	72.6	32.8	31.7	29.3	31.2	25.6	28.4	24.2	27.8
W-GAN (Zhang et al., 2017b)	78.2	77.4	75.3	74.8	38.6	37.5	23.4	26.7	24.0	25.3	21.2	22.8
GAN-NN (Conneau et al., 2018)	69.8	71.3	72.1	71.5	41.3	40.2	21.4	24.3	22.7	23.2	21.3	21.8
Deep-SMI (Ours)	75.9	80.6	75.7	75.2	38.5	38.1	27.3	28.2	25.7	26.4	22.5	22.3
Deep-SMI-CSLS	79.2	84.5	78.8	78.5	43.7	42.8	28.6	29.3	26.7	28.2	23.2	24.7

Table 1: Unsupervised word translation on MUSE and BLI datasets. Precision @ 1 metric. Top group: Conventional methods. Middle group: Adversarial methods. Bottom group: Our methods. Language codes zh=Chinese,en=English,es=Spanish,it=Italian

	MUSE Dataset						BLI Datasets					
Methods	es-en	en-es	it-en	en-it	zh-en	en-zh	es-en	en-es	it-en	en-it	zh-en	en-zh
TM (Mikolov et al., 2013)	32.6	30.1	34.3	33.6	32.4	31.2	28.2	32.1	29.2	32.1	28.5	27.4
CCA (Faruqui and Dyer, 2014)	27.3	27.1	25.4	24.2	23.1	20.2	25.8	28.3	24.3	25.1	19.2	22.8
MCCA (Haghighi et al., 2008)	26.3	25.8	22.7	21.3	24.5	23.8	24.2	26.1	17.6	19.2	18.4	21.6
KS (Quadrianto et al., 2009)	34.5	32.6	35.2	33.8	34.3	33.2	27.5	29.1	34.3	32.1	20.0	23.2
Self-Training (Artetxe et al., 2017)	35.8	31.4	36.0	34.6	34.3	33.0	27.8	29.8	39.7	33.8	23.6	21.4
EMDOT (Zhang et al., 2017b)	78.2	76.3	75.0	74.6	33.2	32.0	30.2	28.4	31.7	30.3	29.3	28.7
W-GAN (Zhang et al., 2017b)	81.2	80.5	77.2	75.1	39.0	38.2	28.6	27.9	33.7	29.5	36.7	34.4
GAN-NN (Conneau et al., 2018)	74.8	72.3	74.3	72.5	43.2	42.7	22.8	26.1	27.9	27.1	24.2	23.6
Deep-SMI (Ours)	80.6	75.9	78.2	76.7	45.7	44.6	38.5	37.6	42.3	38.2	29.2	27.4
Deep-SMI-CSLS	84.5	79.2	79.7	78.7	42.3	44.4	28.6	29.3	26.7	28.2	23.2	24.7

Table 2: Semi-supervised word translation on MUSE and BLI using 500 seed pair initial dictionary. Precision @ 1 metric. Top group: Conventional methods. Middle group: Adversarial methods. Bottom group: Our methods.

ing a joint linear subspace. **Kernelized Sorting** (Quadrianto et al., 2009), which directly uses HSIC-based statistical dependency to match heterogeneous data points. **Self Training** (Artetxe et al., 2017) A recent state of the art method that alternate between estimating an orthonormal transformation, and NN matching.

Competitors: Adversarial In terms of competitors that do make use of adversarial training, we compare: **W-GAN** and **EMDOT** (Zhang et al., 2017b) make use of adversarial learning using Wasserstein GAN and Earth Movers Distance respectively. **GAN-NN** (Conneau et al., 2018) uses adversarial learning to train an orthogonal transformation, along with some refinement steps and an improvement to the conventional NN matching procedure called 'cross-domain similarity local scaling' (CSLS). Since this is a distinct step, we also evaluate our method with CSLS.

We use the provided code for GAN-NN and Self-Train, while re-implementing EDOT/W-GAN to avoid dependency on theano.

3.1 Results

Fully Unsupervised Table 1 presents comparative results for unsupervised word translation on BLI and MUSE. From these we observe: (i) Our method (bottom) is consistently and significantly better than non-adversarial alternatives (top). (ii) Compared to adversarial alternatives Deep-SMI performs comparably.

All methods generally perform better on the MUSE dataset than BLI. These differences are due to a few factors: MUSE is a significantly

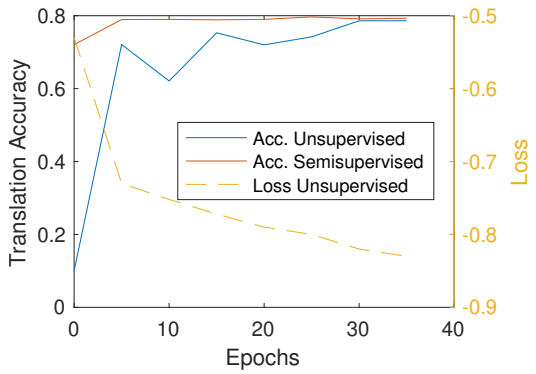

Figure 1: Training process of Deep-SMI

larger dataset than BLI, benefitting methods that can exploit a large amount of training data. In the ground-truth annotation, BLI contains 1-1 translations while MUSE contains more realistic 1-many translations (if any correct translation is picked, a success is counted), making it easier to reach a higher score.

Semi-supervised Results using a 500-word bilingual seed dictionary are presented in Table 2. From these we observe: (i) The conventional methods' performances (top) jump up, showing that they are more competitive if at least some sparse data is available. (ii) Deep-SMI performance also improves, and still outperforms the classic methods significantly overall. (iii) Again, we perform comparably to the GAN methods.

3.2 Discussion

Figure 1 shows the convergence process of Deep-SMI. From this we see that: (i) Unlike the adversarial methods, our objective (Eq. (1)) improves smoothly over time, making convergence much easier to assess. (ii) Unlike the adversarial methods, our accuracy generally mirrors the model's loss. In contrast, the various losses of the adversarial approaches do not well reflect translation accuracy, making model selection or early stopping a challenge in itself. Please compare our Figure 1 with Fig 3 in Zhang et al. (2017b), and Fig 2 in Conneau et al. (2018).

There are two steps in our optimization: matching permutation $\mathbf{\Pi}$ and representation weights $\mathbf{\Theta}$. Although this is an alternating optimization, it is analogous to an EM-type algorithm optimizing latent variables ($\mathbf{\Pi}$) and parameters ($\mathbf{\Theta}$). While local minima are a risk, every optimisation step for either variable reduces our objective Eq. (1). There is no min-max game, so no risk of divergence as in the case of adversarial GAN-type methods.

Our method can also be understood as providing an unsupervised *Deep*-CCA type model for relating heterogeneous data across two views. This is in contrast to the recently proposed unsupervised shallow CCA (Hoshen and Wolf, 2018), and conventional supervised Deep-CCA (Chang et al., 2018) that requires paired data for training; and using SMI rather than correlation as the optimisation objective.

4 Conclusion

We have presented an effective approach to unsupervised word translation that performs comparably to adversarial approaches while being significantly easier to train and diagnose; as well as outperforming prior non-adversarial approaches.

References

Syed M. Ali and Samuel. D. Silvey. 1966. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society. Series B (Methodological)*, 28(1):131–142.

Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *ICML*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *EMNLP*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *ACL*.

Xiaobin Chang, Tao Xiang, and Timothy M. Hospedales. 2018. Scalable and effective deep CCA via soft decorrelation. In *CVPR*.

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

Georgiana Dinu and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem. *ICLR Workshops*.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *EACL*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *ICML*.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2016. A distributed representation-based framework for cross-lingual transfer parsing. *JAIR*, 55(1):995–1023.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *ACL*.

Yedid Hoshen and Lior Wolf. 2018. Unsupervised correlation analysis. In *CVPR*.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *COLING*.

K. Knight, A. Nair, N. Rathod, and K. Yamada. 2006. Unsupervised analysis for decipherment problems. In *Proc. ACL-COLING*.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. 2015. Hubness and pollution: Delving into cross-space mapping for zero-shot learning. In *ACL*.

Tomas Mikolov, Google Inc, Mountain View, Quoc V. Le, Google Inc, Ilya Sutskever, and Google Inc. 2013. Exploiting similarities among languages for machine translation.

Karl Pearson. 1900. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175.

Novi Quadrianto, Le Song, and Alex J Smola. 2009. Kernelized sorting. In *NIPS*.

Taiji Suzuki and Masashi Sugiyama. 2010. Sufficient dimension reduction via squared-loss mutual information estimation. In *AISTATS*.

Ivan Vulic and Marie-Francine Moens. 2013. Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In *HLT-NAACL*.

Makoto Yamada, Leonid Sigal, Michalis Raptis, Machiko Toyoda, Yi Chang, and Masashi Sugiyama. 2015. Cross-domain matching with squared-loss mutual information. *IEEE TPAMI*, 37(9):1764–1776.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017a. Adversarial training for unsupervised bilingual lexicon induction. In *ACL*.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017b. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *EMNLP*.