Embedding-Enhanced GIZA++: Improving Low-Resource Word Alignment Using Embeddings

Anonymous EMNLP submission

Abstract

Word alignment has been dominated until recently by GIZA++, a statistical method based on the 30-year-old IBM models. New methods primarily rely on large machine translation models, massively multilingual language models, or supervision. We introduce Embedding-Enhanced GIZA++, and outperform GIZA++ without any of the aforementioned factors. Taking advantage of monolingual embedding spaces of source and target language only, we exceed GIZA++'s performance in every tested scenario for three languages pairs. In the lowest-resource setting, we outperform GIZA++ by 8.5, 10.9, and 12 AER for Ro-En, De-En, and En-Fr, respectively. We release our code at `www.blind-review.code`.

1 Introduction

Once ubiquitous, word alignment is no longer a step in typical machine translation (MT) using neural models, but is still important for low-resource and unsupervised MT methods (e.g. Lample et al., 2018; Artetxe et al., 2019) that use statistical MT because it can be trained using less data (Koehn et al., 2003; Koehn and Knowles, 2017; Sennrich and Zhang, 2019). Alignments are also useful for annotation transfer (e.g. Yarowsky and Ngai, 2001; Rasooli et al., 2018) and as a post-processing step to reinsert markup (e.g. Müller, 2017).

GIZA++ (Och, 2003), a statistical alignment model, has been the most commonly used tool for word alignment quality for 20 years and is based the IBM translation models that are yet a decade older (Brown et al., 1993). Though a handful of neural systems have outperformed GIZA++, these rely on large MT models (e.g. Chen et al., 2020; Zenkel et al., 2020; Stengel-Eskin et al., 2019), massively multilingual language models (e.g. Sabet et al., 2020; Dou and Neubig, 2021; Garg et al., 2019b), supervision from human-annotated alignments (Nagata et al., 2020), or combinations of the above. Though successful on the large high-resource data sets on which they are trained and tested, NMT models notoriously require large amounts of bitext for adequate performance.

We introduce Embedding-Enhanced GIZA++ (EE-GIZA++), an improvement to GIZA++ without any of the aforementioned factors. EE-GIZA++ biases GIZA++ to align semantically similar words from a shared embedding space. We outperform GIZA++ in all tested settings on three languages pairs. EE-GIZA++ is particularly well-suited for very low-resource scenarios; using only ∼500 lines of bitext, it outperforms GIZA++ by 10.9 AER and 12.0 AER for De-En and Fr-En, respectively.

2 Related Work

Recent work involves using neural translation models to guide or extract alignments, viewing attention as a proxy for alignment (e.g. Peter et al., 2017; Li et al., 2018; Garg et al., 2019b; Zenkel et al., 2019, 2020; Chen et al., 2020). Because neural models are notoriously data-hungry, they often fail in low-resource settings (our focus).

Other aligners use massive multilingual language models with contextualized embeddings such as mBERT (Devlin et al., 2019). Reminiscent of our approach, Dou and Neubig (2021) calculate a probability distribution over possible alignments from a finetuned mBERT embedding space and extract alignments using optimal transport. Like us, Sabet et al. (2020) experiment with mapped monolingual embedding spaces, but exceed the GIZA++ baseline only when using spaces such as mBERT and XLM-R (Conneau et al., 2020). Nagata et al. (2020) use mBERT and require supervision with human-annotated alignments.

Like us, Pourdamghani et al. (2018) improve low-resource alignment with word vectors. Jalili Sabet et al. (2016) also use nearest-neighbors in a word embedding space to alter IBM Model 1, but their performance does not match ours.

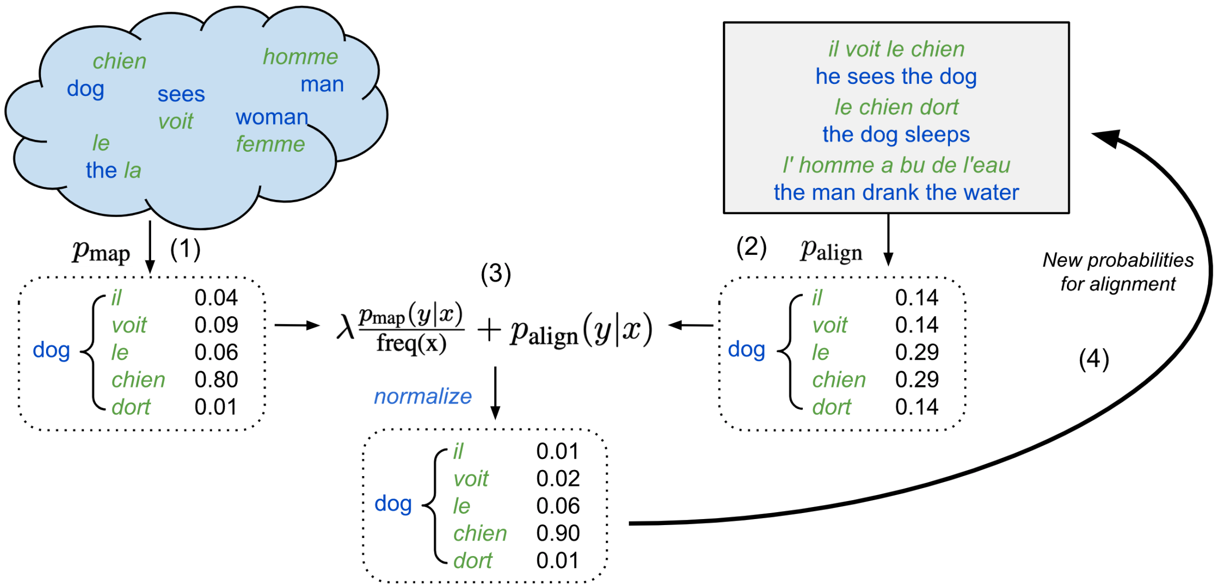

Figure 1: Proposed Method: Embedding-Enhanced GIZA++. 1) Map monolingual embeddings to crosslingual space. Calculate CSLS for cooccurring words and take softmax to calculate a probability distribution (*p_map*). 2) Use statistical aligner to calculate separate probability distribution over cooccuring words (*p_align*). 3) Interpolate the distributions with weight proportional to source word's frequency. Normalize. 4) Replace the statistical model's translation probability table with updated probability distribution. 5) Repeat Steps 2-4 for each iteration of EM.

3 Background

Let S be a source-language sentence of tokens $(s_1, s_2, ..., s_m)$ and T be a target-language sentence $(t_1, t_2, ..., t_l)$. Alignments are defined as $A \subseteq \{(s,t) \in S \times T\}$ where each s, t are meaningfully related—usually, translations of one another. Performance is typically measured with Alignment Error Rate (AER) (Och and Ney, 2000a).

3.1 GIZA++

GIZA++ is a popular statistical alignment and MT toolkit (Och and Ney, 2000b, 2003) which implements IBM Models 1-5 (Brown et al., 1993) and the HMM Model (Vogel et al., 1996), trained using expectation-maximization (EM). The default training setup is to run five iterations each of IBM Model 1, HMM, Model 3, and Model 4. GIZA++ is highly effective at aligning frequent words in a corpus, but error-prone for infrequent words.

IBM Models The IBM models developed more than 30 years ago for MT are useful for alignment. IBM Model 1 relies on lexical translation probabilities $p(f|e)$ for source word e and target word f. Model 2 adds an alignment model $p(j \mid i, l, m)$, predicting source position j from target position i of sentences with lengths m and l, respectively. Model 3 adds a fertility model. Model 4 and the HMM Model replace the alignment with a relative reordering model. After training, the most likely alignment can be computed for a sentence pair.

3.2 Monolingual Embedding Space Mapping

Non-contextual vector representations of words ("word embeddings", "word vectors") are ubiquitous in modern NLP (e.g. Mikolov et al., 2013; Bojanowski et al., 2017). Word vectors trained on monolingual data alone *embed* the word into an N-dimensional monolingual embedding space, where distance and angle have meaning. Mapping monolingual embedding spaces to a shared crosslingual space is common, particularly for bilingual lexicon induction and cross-lingual information retrieval.

Procrustes Problem Techniques that map monolingual embedding spaces to a crosslingual space typically solve a variation of the generalized Procrustes problem (e.g., Artetxe et al., 2018b; Conneau et al., 2018; Patra et al., 2019; Ramírez et al., 2020). Given word embedding matrices $X, Y \in \mathbb{R}^{n \times d}$ where $x \in X$, $y \in Y$ are word vectors in source and target languages, the goal is to find the map $W \in \mathbb{R}^{d \times d}$ that minimizes distances for each pair (x, y) known to be translations:

$$\arg\min_{W} \|XW - Y\|_F$$

When restricting W to be orthogonal ($WW^T = I$), Schönemann (1966) showed that the closed-form solution is $W = VU^T$, where $U\Sigma V$ is the singular value decomposition of $Y^T X$.

After mapping X and Y to a shared space with W, translations are extracted via nearest-neighbor search. A popular distance metric is cross-domain similarity local scaling (CSLS) to mitigate the "hubness problem" (Conneau et al., 2018).

4 Method

GIZA++ is highly effective at inducing the correct alignment for frequent words when parallel resources are abundant, but is error-prone for rare words. Because word embeddings can be trained on large amounts of monolingual data, rare words from a parallel corpus may be well-enough represented in a large monolingual corpus that reasonable word embeddings can be trained. Our key insight is that for infrequent words, finding a translation via nearest-neighbors in a shared embedding space may be more reliable than using a statistical aligner. We thus incorporate embedding space mapping into GIZA++ training, giving more or less influence to the statistical aligner depending on word frequency. Figure 1 shows the method.

1. Map embedding spaces. Word embedding spaces X and Y for source and target language, respectively, are mapped to a crosslingual space using VecMap.

2. Calculate translation probability distribution from mapped spaces. Let $\mathrm{Co}_Y(x)$ be the words from the target language that cooccur with source word x in the corpus. For each x, we calculate a probability distribution over possible alignments from $\mathrm{Co}_Y(x)$ with a softmax over the CSLS scores.[1] We use the mapped embedding spaces for source and target languages for CSLS.

$$p_{map}(y|x) = \frac{\exp\left(\mathrm{CSLS}(x, y)/\tau\right)}{\sum_{y' \in \mathrm{Co}_Y(x)} \exp\left(\mathrm{CSLS}(x, y')\right)/\tau)}$$

3. Integrate with GIZA++. Recall that IBM Models 1, 3, 4, and HMM maintain a lexical translation table of $p_{\text{align}}(y|x)$ for every cooccurring source-target word pair.

During training of IBM Model 1 and the HMM Model, we interpolate the lexical translation table with embedding-based translation probabilities after each iteration of EM. For each cooccurring pair (x, y), calculate:

$$score(x, y) = \lambda \frac{p_{\text{map}}(y|x)}{\text{freq(x)}} + p_{\text{align}}(y|x)$$

where freq(x) is the raw frequency of x in the source-side of the corpus and λ is a hyperparameter. The effect of this is that p_{map} is given more weight for infrequent words, in accordance with our goal to trust the embedding space mapper for infrequent words and the statistical aligner for frequent words. We then normalize over cooccuring words:

$$p(y|x) = \frac{score(x, y)}{\sum_{y_i \in \mathrm{Co}_Y(x)} score(x, y_i)} \quad (1)$$

We update GIZA++'s lexical translation table with the new value from Equation 1 for all cooccurring pairs, then begin the next iteration of EM.[2] This process is repeated for all iterations of IBM Model 1 and HMM model training. IBM Model 3 and 4 are trained as usual. Integrating probabilites from p_{map} into IBM Models 3 and 4 is for future work.

Steps 1-3 are done in source→target and target→source directions. Alignments are symmetrized with grow-diag-final (Koehn et al., 2003).

5 Experimental Setup

We use the same training setup as previous work[3] (Garg et al., 2019b; Zenkel et al., 2019, 2020; Chen et al., 2020; Dou and Neubig, 2021). Training corpora for German-English (De-En), English-French (En-Fr), and Romanian-English (Ro-En) are 1.9M, 1.1M, and 448K lines, respectively. Test sets are 508, 447, and 248 lines, respectively. Validation sets do not exist, so we tune λ on a 1M-line subset of De-En.[4] λ is set to 10,000. We use the VecMap[5] (Artetxe et al., 2018a) implementation of CSLS and SciPy for some utility functions and softmax calculation (Virtanen et al., 2020; Harris et al., 2020). For pretrained monolingual word embedding spaces, we use the publicly-available Wikipedia word vectors trained using fastText from (Bojanowski et al., 2017)[6]. We limit vocabulary to 200,000. Embedding mapping is done with VecMap (unsupervised).

[1]We use $\tau = 0.1$.

[2]If a word from the bitext is not present in the word embedding space, its translation probability is not updated.

[3]github.com/lilt/alignment-scripts Data: (Mihalcea and Pedersen, 2003; Koehn, 2005; Vilar et al., 2006)

[4]Approx. average size of training data for all languages.

[5]github.com/artetxem/vecmap

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

Corpus Size	De-En		Ro-En		En-Fr	
	GIZA++	Ours	GIZA++	Ours	GIZA++	Ours
Test Set	44.2	**33.3** *(-10.9)*	42.8	**34.3** *(-8.5)*	26.9	**14.9** *(-12.0)*
1000	41.0	**31.1** *(-9.9)*	41.5	**33.6** *(-7.9)*	20.0	**11.4** *(-8.6)*
2000	37.7	**29.1** *(-8.6)*	39.6	**32.9** *(-6.7)*	17.2	**10.1** *(-7.1)*
5000	34.5	**26.9** *(-7.6)*	38.2	**32.0** *(-6.2)*	14.0	**8.5** *(-5.5)*
10,000	31.9	**25.5** *(-6.4)*	36.1	**30.4** *(-5.7)*	11.7	**7.5** *(-4.2)*
20,000	29.3	**24.2** *(-5.1)*	35.2	**30.3** *(-4.9)*	10.0	**7.1** *(-2.9)*
50,000	26.6	**22.6** *(-4.0)*	34.2	**29.7** *(-4.5)*	8.6	**6.3** *(-2.3)*
100,000	25.4	**21.9** *(-3.5)*	33.4	**29.3** *(-4.1)*	7.8	**6.1** *(-1.7)*
200,000	24.0	**21.2** *(-2.8)*	32.7	**29.4** *(-3.3)*	7.0	**5.8** *(-1.2)*
500,000	21.6	**20.3** *(-1.3)*	26.5	**25.5** *(-1.0)*	6.1	**5.7** *(-0.4)*
1,000,000	20.7	**20.1** *(-0.6)*	*n/a*	*n/a*	6.1	**5.5** *(-0.6)*
1,900,000	20.6	**19.9** *(-0.7)*	*n/a*	*n/a*	*n/a*	*n/a*

Table 1: Main Results. Alignment Error Rate (AER) of EE-GIZA++ vs. GIZA++ baseline (lower is better). Test set is included in corpus size. Ro-En 500K is full 448K training set. Bidirectional, symmetrized (grow-diag-final).

6 Results

Main results are in Table 1. EE-GIZA++ consistently outperforms GIZA++ by a large margin in every tested scenario. When aligning the test set alone with no additional bitext, our method outperforms GIZA++ by 8.5 AER for Ro-En, 10.9 AER for De-En, and 12 AER for En-Fr.

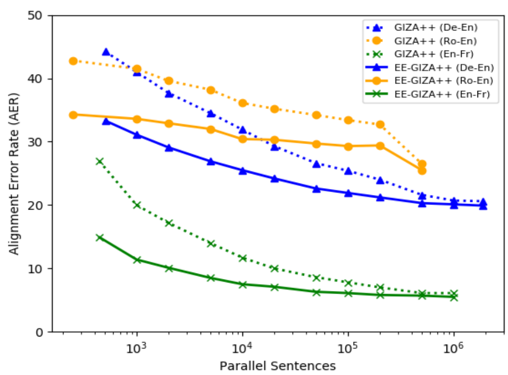

Figure 2: Visualization of Main Results. Alignment Error Rate (AER) of EE-GIZA++ vs. GIZA++ baseline for increasing amounts of training data. Lower is better.

Supplemental Results: High-Resource We compare EE-GIZA++ with existing models in high-resource settings (full training set). These use additional resources like mBERT or data-hungry NMT models that likely fail in low-resource settings (our focus). We perform on-par. Notably, Garg et al. (2019a) use GIZA++ output as supervision. EE-GIZA++ performs better than GIZA++, so AER might improve if supervised with our alignments.

Statistical Baselines	**De-En**	**Ro-En**	**En-Fr**
GIZA++	20.6	26.5	6.2
eflomal*	22.6	25.1	8.2
fast-align*	27.0	32.1	10.5
Massively-Multilingual			
Sabet et al. (2020)*	18.8	27.2	7.6
Dou and Neubig (2021)	15.6	23.0	4.4
no fine-tuning	17.4	27.9	5.6
Bilingual NMT-Based			
Zenkel et al. (2019)	21.2	27.6	10.0
Garg et al. (2019b)	20.2	26.0	7.7
using GIZA++ output	16.0	23.1	4.6
Zenkel et al. (2020)	16.3	23.4	5.0
Chen et al. (2020)	15.4	21.2	4.7
Ours	19.9	25.5	5.3

Table 2: Supplemental results in high-resource settings compared to models that use additional resources. "Massively multilingual" models use mBERT. NMT models likely fail in low-resource (our focus). Bidirectional. *reported in Dou and Neubig (2021).

7 Conclusion and Future Work

We introduce EE-GIZA++, an unsupervised enhancement to GIZA++ that uses word embeddings for improved word alignment in low-resource settings, without the use of NMT or massively-multilingual language models that to-date have been the strongest competitors to GIZA++. EE-GIZA++ outperforms GIZA++ by 8.5, 10.9, and 12 AER in lowest-resource settings for Ro-En, De-En, and En-Fr, respectively. Future work should examine performance of EE-GIZA++ on a diverse set of languages with varying scripts and amounts of data available.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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. An effective approach to unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 194–203, Florence, Italy. Association for Computational Linguistics.

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.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. 2020. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 566–576, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. *arXiv preprint arXiv:2101.08231*.

Sarthak Garg, Joel Ruben Antony Moniz, Anshu Aviral, and Priyatham Bollimpalli. 2019a. Learning to relate from captions and bounding boxes. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6597–6603, Florence, Italy. Association for Computational Linguistics.

Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. 2019b. Jointly learning to align and translate with transformer models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4453–4462, Hong Kong, China. Association for Computational Linguistics.

Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. *Nature*, 585:357–362.

Masoud Jalili Sabet, Heshaam Faili, and Gholamreza Haffari. 2016. Improving word alignment of rare words with word embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3209–3215, Osaka, Japan. The COLING 2016 Organizing Committee.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. *CoRR*, abs/1804.07755.

Xintong Li, Lemao Liu, Zhaopeng Tu, Shuming Shi, and Max Meng. 2018. Target foresight based attention for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1380–1390.

Rada Mihalcea and Ted Pedersen. 2003. An evaluation exercise for word alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts: data driven machine translation and beyond*, pages 1–10.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mathias Müller. 2017. Treatment of markup in statistical machine translation. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 36–46, Copenhagen, Denmark. Association for Computational Linguistics.

Masaaki Nagata, Katsuki Chousa, and Masaaki Nishino. 2020. A supervised word alignment method based on cross-language span prediction using multilingual BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 555–565, Online. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2000a. Improved statistical alignment models. In *Proceedings of the 38th annual meeting of the association for computational linguistics*, pages 440–447.

Franz Josef Och and Hermann Ney. 2000b. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R. Gormley, and Graham Neubig. 2019. Bilingual lexicon induction with semi-supervision in non-isometric embedding spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 184–193, Florence, Italy. Association for Computational Linguistics.

Jan-Thorsten Peter, Arne Nix, and Hermann Ney. 2017. Generating alignments using target foresight in attention-based neural machine translation. *The Prague Bulletin of Mathematical Linguistics*, 108(1):27–36.

Nima Pourdamghani, Marjan Ghazvininejad, and Kevin Knight. 2018. Using word vectors to improve word alignments for low resource machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 524–528, New Orleans, Louisiana. Association for Computational Linguistics.

Guillem Ramírez, Rumen Dangovski, Preslav Nakov, and Marin Soljačić. 2020. On a novel application of wasserstein-procrustes for unsupervised cross-lingual learning. *arXiv preprint arXiv:2007.09456*.

Mohammad Sadegh Rasooli, Noura Farra, Axinia Radeva, Tao Yu, and Kathleen McKeown. 2018. Cross-lingual sentiment transfer with limited resources. *Machine Translation*, 32(1):143–165.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643.

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

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy. Association for Computational Linguistics.

Elias Stengel-Eskin, Tzu-ray Su, Matt Post, and Benjamin Van Durme. 2019. A discriminative neural model for cross-lingual word alignment. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 910–920, Hong Kong, China. Association for Computational Linguistics.

David Vilar, Maja Popović, and Hermann Ney. 2006. Aer: Do we need to "improve" our alignments? In *International Workshop on Spoken Language Translation (IWSLT) 2006*.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A.

Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. Hmm-based word alignment in statistical translation. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2019. Adding interpretable attention to neural translation models improves word alignment. *arXiv preprint arXiv:1901.11359*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2020. End-to-end neural word alignment outperforms GIZA++. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1605–1617, Online. Association for Computational Linguistics.