An Exploration in L2 Word Embedding Alignment

Peiyu Liao
Department of Computer Science
Stanford University
Stanford, CA 94305
pyliao@stanford.edu

Abstract

Unsupervised cross-lingual word embedding space alignment is seen useful in bilingual dictionary construction without parallel data. In this project, we propose that the word embeddings trained from a second language (L2) corpus is likely to achieve a better word-by-word alignment with the source first language (L1) embedding than the original target L1 embedding. Two sets of English corpora, one from Chinese speakers with 880K sentences, and the other from native speakers with 480K sentences, are collected from arXiv.org. Another set of L2 language learner essay corpus has also been aggregated. The word embeddings are trained on each corpus and aligned with several source L1 embedding models for evaluation. The word translation evaluation has seen unsatisfactory results on all experiments in unsupervised alignment, thus the hypothesis fails to be justified from these experiments. Nonetheless, several key findings are found through analysis, which provide useful insights into further improvement on the task.

1 Introduction

Dictionaries have been a crucial part of many machine translation systems. Although modern Neural Machine Translation (NMT) systems [2] trained on parallel corpus no longer rely much on a pre-built dictionary, recent works on unsupervised machine translation [13; 14] have claimed the bilingual dictionary to be a critical initialization step for the models. Unsupervised cross-lingual word embedding alignment for bilingual dictionary construction is thus an interesting topic that may be beneficial to some downstream tasks.

Recent attempts in unsupervised cross-lingual word embedding alignment include the adversarial training method by Zhang et al. [22] which the training procedure is enhanced by Conneau et al. [6], and the multi-lingual neural langauge model proposed by Wada et al. [20] that shares parameters of the model among different languages. The experiments of these methods are all based on native corpus, or L1 corpus.

We propose a novel idea to conduct similar experiments on non-native corpus, in particular L2 corpus, with the aim of further improving word-by-word mapping. The hypothesis is that an L2 corpus of a target language that is based on native speakers of a source language is linguistically more similar to an L1 corpus of the source language than an L1 corpus of the target language. With this intuition, the cross-lingual word embedding alignment is likely to be enhanced by using an L2 corpus instead of an L1 corpus in training the target word embeddings.

This project will be based heavily on L2 data collection, experiments to verify the hypothesis, and analysis of the experimental results.

2 Related Work

Cross-lingual word alignment based on distributed word representations can be traced back to Mikolov et al. [16] that learns linear mapping between the source and target word embeddings by formulating the problem into an optimization problem, and using a small parallel vocabulary as anchor points to solve the problem. Some later works also use digits [1] or common alphabets [19] as parallel vocabulary. These methods either require a small seed dictionary, or cannot generalize to languages without common lexicons.

Various attempts in unsupervised cross-lingual word embedding alignment have been made. Cao et al. [5] assumed that the hidden states of different languages follow normal distribution, and the alignment is made by matching the parameters of the distributions. Zhang et al. [22] made no such assumption and proposed a method based on adversarial training, which involves a generator learning the mapping and a discriminator trying to differentiate between a mapped embedding from the source space and an embedding originally in the target space. Conneau et al. [6] extended on this method, proposing a better metric for model selection that improves performance.

These related works are based on L1 corpus. To the best of our knowledge, no attempts were made in experimenting with L2 corpus in cross-lingual word embedding alignment, or making use of word embeddings trained on an L2 corpus.

3 Approach

The main goal of this project is to verify the hypothesis that word embeddings trained from an L2 corpus better align with the source L1 embeddings than the target L1 embeddings. The pipeline of the approach and how the baseline is set up are described below. The project is available on GitHub[1].

From here on, when we mention aligning corpus, it is equivalent to aligning embeddings trained on the corpus.

3.1 Pipeline

3.1.1 Data Collection

Unlike some other tasks where the model design is the most crucial, there is essentially no new model involved in this project. Collecting clean, consistent, and large enough data for the source L1 corpus, L2 corpus, and target L1 corpus demands the most attention. In this project, we focus on the Chinese to English alignment task where the source is Chinese, target is English, and the L2 corpus is in English written by Chinese native speakers.

A main source of the L2 data is the language learner corpora. An aggregated corpus is built from various free data sources, including EFCAMDAT 2 [11; 8], NUCLE [7], and ICNALE [12], composing of essay data written by students from China, Hong Kong, Taiwan, and Singapore.

These data are limited in amount. In order to build a larger corpus, a web crawler[2] is implemented to collect text data from papers on arXiv[3]. Both the L2 and the target L1 corpus are collected. To identify the native language of the authors, we compiled a list of common last names and higher-education institutions[4] for identifying Chinese and English native users.

It is most ideal for the three corpora under alignment to be from similar domains. However, no L1 data source can be found for the essay corpus, and we also have not identified a centralized Chinese paper archive on the web that is bot-friendly. For the essay corpus, we decided to use Weibo corpus [17] as the L1 source corpus and Twitter corpus[5] as the L1 target corpus, as the content tends to be more colloquial and might better match with the level of essays written by language learners.

[1] `https://github.com/pyliaorachel/bridge-with-l2`

[2] `https://github.com/pyliaorachel/paperscraper`

[3] `https://arxiv.org/`

[4] The list is available on the project GitHub repository.

[5] The GloVe pretrained embedding of Twitter is directly used. The source of the corpus is not known.

For the paper data, a crawler[6] for certain topics on Wikipedia is built, and text data related to some selected topics[7] that match the arXiv paper data are collected.

3.1.2 Data Cleaning

The papers from arXiv are gathered in LaTex format. All commands, equations, and other components that are not pure text are removed as much as possible using regular expression. As a result, some discontinuance in text are present.

The Chinese corpus from Wikipedia is segmented with jieba[8]. All text is transformed to traditional Chinese with OpenCC[9].

3.1.3 Word Embedding Training

Three popular word embedding training methods with standard implementations available are word2vec [15], GloVe [18], and fastText [4]. In this project, most experiments will be based on word2vec for its convenient use[10] and better performance in the experiments. FastText and GloVe are occasionally used for comparison purpose or pretrained embedding availability.

3.1.4 Word Embedding Space Alignment

Since the comparison between L1 and L2 corpora is the focus, the alignment method with implementation convenient to use is adopted in our approach. The work by Conneau et al. [6] is directly available[11] for our use. Here we provide details in the alignment method.

Technically, we want to learn a linear mapping W that maps between the word embeddings $X \in \mathbb{R}^{d\times n}$ in the source space and the word embeddings $Y \in \mathbb{R}^{d\times n}$ in the target space by solving:

$$W^* = \arg\min_{W\in M_d(\mathbb{R})} \|WX - Y\|_F \tag{1}$$

where d is the embedding dimension, n is the number of words, and $M_d(\mathbb{R})$ is the $d \times d$ matrix space of real numbers. Posing an orthogonality constraint on W is found by [21] to improve the results. The problem then reduces to the orthogonal Procrustes problem [10] and can be solved by singular valud decomposition (SVD).

Solving Procrustes requires X and Y to be in parallel. To solve the linear mapping in an unsupervised manner, a discriminator is first trained to learn an initial W. The best-matched word pairs are used in Procrustes to further refine the mapping. Finally, the metric of the space is tuned to spread out points in dense regions.

Here we briefly describe the adversarial training process.

Adversarial Training In the adversarial network, a discriminator is trained to correctly identify whether the given word is from the source or target language. On the other hand, the mapping matrix W is trained to prevent the discriminator from correct predictions. The procedure generally follows from the Generative Adversarial Network (GAN) [9].

Let θ_D be the parameters of the discriminator model, and $P_{\theta_D}(\text{source} = 1|z)$ be the probability that the discriminator considers the embedding vector z be from the source embedding. The discriminator objective is:

$$\mathcal{L}_D(\theta_D|W) = -\frac{1}{n}\sum_{i=1}^{n} \log P_{\theta_D}(\text{source} = 1|Wx_i) - \frac{1}{m}\sum_{i=1}^{m} \log P_{\theta_D}(\text{source} = 0|y_i) \tag{2}$$

[6] `https://github.com/pyliaorachel/wiki-topic-extractor`

[7] Topics include: Computer Science, Electrical Engineering, Systems Science, Math, Physics, Statistics, Economics, Science, Engineering

[8] `https://github.com/fxsjy/jieba`

[9] `https://github.com/BYVoid/OpenCC`

[10] Gensim package: `https://radimrehurek.com/gensim/models/word2vec.html`

[11] `https://github.com/facebookresearch/MUSE`

	wiki	wiki_t	arxiv_l2	arxiv_l2	arxiv_l2	arxiv_l1	arxiv_l1	essay	weibo	twitter
Num. of Tokens	54M	26M	14.3M	8.3M	2.5M	8.6M	2.7 M	14.8M	18.9M	27B
Num. of Sentences	-	880K	880K	480K	150K	480K	150K	1.3M	1.1M	-
Corpus Type	L1	L1	L2	L2	L2	L1	L1	L2	L1	L1
Text Language	zh	zh	en	en	en	en	en	en	zh	en
User Language	zh	zh	zh	zh	zh	en	en	zh	zh	en

Table 1: The size information of various corpora. `wiki_t`: Wikipedia with selected topics; `arxiv_lx`: arXiv L1 or L2 corpus.

where x is the source embedding, y is the target embedding, and n, m are the number of words for discrimination in each domain.

The mapping objective is the opposite:

$$\mathcal{L}_W(W|\theta_D) = -\frac{1}{n}\sum_{i=1}^{n}\log P_{\theta_D}(\text{source}=0|Wx_i) - \frac{1}{m}\sum_{i=1}^{m}\log P_{\theta_D}(\text{source}=1|y_i) \quad (3)$$

3.2 Baseline

At the time of the milestone, only around 150K L1 English sentences and 700K L2 English sentences from Chinese users[12] were collected. As a baseline experiment, the L2 corpus is truncated to 150K sentences. Word embeddings are trained on these two sets of corpora, and each is trained to be aligned with a pretrained source L1 Chinese word embedding model trained on Wikipedia[13].

Note that we rerun the baseline experiments with the settings provided in the next section, hence the results will be different from the milestone.

4 Experiments

We will explore the alignment performance with more data, different data sources, and different word embedding training methods in the experiments.

4.1 Data

The details of the dataset are provided in the previous section. Here we give a summary. The size information of the corpora is provided in Table 1.

arXiv The source L1 Chinese corpus is collected from Wikipedia based on some selected topics. The L2 English corpus and the target L1 English corpus are collections of paper text crawled from arXiv.org, in which the native language of the paper authors is detected by last name and institution matching. The topics of the papers are mainly related to computer science.

Due to various constraints[14], only around 480K L1 sentences and 880K L2 sentences can be collected in the end. The experiments will be based on the full size of these two corpora, and in addition, the L2 corpus truncated to 480K sentences.

Essay The L2 essay corpus is an aggregation of the EFCAMDAT 2 [11; 8], NUCLE [7], and ICNALE [12] corpora. The essays are written by students from China, Hong Kong, Taiwan, and Singapore. Weibo [17] is the source L1 corpus, and the Twitter pretrained GloVe embeddings are used as the target L2 word embedding model.

4.2 Evaluation Method

To evaluate the aligned mappings and the constructed bilingual dictionary, we conduct the same word translation task as in [6]. For the zh-en language pair, the mapping is tested on 1,500 query source

[12] Interestingly, more data can be collected for the L2 corpus than the L1 corpus under our constraints on last name and institution matching.

[13] `https://github.com/Kyubyong/wordvectors`

[14] Aside from the time constraint, the IP of our server is likely banned by arXiv.

words that are mapped to the target space. For each source word, the predicted word translation is the one with the largest Cross-Domain Similarity Local Scaling (CSLS) measure[15] among the set of target words. The translation precision@k is reported for k = 1, 5, 10.

To understand how the quality of the monolingual word embeddings affect in the alignment performance, word similarity evaluation [3] is carried out for each L2 and L1 target embeddings.

4.3 Experimental Details

For the word embedding training of the L2 corpus and the target L1 corpus with word2vec and fastText, the embedding dimension is 300, while those trained with GloVe has embedding dimension 200 to be consistent with the pretrained Twitter embeddings. The minimum word occurrence count is 5, and they are trained over 5 iterations.

The discriminator in the adversarial training has a hidden dimension of 2048. The training batch size is 32, and the initial learning rate is set to 0.1 with a decay rate of 0.98. These are the settings in the original implementation. To account for the smaller vocabulary size in the word embeddings, the number of most frequent words for discrimination in the adversarial step is set to 5000, and the maximum vocabulary size is 40000.

All experiments are done over 20 epochs, each being 100K iterations. The step size of the discriminator is 5 in each iteration.

There are several groups of experiments conducted for different purposes. The details of each are described below.

Large Corpus v.s. Large Corpus, Unsupervised Unsupervised word alignment is run for Wikipedia Chinese fastText embeddings to English embeddings to verify the above settings are reasonable. It is also run for Wikipedia Chinese fastText embeddings to Common Crawl English fastText embeddings to test the method on a large corpus of a different domain. All fastText embeddings are pretrained and downloaded from their website[16].

We refer to these two experiments as `wiki-wiki` and `wiki-crawl`.

L2 v.s. Target L1, Unsupervised For the paper data, except for the baseline experiments, the L2 and target L1 corpora of various sizes are aligned with two source embeddings, the large pretrained Wikipedia Chinese word2vec embedding, and the smaller and more topic-specific Wikipedia topic corpus, using the unsupervised method. The corpora with 150K sentences in baseline experiments are only aligned with the large pretrained Wikipedia corpus. This is the major set of experiments for verifying the hypothesis. Aligning with two different embeddings can provide further information about how the size and domain of the source corpus affect the alignment performance.

We refer to the experiments aligning the L2 corpus against the large Wikipedia corpus as `wiki-arxiv_l2`, and the ones aligning the target L1 corpus against the large Wikipedia corpus as `wiki-arxiv_l1`. The ones against the Wikipedia topic corpus are named similarly but with prefix `wiki_t`.

For the essay data, the source L1 Weibo corpus is aligned with the L2 essay corpus and the target L1 Twitter corpus. These two experiments are named `weibo-essay` and `weibo-twitter`.

L2 v.s. Target L1, Supervised Supervised word alignment, which is essentially solving the Procrustes problem as described in Section 3.1.4 with a parallel dictionary of 5000 pairs of words, is carried out for all experiments except the large corpora experiments. The supervised experiments can be viewed as the topline alignment performance for each pair of the embeddings.

L2 v.s. Target L1, Different Word Embedding Training Methods For the alignment experiments with the Wikipedia topic corpus, both the word2vec and fastText word embeddings are evaluated and compared to unveal the effect of the different word embedding training methods to the alignment performance.

[15] A similarity measurement of the mapped source embedding to the target embedding, as defined in [6].

[16] `https://fasttext.cc/`

	# Sent (tgt)	U / S	Emb	Word Sim (tgt)	NN p@1	NN p@5	NN p@10	CSLS p@1	CSLS p@5	CSLS p@10
wiki-wiki (full)	-	U	f	0.65	32.93	50.53	57.52	36.64	55.95	62.22
wiki-wiki	-	U	f	0.65	1.21	3.42	4.56	1.07	4.2	6.70
wiki-crawl	-	U	f	0.71	0	0	0	0	0.075	0.075
wiki-arxiv_l2	150K	S	w	0.20	1.73	4.03	6.63	2.45	4.90	6.63
wiki-arxiv_l2	150K	U	w	0.20	0	0	0	0	0	0
wiki-arxiv_l1	150K	S	w	0.14	1.11	3.21	4.94	1.48	3.21	5.69
wiki-arxiv_l1	150K	U	w	0.14	0	0	0	0	0	0
wiki_t-arxiv_l2	880K	S	w	0.24	4.55	10.88	14.74	5.37	11.16	15.01
wiki_t-arxiv_l2	880K	U	w	0.24	0	0	0.14	0	0	0.14
wiki_t-arxiv_l2	880K	S	f	0.072	2.07	4.82	6.47	2.20	4.27	6.47
wiki_t-arxiv_l2	880K	U	f	0.072	0	0	0	0	0	0
wiki-arxiv_l2	880K	S	w	0.24	4.13	10.13	14.25	6.13	12.13	15.88
wiki-arxiv_l2	880K	U	w	0.24	0	0	0	0	0	0
wiki_t-arxiv_l2	480K	S	w	0.21	3.31	8.12	10.53	3.61	8.57	10.98
wiki_t-arxiv_l2	480K	U	w	0.21	0	0	0.15	0	0	0
wiki_t-arxiv_l2	480K	S	f	0.031	1.80	3.46	5.86	1.95	3.91	5.86
wiki_t-arxiv_l2	480K	U	f	0.031	0	0	0	0	0	0
wiki-arxiv_l2	480K	S	w	0.21	4.68	10.33	14.74	4.55	12.26	16.12
wiki-arxiv_l2	480K	U	w	0.21	0	0	0.28	0	0	0
wiki_t-arxiv_l1	480K	S	w	0.175	3.46	7.59	10.12	3.86	7.19	10.25
wiki_t-arxiv_l1	480K	U	w	0.175	0	0	0	0	0	0
wiki_t-arxiv_l1	480K	S	f	0.173	1.46	4.39	6.39	1.86	4.39	6.13
wiki_t-arxiv_l1	480K	U	f	0.173	0	0	0	0	0	0
wiki-arxiv_l1	480K	S	w	0.175	3.21	8.52	10.99	4.57	9.88	13.83
wiki-arxiv_l1	480K	U	w	0.175	0	0.12	0.25	0	0.12	0.12
weibo-essay	1.3M	S	g	0.29	2.50	6.51	8.50	2.50	7.01	9.64
weibo-essay	1.3M	U	g	0.29	0	0.13	0.13	0	0	0
weibo-twitter	-	S	g	0.54	6.86	15.69	20.03	7.14	15.69	19.19
weibo-twitter	-	U	g	0.54	0	0	0	0	0.14	0.14

Table 2: Precision@k for k = 1, 5, 10 in the word translation task, and word similarity scores for target embeddings. U: unsupervised, S: supervised, f: fastText, w: word2vec, g: GloVe. `wiki-wiki (full)` is with the settings used in the original paper with the number of most frequent words for discrimination being 75000, and maximum vocabulary size being 200000.

4.4 Results

The precision@k of the word translation evaluation task and the word similarity score on each monolingual target corpus of the various experiments are reported in Table 2. `wiki-wiki (full)` is the `wiki-wiki` experiment with enhanced settings that produces the same results as in the original paper.

General Performance The performance is unexpectedly poor for unsupervised alignments. `wiki-wiki` also does not compete with `wiki-wiki (full)`. All precision values are close to 0 even for p@10 except for `wiki-wiki`. Surprisingly, this is also true for `wiki-crawl`, although both corpora are large. The performance for supervised alignments is slightly better as expected.

L2 v.s. Target L1 Comparing the L2 and target L1 embeddings of 480K sentences, it is observed that the performance does not differ much for both unsupervised and supervised alignments against the Wikipedia topic corpus. The L2 performance is slightly better than the target L1 performance when aligned with the large Wikipedia corpus in the supervised version. However, given the minor performance difference, the hypothesis that the L2 embedding aligns better to the L1 source embedding than the L1 target embedding cannot be verified.

FastText v.s. Word2vec Under similar conditions, word2vec generally outperforms fastText in both supervised and unsupervised alignments. The word similarity score of word2vec embeddings is also significantly better than fastText for the L2 corpus. This suggests that the `wiki-wiki (full)` results may be further improved if word2vec is used in place of fastText[17].

5 Analysis

In this section, possible reasons that led to the results are inspected, and some key observations that provide insights for further improvement on the unsupervised alignment task are concluded.

[17]We have attempted to verify this with experiments. However, there is no pretrained word2vec English embeddings found online, and training one from scratch has exceeded the space limit on the server. We plan to conduct this experiment in the future.

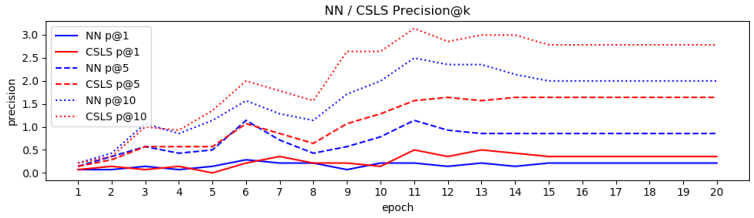

Figure 1: Precision over epochs for the `wiki-wiki` experiment.

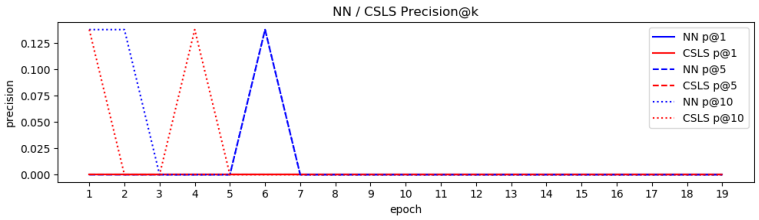

Figure 2: Precision over epochs for the `wiki_t-arxiv_l2` 880K experiment.

The Effect of Corpus Size and Vocabulary Size Experiments with more data have outperformed the baseline experiments both in the alignment precision scores and the monolingual word similarity scores. While the L2 corpus with 880K sentences outperformed that with 480K sentences in supervised alignment, the performance slightly degrades under the unsupervised setting.

Moreover, from the significantly better performance of `wiki-wiki (full)` than `wiki-wiki`, it is obvious that the vocabulary size is critical to the overall performance. However, most of our embeddings trained from the corpora or the pretrained embeddings do not have as much as 200000 words in vocabulary, hence setting it to 40000 in our experiments. This is an effect that is crucial but cannot be efficiently improved unless we have larger corpus.

We conclude that a larger corpus size benefits the alignment, but the critical reason is not because the word embeddings can be trained better, but more words can be included in the vocabulary.

The Effect of Training Epochs Figure 1 and Figure 2 show the precisions over training epochs for the slightly more successful `wiki-wiki` experiment and the unsuccessful `wiki_t-arxiv_l2` experiment with 880K sentences in the L2 corpus.

The performance stops to improve after around 11 epochs in the `wiki-wiki` experiment and around 7 epochs in the `wiki_t-arxiv_l2` experiment. Similar behavior can be observed in other experiments. It is thus justifiable to conclude that increasing the training epochs beyond 20 is likely not beneficial to the alignment performance.

The Effect of Corpus Choice The results from the `wiki-crawl` and `weibo-twitter` experiments are surprising in the sense that both the source and target embeddings are trained on a fairly large corpus and with good word similarity scores, yet the performance is far below `wiki-wiki`.

This is a crucial indication that the poor performance of our experiments with the arXiv corpora may simply be due to the fact that the source and target corpora are not aligned as well as Wikipedia corpora. As mentioned in [6], the good performance in their work likely benefits from the similar co-occurrence statistics in the Wikipedia corpora. This further suggests that the corpus choice may be the dominant factor in determining the word alignment performance rather than the method itself.

The Effect of Word Embedding Methods To understand the cause of the performance difference between fastText and word2vec, the top-3 similar words of some chosen words are inspected with the result shown in Table 3.

For fastText, words with similar spellings, or subword structures, are closer in the embedding space. For word2vec, words that are semantically similar are closer in the embedding space. Since subword

word		word2vec	fastText
computer	top-1	robotics	brain-computer
	top-2	graphics	supercomputer
	top-3	electronic	human-computer
simple	top-1	straightforward	newhope-simple
	top-2	generic	simplex-bso
	top-3	universal	simplex
brain	top-1	biological	brainweb
	top-2	tissue	grain
	top-3	chemical	terrain

Table 3: Word2vec and fastText top-3 neighbors for the `arxiv_l2` 880K corpus.

structures of a language can be totally non-existent in another language, we claim that word2vec is a better choice in word embedding alignment since it captures semantic meanings better. The experimental results also support the claim.

6 Conclusion

In this project, a number of experiments were set up to verify the hypothesis that word embeddings trained on an L2 corpus aligns better with the source L1 embeddings than embeddings trained on an target L1 corpus.

The hypothesis fails to be justified due to the generally poor performance across the experiments. While the performance of supervised alignment is generally better for the L2 embeddings than the target L1 embeddings, the unsupervised alignment is slightly worse.

Nevertheless, several key findings that may help in further improvement on the unsupervised alignment tasks are found through the experiments. A larger corpus size is likely to improve the performance as the vocabulary size can increase, which is critical in the overall alignment performance. The number of training epochs has minor effect. The most crucial component might be the choice of corpus, and the naturally-aligned Wikipedia corpora may be most suitable for the task at this point; this is regardless of whether the hypothesis can be proved true or not, since the performance gain from an L2 corpus may not be comparable to that from an aligned corpus. Furthermore, we suggest that using word2vec instead of fastText as the word embedding training method may further enhance the alignment performance.

Although the main goal of the project is not achieved, some interesting insights into the L2 corpus and how its use can be extended to topics other than grammar correction, natural language identification, and essay scoring can be provided by the project, and hopefully it can be a motivation to future works.

7 Additional Information

Mentor: Chris Manning

References

[1] ARTETXE, M., LABAKA, G., AND AGIRRE, E. 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)* (2017), vol. 1, pp. 451–462.

[2] BAHDANAU, D., CHO, K., AND BENGIO, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[3] BAKAROV, A. A survey of word embeddings evaluation methods. *arXiv preprint arXiv:1801.09536* (2018).

[4] BOJANOWSKI, P., GRAVE, E., JOULIN, A., AND MIKOLOV, T. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics 5* (2017), 135–146.

[5] CAO, H., ZHAO, T., ZHANG, S., AND MENG, Y. A distribution-based model to learn bilingual word embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (2016), pp. 1818–1827.

[6] CONNEAU, A., LAMPLE, G., RANZATO, M., DENOYER, L., AND JÉGOU, H. Word translation without parallel data. *arXiv preprint arXiv:1710.04087* (2017).

[7] DAHLMEIER, D., NG, H. T., AND WU, S. M. Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications* (2013), pp. 22–31.

[8] GEERTZEN, J., ALEXOPOULOU, T., AND KORHONEN, A. Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project* (2013).

[9] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A., AND BENGIO, Y. Generative adversarial nets. In *Advances in neural information processing systems* (2014), pp. 2672–2680.

[10] GOWER, J. C. Generalized procrustes analysis. *Psychometrika 40*, 1 (1975), 33–51.

[11] HUANG, Y., MURAKAMI, A., ALEXOPOULOU, T., AND KORHONEN, A. Dependency parsing of learner english. *International Journal of Corpus Linguistics 23*, 1 (2018), 28–54.

[12] ISHIKAWA, S. The icnale and sophisticated contrastive interlanguage analysis of asian learners of english. *Learner corpus studies in Asia and the world 1* (2013), 91–118.

[13] LAMPLE, G., CONNEAU, A., DENOYER, L., AND RANZATO, M. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043* (2017).

[14] LAMPLE, G., OTT, M., CONNEAU, A., DENOYER, L., AND RANZATO, M. Phrase-based & neural unsupervised machine translation. *arXiv preprint arXiv:1804.07755* (2018).

[15] MIKOLOV, T., CHEN, K., CORRADO, G., AND DEAN, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[16] MIKOLOV, T., LE, Q. V., AND SUTSKEVER, I. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168* (2013).

[17] NATURAL LANGUAGE PROCESSING AND INFORMATION RETRIEVAL SHARING PLATFORM. 500 万微博语料, 2018.

[18] PENNINGTON, J., SOCHER, R., AND MANNING, C. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (2014), pp. 1532–1543.

[19] SMITH, S. L., TURBAN, D. H., HAMBLIN, S., AND HAMMERLA, N. Y. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859* (2017).

[20] WADA, T., AND IWATA, T. Unsupervised cross-lingual word embedding by multilingual neural language models. *arXiv preprint arXiv:1809.02306* (2018).

[21] XING, C., WANG, D., LIU, C., AND LIN, Y. 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* (2015), pp. 1006–1011.

[22] ZHANG, M., LIU, Y., LUAN, H., AND SUN, M. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (2017), vol. 1, pp. 1959–1970.