Adversarial Deep Averaging Networks for Cross-Lingual Sentiment Classification

Xilun Chen[†] xlchen@cs.cornell.edu
Yu Sun[†] ys646@cornell.edu
Ben Athiwaratkun[‡] pa338@cornell.edu
Claire Cardie[†] cardie@cs.cornell.edu
Kilian Weinberger[†] kqw4@cornell.edu

†Dept. of Computer Science, Cornell University, Ithaca NY, USA
‡Dept. of Statistical Science, Cornell University, Ithaca NY, USA

Abstract

In recent years great success has been achieved in sentiment classification for English, thanks in part to the availability of copious annotated resources. Unfortunately, most languages do not enjoy such an abundance of labeled data. To tackle the sentiment classification problem in low-resource languages without adequate annotated data, we propose an Adversarial Deep Averaging Network (ADAN[1]) to transfer the knowledge learned from labeled data on a resource-rich source language to low-resource languages where *only unlabeled* data exists. ADAN has two discriminative branches: a *sentiment classifier* and an adversarial *language discriminator*. Both branches take input from a shared *feature extractor* to learn hidden representations that are simultaneously indicative for the classification task and *invariant* across languages. Experiments on Chinese and Arabic sentiment classification demonstrate that ADAN significantly outperforms state-of-the-art systems.

1 Introduction

Many state-of-the-art models for sentiment classification (Socher et al., 2013; Iyyer et al., 2015; Tai et al., 2015) are supervised learning approaches that rely on the availability of an adequate amount of labeled training data. For a few resource-rich languages including English, such labeled data is indeed available. For the vast majority of languages, however, it is the norm that only a limited amount of annotated text exists. Worse still, many low-resource languages have no labeled data at all.

To aid the creation of sentiment classification systems in such low-resource languages, an active research direction is cross-lingual sentiment classification (CLSC) in which the abundant resources of a *source* language (likely English, denoted as SOURCE) are leveraged to produce sentiment classifiers for a *target* language (TARGET). In general, CLSC methods make use of general-purpose bilingual resources — such as hand-crafted bilingual lexica or parallel corpora — to alleviate or eliminate the need for task-specific TARGET annotations. In particular, the bilingual resource of choice for the majority of previous CLSC models is a full-fledged Machine Translation (MT) system (Wan, 2008, 2009; Lu et al., 2011; Zhou et al., 2016), a component that is expensive to obtain. In this work, we propose a **language-adversarial training** approach that does not need a highly engineered MT system, and requires orders of magnitude less in terms of the size of parallel corpus. Specifically, we propose an Adversarial Deep Averaging Network (ADAN) that leverages a set of Bilingual Word Embeddings (BWEs, Zou et al., 2013) trained on bitexts, in order to eliminate the need for labeled TARGET training data[2].

We introduce the ADAN model in §2, and in §3 evaluate ADAN using English as the SOURCE with two TARGET choices: Chinese and Arabic. ADAN is first compared to two baseline systems: i) one trained only on labeled SOURCE data, relying on BWEs for cross-lingual generalization; and ii) a domain adaptation method (Chen et al., 2012) that views the two languages simply as two distinct domains. We then validate ADAN against two state-of-the-art CLSC methods: iii) an approach that employs a powerful MT system, and iv) the cross-lingual "distillation" approach of Xu and Yang (2017) that makes direct use of a parallel corpus

[1]The source code of ADAN is available at https://github.com/ccsasuke/adan

[2]When not using any TARGET annotations, the setting is sometimes referred to as *unsupervised* (in the target language) in the literature. Similarly, when some labeled data is used, it is called the *semi-supervised* setting.

(see §3.2). In all cases, we find that ADAN achieves statistically significantly better results.

We further investigate the semi-supervised setting, where a small amount of annotated TARGET data exists, and show that ADAN continues to outperform the alternatives given the same amount of TARGET supervision (§3.3.1). We provide an analysis and visualization of ADAN (§3.3.2), shedding light on how our approach manages to achieve its strong cross-lingual performance. Additionally, we study the bilingual resource that ADAN depends on, the Bilingual Word Embeddings, and demonstrate that ADAN's performance is robust with respect to the choice of BWEs. Furthermore, even without the pre-trained BWEs (i.e. using random initialized embeddings), ADAN outperforms all but the state-of-the-art MT-based and distillation systems (§3.3.3). This makes ADAN the first CLSC model that outperforms BWE-based baseline systems without relying on *any* bilingual resources.

A final methodological contribution distinguishes ADAN from previous adversarial networks for text classification (Ganin et al., 2016): ADAN minimizes the Wasserstein distance (Arjovsky et al., 2017) between the feature distributions of SOURCE and TARGET (§2.2), which yields better performance and smoother training than the standard GRL training method (Ganin et al., 2016, see §3.3.5).

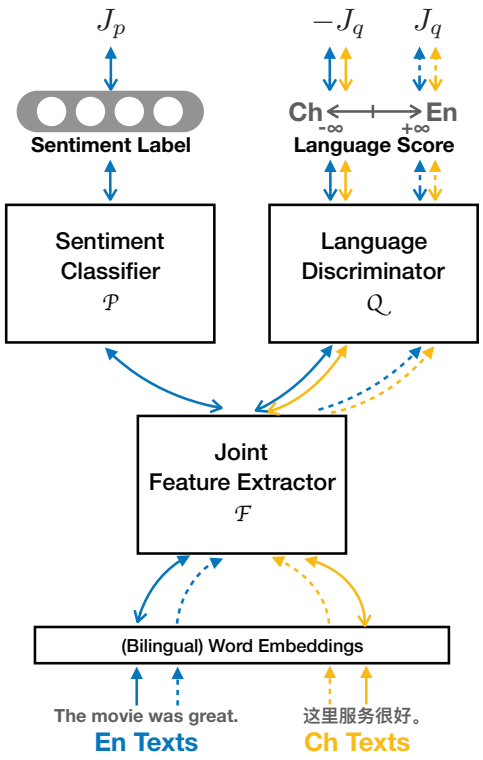

Figure 1: ADAN with Chinese as the target language. The lines illustrate the training flows and the arrows indicate forward and/or backward passes. Blue lines show the flow for English samples while Yellow ones are for Chinese. J_p and J_q are the training objectives of $\mathcal{P}$ and $\mathcal{Q}$, respectively (§2.2). The parameters of $\mathcal{F}$, $\mathcal{P}$ and the embeddings are updated together (solid lines). The parameters of $\mathcal{Q}$ are updated using a separate optimizer (dotted lines) due to its adversarial objective.

2 The ADAN Model

The central hypothesis of ADAN is that an ideal model for CLSC should learn features that both perform well on sentiment classification for the SOURCE, and are invariant with respect to the shift in language. Therefore, as shown in Figure 1, ADAN has a joint *feature extractor* $\mathcal{F}$ which aims to learn features that aid prediction of the *sentiment classifier* $\mathcal{P}$, and **hamper** the *language discriminator* $\mathcal{Q}$, whose goal is to identify whether an input text is from SOURCE or TARGET. The intuition is that if a well-trained $\mathcal{Q}$ cannot tell the language of a given input using the features extracted by $\mathcal{F}$, those features are effectively language-invariant. $\mathcal{Q}$ is hence *adversarial* since it does its best to identify language from learned features, yet good performance from $\mathcal{Q}$ indicates that ADAN is not successful in learning language-invariant features. Upon successful ADAN training, $\mathcal{F}$ should have learned features discriminative for sentiment classification, and at the same time providing no information for the adversarial $\mathcal{Q}$ to guess the language of a given input.

As seen in Figure 1, ADAN is exposed to both SOURCE and TARGET texts during training. Unlabeled SOURCE (blue lines) and TARGET (yellow lines) data go through the language discriminator, while only the labeled SOURCE data pass through the sentiment classifier[3]. The feature extractor and the sentiment classifier are then used for TARGET texts at test time. In this manner, we can train ADAN with labeled SOURCE data and only unlabeled TARGET text. When some labeled TARGET data exist, ADAN could naturally be extended to take advantage of that for improved performance (§3.3.1).

[3] "Unlabeled" and "labeled" refer to sentiment labels; all texts are assumed to have the correct language label.

2.1 Network Architecture

As illustrated in Figure 1, ADAN is a feed-forward network with two branches. There are three main components in the network, a joint *feature extractor* $\mathcal{F}$ that maps an input sequence x to the shared feature space, a *sentiment classifier* $\mathcal{P}$ that predicts the label for x given the feature representation $\mathcal{F}(x)$, and a *language discriminator* $\mathcal{Q}$ that also takes $\mathcal{F}(x)$ but predicts a scalar score indicating whether x is from SOURCE or TARGET.

An input document is modeled as a sequence of words $x = w_1, \ldots, w_n$, where each w is represented by its word embedding v_w (Turian et al., 2010). For improved performance, pre-trained bilingual word embeddings (BWEs, Zou et al., 2013; Gouws et al., 2015) can be employed to induce bilingual distributed word representations so that similar words are closer in the embedded space regardless of language.

A parallel corpus is often required to train high-quality BWEs, making ADAN implicitly dependent on the bilingual corpus. However, compared to the MT systems used in other CLSC methods, training BWEs only requires one to two orders of magnitude less parallel data, and some methods only take minutes to train on a consumer CPU (Gouws et al., 2015), while state-of-the-art MT systems need days to weeks for training on multiple GPUs. Moreover, even with randomly initialized embeddings, ADAN can still outperform some baseline methods that use pre-trained BWEs (§3.3.3). Another possibility is to take advantage of the recent work that trains BWEs with no bilingual supervision (Lample et al., 2018).

We adopt the Deep Averaging Network (DAN) by Iyyer et al. (2015) for the feature extractor $\mathcal{F}$. We choose DAN for its simplicity to illustrate the effectiveness of our language-adversarial training framework, but other architectures can also be used for the feature extractor (§3.3.4). For each document, DAN takes the arithmetic mean of the word vectors as input, and passes it through several fully-connected layers until a softmax for classification. In ADAN, $\mathcal{F}$ first calculates the average of the word vectors in the input sequence, then passes the average through a feed-forward network with ReLU nonlinearities. The activations of the last layer in $\mathcal{F}$ are considered the extracted features for the input and are then passed on to $\mathcal{P}$ and $\mathcal{Q}$. The sentiment classifier $\mathcal{P}$ and the language discriminator $\mathcal{Q}$ are standard feed-forward networks. $\mathcal{P}$ has a softmax layer on top for sentiment classification and $\mathcal{Q}$ ends with a linear layer of output width 1 to assign a language identification score[4].

2.2 Adversarial Training

For clarity, we first introduce an ADAN variant where training is done using Gradient Reversal Layer (Ganin et al., 2016), which is denoted as ADAN-GRL. ADAN-GRL employs standard adversarial training techniques in previous literature (Ganin et al., 2016), but as we will detail later in this section, the training of ADAN-GRL is less stable and the performance is worse compared to our ADAN model (see §3.3.5 for empirical results).

Specifically, in ADAN-GRL, $\mathcal{Q}$ is a binary classifier with a sigmoid layer on top so that the language identification score is always between 0 and 1 and is interpreted as the probability of whether an input text x is from SOURCE or TARGET given its hidden features $\mathcal{F}(x)$. For training, $\mathcal{Q}$ is connected to $\mathcal{F}$ via a Gradient Reversal Layer (Ganin and Lempitsky, 2015), which preserves the input during the a forward pass but multiplies the gradients by $-\lambda$ during a backward pass. λ is a hyperparameter that balances the effects that $\mathcal{P}$ and $\mathcal{Q}$ have on $\mathcal{F}$ respectively. This way, the entire network can be trained in its entirety using standard backpropagation.

Unfortunately, researchers have found that the training of $\mathcal{F}$ and $\mathcal{Q}$ in ADAN-GRL might not be fully in sync (Ganin and Lempitsky, 2015), and efforts need to be made to coordinate the adversarial training. This is achieved by setting λ to a non-zero value only once out of k batches as in practice we observe that $\mathcal{F}$ trains faster than $\mathcal{Q}$. Here, k is another hyperparameter that coordinates the training of $\mathcal{F}$ and $\mathcal{Q}$. When $\lambda = 0$, the gradients from $\mathcal{Q}$ will not be back-propagated to $\mathcal{F}$. This allows $\mathcal{Q}$ more iterations to adapt to $\mathcal{F}$ before $\mathcal{F}$ makes another adversarial update.

To illustrate the limitations of ADAN-GRL and motivate the formal introduction of our ADAN model, consider the distribution of the joint hidden features $\mathcal{F}$ for both SOURCE and TARGET instances:

$$P_{\mathcal{F}}^{src} \triangleq P(\mathcal{F}(x)|x \in \text{SOURCE})$$
$$P_{\mathcal{F}}^{tgt} \triangleq P(\mathcal{F}(x)|x \in \text{TARGET})$$

[4] $\mathcal{Q}$ simply tries to maximize scores for SOURCE texts and minimize for TARGET, and the scores are not bounded.

Algorithm 1 ADAN Training

Require: labeled SOURCE corpus $\mathbb{X}_{src}$; unlabeled TARGET corpus $\mathbb{X}_{tgt}$; Hyperpamameter $\lambda > 0$, $k \in \mathbb{N}$, $c > 0$.

repeat
  ▷ $\mathcal{Q}$ iterations
  for $qiter = 1$ to k **do**
    Sample unlabeled batch $\boldsymbol{x}_{src} \sim \mathbb{X}_{src}$
    Sample unlabeled batch $\boldsymbol{x}_{tgt} \sim \mathbb{X}_{tgt}$
    $\boldsymbol{f}_{src} = \mathcal{F}(\boldsymbol{x}_{src})$
    $\boldsymbol{f}_{tgt} = \mathcal{F}(\boldsymbol{x}_{tgt})$ ▷ feature vectors
    $loss_q = -\mathcal{Q}(\boldsymbol{f}_{src}) + \mathcal{Q}(\boldsymbol{f}_{tgt})$ ▷ Eqn (2)
    Update $\mathcal{Q}$ parameters to minimize $loss_q$
    $ClipWeights(\mathcal{Q}, -c, c)$
  ▷ Main iteration
  Sample labeled batch $(\boldsymbol{x}_{src}, \boldsymbol{y}_{src}) \sim \mathbb{X}_{src}$
  Sample unlabeled batch $\boldsymbol{x}_{tgt} \sim \mathbb{X}_{tgt}$
  $\boldsymbol{f}_{src} = \mathcal{F}(\boldsymbol{x}_{src})$
  $\boldsymbol{f}_{tgt} = \mathcal{F}(\boldsymbol{x}_{tgt})$
  $loss = L_p(\mathcal{P}(\boldsymbol{f}_{src}); \boldsymbol{y}_{src}) + \lambda(\mathcal{Q}(\boldsymbol{f}_{src}) - \mathcal{Q}(\boldsymbol{f}_{tgt}))$ ▷ Eqn (4)
  Update $\mathcal{F}$, $\mathcal{P}$ parameters to minimize $loss$
until convergence

In order to learn language-invariant features, ADAN trains $\mathcal{F}$ to make these two distributions as close as possible for better cross-lingual generalization. In particular, as argued by Arjovsky et al. (2017), previous approaches to training adversarial networks such as ADAN-GRL are equivalent to minimizing the Jensen-Shannon divergence between two distributions, in our case $P_{\mathcal{F}}^{src}$ and $P_{\mathcal{F}}^{tgt}$. And because the Jensen-Shannon divergence suffers from discontinuities, providing less useful gradients for training $\mathcal{F}$, Arjovsky et al. (2017) propose instead to minimize the Wasserstein distance and demonstrate its improved stability for hyperparameter selection.

As a result, departing from the previous ADAN-GRL training method, in our ADAN model, we minimize the Wasserstein distance W between $P_{\mathcal{F}}^{src}$ and $P_{\mathcal{F}}^{tgt}$ according to the Kantorovich-Rubinstein duality (Villani, 2008):

$$W(P_{\mathcal{F}}^{src}, P_{\mathcal{F}}^{tgt}) = \sup_{\|g\|_L \le 1} \mathop{\mathbb{E}}_{f(x)\sim P_{\mathcal{F}}^{src}} [g(f(x))] - \mathop{\mathbb{E}}_{f(x')\sim P_{\mathcal{F}}^{tgt}} \left[g(f(x'))\right] \quad (1)$$

where the supremum (maximum) is taken over the set of all 1-Lipschitz[5] functions g. In order to (approximately) calculate $W(P_{\mathcal{F}}^{src}, P_{\mathcal{F}}^{tgt})$, we use the language discriminator $\mathcal{Q}$ as the function g in (1), whose objective is then to seek the supremum in (1). To make $\mathcal{Q}$ a Lipschitz function (up to a constant), the parameters of $\mathcal{Q}$ are always clipped to a fixed range $[-c, c]$. Let $\mathcal{Q}$ be parameterized by θ_q, then the objective J_q of $\mathcal{Q}$ becomes:

$$J_q(\theta_f) \equiv \max_{\theta_q} \mathop{\mathbb{E}}_{\mathcal{F}(x)\sim P_{\mathcal{F}}^{src}} [\mathcal{Q}(\mathcal{F}(x))] - \mathop{\mathbb{E}}_{\mathcal{F}(x')\sim P_{\mathcal{F}}^{tgt}} \left[\mathcal{Q}(\mathcal{F}(x'))\right] \quad (2)$$

Intuitively, $\mathcal{Q}$ tries to output higher scores for SOURCE instances and lower scores for TARGET. More formally, J_q is an approximation of the Wasserstein distance between $P_{\mathcal{F}}^{src}$ and $P_{\mathcal{F}}^{tgt}$ in (1).

For the sentiment classifier $\mathcal{P}$ parameterized by θ_p, we use the traditional cross-entropy loss, denoted as $L_p(\hat{y}, y)$, where $\hat{y}$ and y are the predicted label distribution and the true label, respectively. L_p is the negative log-likelihood that $\mathcal{P}$ predicts the correct label. We therefore seek the minimum of the following loss function for $\mathcal{P}$:

$$J_p(\theta_f) \equiv \min_{\theta_p} \mathop{\mathbb{E}}_{(x,y)} [L_p(\mathcal{P}(\mathcal{F}(x)), y)] \quad (3)$$

Finally, the joint feature extractor $\mathcal{F}$ parameterized by θ_f strives to minimize both the sentiment classifier loss J_p and $W(P_{\mathcal{F}}^{src}, P_{\mathcal{F}}^{tgt}) = J_q$:

$$J_f \equiv \min_{\theta_f} J_p(\theta_f) + \lambda J_q(\theta_f) \quad (4)$$

where λ is a hyper-parameter that balances the two branches $\mathcal{P}$ and $\mathcal{Q}$.

As proved by Arjovsky et al. (2017) and observed in our experiments (§3.3.5), minimizing the Wasserstein distance is much more stable w.r.t. hyperparameter selection compared to ADAN-GRL, saving the hassle of carefully varying λ during training (Ganin and Lempitsky, 2015). In addition, ADAN-GRL needs to laboriously coordinate the alternating training of the two competing components by setting the hyperparameter k, which indicates the number of iterations one component is trained before training the other. The performance can degrade substantially if k is not properly set. In our case, however, delicate tuning of k is no longer necessary since $W(P_{\mathcal{F}}^{src}, P_{\mathcal{F}}^{tgt})$ is approximated by maximizing (2); thus, training $\mathcal{Q}$ to optimum using a large k can provide better performance (but is slower to train). In our experiments, we fix $\lambda = 0.1$ and $k = 5$ for all experiments (train 5 $\mathcal{Q}$ iterations per $\mathcal{F}$ and $\mathcal{P}$ iteration), and the performance is stable over a large set of hyperparameters (§3.3.5).

ADAN training is depicted in Algorithm 1.

[5] A function g is 1-Lipschitz iff $|g(x) - g(y)| \le |x - y|$ for all x and y.

Methodology	Approach	Accuracy	
		Chinese	Arabic
Train-on-SOURCE-only	Logistic Regression	30.58%	45.83%
	DAN	29.11%	48.00%
Domain Adaptation	mSDA (Chen et al., 2012)	31.44%	48.33%
Machine Translation	Logistic Regression + MT	34.01%	51.67%
	DAN + MT	39.66%	52.50%
CLD-based CLTC	CLD-KCNN (Xu and Yang, 2017)	40.96%	52.67%†
	CLDFA-KCNN (Xu and Yang, 2017)	41.82%	53.83%†
Ours	ADAN	**42.49%** ±0.19%	**54.54%** ±0.34%

† As Xu and Yang (2017) did not report results for Arabic, these numbers are obtained based on our reproduction using their code.

Table 1: ADAN performance for Chinese (5-cls) and Arabic (3-cls) sentiment classification without using labeled TARGET data. All systems but the CLD ones use BWE to map SOURCE and TARGET words into the same space. CLD-based CLTC represents cross-lingual text classification methods based on cross-lingual distillation (Xu and Yang, 2017) and is explained in §3.2. For ADAN, average accuracy and standard errors over five runs are shown. Bold numbers indicate statistical significance over all baseline systems with $p < 0.05$ under a One-Sample T-Test. As a comparison, the supervised *English* accuracy of our ADAN model is 58.7% (5-class) and 75.6% (3-class).

3 Experiments and Discussions

To demonstrate the effectiveness of our model, we experiment on Chinese and Arabic sentiment classification, using English as SOURCE for both. For all data used in experiments, tokenization is done using Stanford CoreNLP (Manning et al., 2014).

3.1 Data

Labeled English Data. We use a balanced dataset of $700k$ Yelp reviews from Zhang et al. (2015) with their ratings as labels (scale 1-5). We also adopt their train-validation split: $650k$ reviews for training and $50k$ form a validation set.

Labeled Chinese Data. Since ADAN does not require labeled Chinese data for training, this annotated data is solely used to validate the performance of our model. $10k$ balanced Chinese hotel reviews from Lin et al. (2015) are used as validation set for model selection and parameter tuning. The results are reported on a separate test set of another $10k$ hotel reviews. For Chinese, the data are annotated with 5 labels (1-5).

Unlabeled Chinese Data. For the unlabeled TARGET data used in training ADAN, we use another $150k$ unlabeled Chinese hotel reviews.

English-Chinese Bilingual Word Embeddings. For Chinese, we used the pre-trained bilingual word embeddings (BWE) by Zou et al. (2013). Their work provides 50-dimensional embeddings for $100k$ English words and another set of $100k$ Chinese words. See §3.3.3 for more experiments and discussions.

Labeled Arabic Data. We use the BBN Arabic Sentiment Analysis dataset (Mohammad et al., 2016) for Arabic sentiment classification. The dataset contains 1200 sentences (600 validation + 600 test) from social media posts annotated with 3 labels ($-$, 0, $+$). The dataset also provides machine translated text to English. Since the label set does not match with the English dataset, we map all the rating 4 and 5 English instances to $+$ and the rating 1 and 2 instances to $-$, while the rating 3 sentences are converted to 0.

Unlabeled Arabic Data. For Arabic, no additional unlabeled data is used. We only use the text from the validation set (without labels) during training.

English-Arabic Bilingual Word Embeddings. For Arabic, we train a 300d BilBOWA BWE (Gouws et al., 2015) on the United Nations corpus (Ziemski et al., 2016).

3.2 Cross-Lingual Sentiment Classification

Our main results are shown in Table 1, which shows very similar trends for Chinese and Arabic. Before delving into discussions on the performance of ADAN compared to various baseline systems in the following paragraphs, we begin by clarifying the bilingual resources used in all the methods. Note first that in all of our experiments,

traditional features like bag of words cannot be directly used since SOURCE and TARGET have completely different vocabularies. Therefore, unless otherwise specified, BWEs are used as the input representation for all systems to map words from both SOURCE and TARGET into the same feature space. (The only exceptions are the CLD-based CLTC systems of Xu and Yang (2017) explained later in this section, which directly make use of a parallel corpus instead of relying on BWEs.) The same BWEs are adopted in all systems that utilize BWEs.

Train-on-SOURCE-only baselines We start by considering two baselines that train only on the SOURCE language, English, and rely solely on the BWEs to classify the TARGET. The first variation uses a standard supervised learning algorithm, Logistic Regression (LR), shown in Row 1 in Table 1. In addition, we evaluate a non-adversarial variation of ADAN, just the DAN portion of our model (Row 2), which is one of the modern neural models for sentiment classification. We can see from Table 1 that, in comparison to ADAN (bottom line), the train-on-SOURCE-only baselines perform poorly. This indicates that BWEs by themselves do not suffice to transfer knowledge of English sentiment classification to TARGET.

Domain Adaptation baselines We next compare ADAN with domain adaptation baselines, since domain adaptation can be viewed as a generalization of the cross-lingual task. Nonetheless, the divergence between languages is much more significant than the divergence between two domains, which are typically two product categories in practice. Among domain adaptation methods, the widely-used TCA (Pan et al., 2011) did not work since it required quadratic space in terms of the number of samples (650k). We thus compare to mSDA (Chen et al., 2012), a very effective method for cross-domain sentiment classification on Amazon reviews. However, as shown in Table 1 (Row 3), mSDA did not perform competitively. We speculate that this is because many domain adaptation models including mSDA were designed for the use of bag-of-words features, which are ill-suited in our task where the two languages have completely different vocabularies. In summary, this suggests that even strong domain adaptation algorithms cannot be used out of the box with BWEs for the CLSC task.

Machine Translation baselines We then evaluate ADAN against Machine Translation baselines (Rows 4-5) that (1) translate the TARGET text into English and then (2) use the better of the train-on-SOURCE-only models for sentiment classification. Previous studies (Banea et al., 2008; Salameh et al., 2015) on sentiment classification for Arabic and European languages claim this MT approach to be very competitive and find that it can sometimes match the state-of-the-art system trained on that language. For Chinese, where translated text was not provided, we use the commercial Google Translate engine[6], which is highly engineered, trained on enormous resources, and arguably one of the best MT systems currently available. As shown in Table 1, our ADAN model substantially outperforms the MT baseline on both languages, indicating that our adversarial model can successfully perform cross-lingual sentiment classification without any annotated data in the target language.

Cross-lingual Text Classification baselines Finally, we conclude ADAN's effectiveness by comparing against a state-of-the-art cross-lingual text classification (CLTC) method (Xu and Yang, 2017), as sentiment classification is one type of text classification. They propose a cross-lingual distillation (CLD) method that makes use of soft SOURCE predictions on a parallel corpus to train a TARGET model (CLD-KCNN). They further propose an improved variant (CLDFA-KCNN) that utilizes adversarial training to bridge the *domain* gap between the labeled and unlabeled texts within the source and the target language, similar to the adversarial domain adaptation by Ganin et al. (2016). In other words, CLDFA-KCNN consists of three conceptual adaptation steps: (i) Domain adaptation from source-language labeled texts to source-language unlabeled texts using adversarial training; (ii) Cross-lingual adaptation using distillation; and (iii) Domain adaptation in the target language from unlabeled texts to the test set. Note, however, Xu and Yang (2017) use adversarial training for domain adaptation within a single language vs. our work that uses adversarial training directly for cross-lingual generalization.

As shown in Table 1, ADAN significantly outperforms both variants of CLD-KCNN and

[6] https://translate.google.com

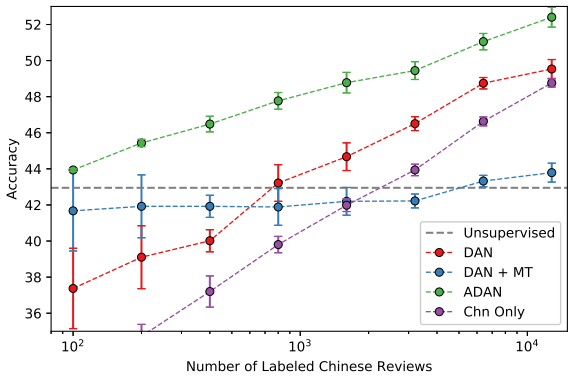

Figure 2: ADAN performance and standard deviation for Chinese in the semi-supervised setting when using various amount of labeled Chinese data.

achieves a new state of the art performance, indicating that our direct use of adversarial neural nets for cross-lingual adaptation can be more effective than chaining three adaptation steps as in CLDFA-KCNN. This is the case in spite of the fact that ADAN does not explicitly separate language variation from domain variation. In fact, the monolingual data we use for the source and target languages is indeed from different domains. ADAN's performance suggests that it could potentially bridge the divergence introduced by both sources of variation in one shot.

Supervised SOURCE accuracy By way of comparison, it is also instructive to compare ADAN's "transferred" accuracy on the TARGET with its (supervised) performance on the SOURCE. As shown in the caption of Table 1, ADAN achieves 58.7% accuracy on English for the 5-class English-Chinese setting, and 75.6% for the 3-class English-Arabic setting. The SOURCE accuracy for the DAN baselines (Rows 2 and 5) is similar to the SOURCE accuracy of ADAN.

3.3 Analysis and Discussion

Since the Arabic dataset is small, we choose Chinese as an example for our further analysis.

3.3.1 Semi-supervised Learning

In practice, it is usually not very difficult to obtain at least a small amount of annotated data. ADAN can be readily adapted to exploit such extra labeled data in the target language, by letting those labeled instances pass through the sentiment classifier $\mathcal{P}$ as the English samples do during training. We simulate this semi-supervised scenario by adding labeled Chinese reviews for training. We start by adding 100 labeled reviews and keep doubling the number until 12800. As shown in Figure 2, when adding the same number of labeled reviews, ADAN can better utilize the extra supervision and outperform the DAN baseline trained with combined data, as well as the supervised DAN using only labeled Chinese reviews. The margin is naturally decreasing as more supervision is incorporated, but ADAN is still superior when adding 12800 labeled reviews. On the other hand, the DAN with translation baseline seems unable to effectively utilize the added supervision in Chinese, and the performance only starts to show a slightly increasing trend when adding 6400 or more labeled reviews. One possible reason is that when adding to the training data a small number of English reviews translated from the labeled Chinese data, the training signals they produce might be lost in the vast number of English training samples, and thus not effective in improving performance. Another potentially interesting find is that it seems a very small amount of supervision (e.g. 100 labels) could significantly help DAN. However, with the same number of labeled reviews, ADAN still outperforms the DAN baseline.

3.3.2 Qualitative Analysis and Visualizations

To qualitatively demonstrate how ADAN bridges the distributional discrepancies between English and Chinese instances, t-SNE (Van der Maaten and Hinton, 2008) visualizations of the activations at various layers are shown in Figure 3. We randomly select 1000 reviews from the Chinese and English validation sets respectively, and plot the t-SNE of the hidden node activations at three locations in our model: the averaging layer, the end of the joint feature extractor, and the last hidden layer in the sentiment classifier just prior to softmax. The train-on-English model is the DAN baseline in Table 1. Note that there is actually only one "branch" in this baseline model, but in order to compare to ADAN, we conceptually treat the first three layers as the feature extractor.

Figure 3a shows that BWEs alone do not suffice to bridge the gap between the distributions of the two languages. To shed more light on the surprisingly clear separation given that individual words have a mixed distribution in both languages (not shown in figure), we first try to isolate the content divergence from the language divergence. In particular, the English and Chinese reviews are not

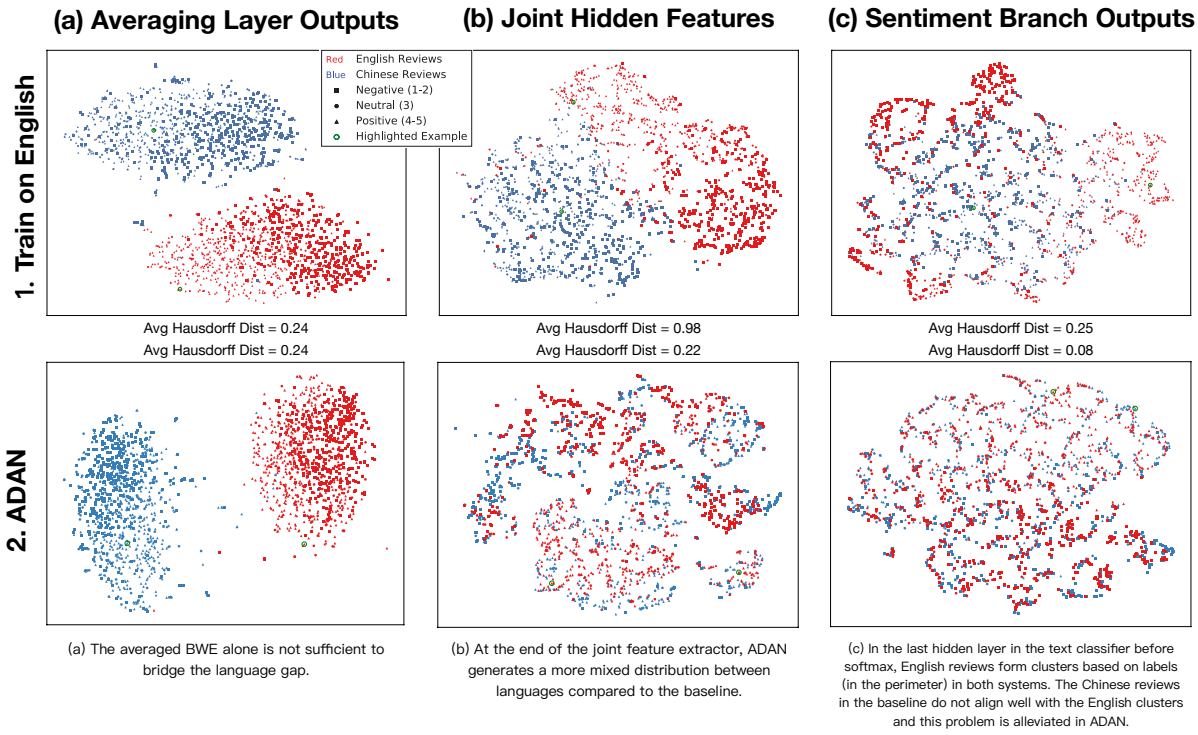

Figure 3: t-SNE visualizations of activations at various layers for the train-on-SOURCE-only baseline model (top) and ADAN (bottom). The distributions of the two languages are brought much closer in ADAN as they are represented deeper in the network (left to right) measured by the Averaged Hausdorff Distance (see text). The green circles are two 5-star example reviews (shown below the figure) that illustrate how the distribution evolves (zoom in for details).

translations of each other, and in fact may even come from different domains. Therefore, the separation could potentially come from two sources: the content divergence between the English and Chinese reviews, and the language divergence of how words are used in the two languages. To control for content divergence, we tried plotting (not shown in figure) the average word embeddings of 1000 random Chinese reviews and their machine translations into English using t-SNE, and surprisingly the clear separation was still present. There are a few relatively short reviews that reside close to their translations, but the majority still form two language islands. (The same trend persists when we switch to a different set of pre-trained BWEs, and when we plot a similar graph for English-Arabic.) When we remove stop words (the most frequent word types in both languages), the two islands finally start to become slightly closer with less clean boundaries, but the separation remains clear. We think this phenomenon is interesting, and a thorough investigation is out of the scope of this work. We hypothesize that at least in certain distant language pairs such as English-Chinese[7], the divergence between languages may not only be determined by word semantics, but also largely depends on how words are used.

Furthermore, we can see in Figure 3b that the distributional discrepancies between Chinese and English are significantly reduced after passing through the joint feature extractor ($\mathcal{F}$). The learned features in ADAN bring the distributions in the two languages dramatically closer compared to the monolingually trained baseline. This is shown via the Averaged Hausdorff Distance (AHD, Shapiro and Blaschko, 2004), which measures the distance between two sets of points.

[7] In a personal correspondence with Ahmed Elgohary, he did not observe the same phenomenon between English and French.

Model	Random	BilBOWA	Zou et al.
DAN	21.66%	28.75%	29.11%
DAN+MT	37.78%	38.17%	39.66%
ADAN	34.44%	40.51%	42.95%

Table 2: Model performance on Chinese with various (B)WE initializations.

The AHD between the English and Chinese reviews is provided for all sub-plots in Figure 3.

Finally, when looking at the last hidden layer activations in the sentiment classifier of the baseline model (Figure 3c), there are several notable clusters of red dots (English data) that roughly correspond to the class labels. These English clusters are the areas where the classifier is the most confident in making decisions. However, most Chinese samples are not close to one of those clusters due to the distributional divergence and may thus cause degraded classification performance in Chinese. On the other hand, the Chinese samples are more in line with the English ones in ADAN, which results in the accuracy boost over the baseline model. In Figure 3, a pair of similar English and Chinese 5-star reviews is highlighted to visualize how the distribution evolves at various points of the network. We can see in 3c that the highlighted Chinese review gets close to the "positive English cluster" in ADAN, while in the baseline, it stays away from dense English clusters where the sentiment classifier trained on English data is not confident to make predictions.

3.3.3 Impact of Bilingual Word Embeddings

In this section we discuss the effect of the bilingual word embeddings. We start by initializing the systems with random word embeddings (WEs), shown in Table 2. ADAN with random WEs outperforms the DAN and mSDA baselines using BWEs and matches the performance of the LR+MT baseline (Table 1), suggesting that ADAN successfully extracts features that could be used for cross-lingual classification tasks without *any* bitext. This impressive result vindicates the power of adversarial training to reduce the distance between two complex distributions without any direct supervision, which is also observed in other recent works for different tasks (Zhang et al., 2017; Lample et al., 2018).

With the introduction of BWEs (Column 2 and 3), the performance of ADAN is further boosted.

Model	Accuracy	Run time
DAN	42.95%	0.127 (s/iter)
CNN	46.24%	0.554 (s/iter)
BiLSTM	44.55%	1.292 (s/iter)
BiLSTM + dot attn	46.41%	1.898 (s/iter)

Table 3: Performance and speed for various feature extractor architectures on Chinese.

Therefore, it seems the quality of the BWEs plays an important role in CLSC. To investigate the impact of the specific choice of BWEs, we also trained 100d BilBOWA BWEs (Gouws et al., 2015) using the UN parallel corpus for Chinese. All systems achieve slightly lower performance compared to the pre-trained BWEs from Zou et al. (2013), yet ADAN still outperforms other baseline methods (Table 2), demonstrating that ADAN's effectiveness is relatively robust with respect to the choice of BWEs. We conjecture that all systems show inferior results with BilBOWA, because it does not require word alignments during training as Zou et al. (2013) do. By only training on a sentence-aligned corpus, BilBOWA requires less resources and is much faster to train, potentially at the expense of quality.

3.3.4 Feature Extractor Architectures

As mentioned in §2.1, the architecture of ADAN's feature extractor is not limited to a Deep Averaging Network (DAN), and one can choose different feature extractors to suit a particular task or dataset. While an extensive study of alternative architectures is beyond the scope of this work, we in this section present a brief experiment illustrating that our adversarial framework works well with other $\mathcal{F}$ architectures. In particular, we consider two popular choices: i) a CNN (Kim, 2014) that has a 1d convolutional layer followed by a single fully-connected layer to extract a fixed-length vector; and ii) a Bi-LSTM with two variants: one that takes the average of the hidden outputs of each token as the feature vector, and one with the dot attention mechanism (Luong et al., 2015) that learns a weighted linear combination of all hidden outputs.

As shown in Table 3, ADAN's performance can be improved by adopting more sophisticated feature extractors, at the expense of slower running time. This demonstrates that ADAN's language-adversarial training framework can be successfully

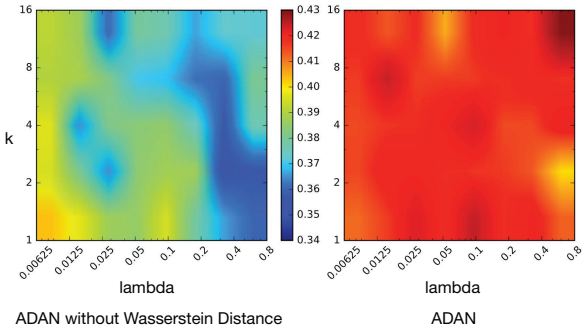

Figure 4: A grid search on k and $lambda$ for ADAN (right) and the ADAN-GRL variant (left). Numbers indicate the accuracy on the Chinese development set.

used with other $\mathcal{F}$ choices.

3.3.5 ADAN Hyperparameter Stability

In this section, we show that the training of ADAN is stable over a large set of hyperparameters, and provides improved performance compared to the standard ADAN-GRL.

To verify the superiority of ADAN, we conduct a grid search over k and λ, which are the two hyperparameters shared by ADAN and ADAN-GRL. We experiment with $k \in \{1, 2, 4, 8, 16\}$, and $\lambda \in \{0.00625, 0.0125, 0.025, 0.05, 0.1, 0.2, 0.4, 0.8\}$. Figure 4 reports the accuracy on the Chinese dev set for both ADAN variants, and shows higher accuracy and greater stability over the Ganin and Lempitsky (2015) variant. This suggests that ADAN overcomes the well-known problem that adversarial training is sensitive to hyperparameter tuning.

3.4 Implementation Details

For all our experiments on both languages, the feature extractor $\mathcal{F}$ has three fully-connected layers with ReLU non-linearities, while both $\mathcal{P}$ and $\mathcal{Q}$ have two. All hidden layers contain 900 hidden units. Batch Normalization (Ioffe and Szegedy, 2015) is used in each hidden layer in $\mathcal{P}$ and $\mathcal{Q}$. $\mathcal{F}$ does not use batch normalization. $\mathcal{F}$ and $\mathcal{P}$ are optimized jointly using Adam (Kingma and Ba, 2015) with a learning rate of 0.0005. $\mathcal{Q}$ is trained with another Adam optimizer with the same learning rate. The weights of $\mathcal{Q}$ are clipped to $[-0.01, 0.01]$. We train ADAN for 30 epochs and use early stopping to select the best model on the validation set. ADAN is implemented in PyTorch (Paszke et al., 2017).

4 Related Work

Cross-lingual Sentiment Classification is motivated by the lack of high-quality labeled data in many non-English languages (Bel et al., 2003; Mihalcea et al., 2007; Banea et al., 2008, 2010; Soyer et al., 2015). For Chinese and Arabic in particular, there are several representative works (Wan, 2008, 2009; He et al., 2010; Lu et al., 2011; Mohammad et al., 2016). Our work is comparable to these in objective but very different in method. The work by Wan uses machine translation to directly convert English training data to Chinese; this is one of our baselines. Lu et al. (2011) instead uses labeled data from both languages to improve the performance on both. Other papers make direct use of a parallel corpus either to learn a bilingual document representation (Zhou et al., 2016) or to conduct cross-lingual distillation (Xu and Yang, 2017). Zhou et al. (2016) require the translation of the entire English training set which is prohibitive for our setting, while ADAN outperforms Xu and Yang (2017)'s approach in our experiments.

Domain Adaptation tries to learn effective classifiers for which the training and test samples are from different underlying distributions (Blitzer et al., 2007; Pan et al., 2011; Glorot et al., 2011; Chen et al., 2012; Liu et al., 2015). This can be thought of as a generalization of cross-lingual text classification. However, one main difference is that, when applied to text classification tasks, most of these domain adaptation work assumes a common feature space such as a bag-of-words representation, which is not available in the cross-lingual setting. See Section 3.2 for experiments on this. In addition, most works in domain adaptation evaluate on adapting product reviews across domains (e.g. books to electronics), where the divergence in distribution is less significant than that between two languages.

Adversarial Networks have enjoyed much success in computer vision (Goodfellow et al., 2014; Ganin et al., 2016). A series of work in image generation has used architectures similar to ours, by pitting a neural image generator against a discriminator that learns to classify real versus generated images (Goodfellow et al., 2014). More relevant to this work, adversarial architectures have produced the state-of-the-art in unsupervised domain adaptation for image object recognition: Ganin et al. (2016) train with many labeled source images and unlabeled target images, similar to our

setup. In addition, other recent work (Arjovsky et al., 2017; Gulrajani et al., 2017) proposes improved methods for training Generative Adversarial Nets. ADAN proposes *language-adversarial training*, the first adversarial neural net for cross-lingual NLP (Chen et al., 2016). As of the writing of this journal paper, there are several other recent works that adopt adversarial training for cross-lingual NLP tasks, such as cross-lingual text classification (Xu and Yang, 2017), cross-lingual word embedding induction (Zhang et al., 2017; Lample et al., 2018) and cross-lingual question similarity reranking (Joty et al., 2017).

5 Conclusion and Future Work

In this work, we presented ADAN, an adversarial deep averaging network for cross-lingual sentiment classification. ADAN leverages the abundant labeled resources from English to help sentiment classification on other languages where little or no annotated data exist. We validate ADAN's effectiveness by experiments on Chinese and Arabic sentiment classification, where we have labeled English data and only *unlabeled* data in the target language. Experiments show that ADAN outperforms several baselines including domain adaptation models, a competitive MT baseline, and state-of-the-art cross-lingual text classification methods. We further show that even without *any* bilingual resources, ADAN trained with randomly initialized embeddings can still achieve encouraging performance. In addition, we show that in the presence of labeled data in the target language, ADAN can naturally incorporate this additional supervision and yields even more competitive results.

For future work, we plan to apply our language-adversarial training framework to other NLP adaptation tasks where explicit MLE training is not feasible due to the lack of direct supervision. Our framework is not limited to sentiment classification or even to generic text classification: It can be applied, for example, to phrase-level tagging tasks (İrsoy and Cardie, 2014) where labeled data might not exist for certain languages. In another direction, we can look beyond a single SOURCE and TARGET language and utilize our adversarial training framework for multi-lingual text classification.

Acknowledgments

We thank the anonymous reviewers and members of Cornell NLP and ML groups for helpful comments. This work was funded in part by a grant from the DARPA Deft Program.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 214–223.

Carmen Banea, Rada Mihalcea, and Janyce Wiebe. 2010. Multilingual subjectivity: Are more languages better? In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 28–36.

Carmen Banea, Rada Mihalcea, Janyce Wiebe, and Samer Hassan. 2008. Multilingual subjectivity analysis using machine translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 127–135.

Nuria Bel, Cornelis H. A. Koster, and Marta Villegas. 2003. Cross-lingual text categorization. In *Research and Advanced Technology for Digital Libraries*, pages 126–139, Berlin, Heidelberg.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447.

Minmin Chen, Zhixiang Xu, Kilian Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 767–774, New York, NY, USA.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2016. Adversarial deep averaging networks for cross-lingual sentiment classification. *ArXiv e-prints 1606.01614v4*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning*.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 513–520, New York, NY, USA.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BilBOWA: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning*.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. 2017. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems 30*, pages 5767–5777.

Yulan He, Harith Alani, and Deyu Zhou. 2010. Exploring English lexicon knowledge for Chinese sentiment analysis. In *CIPS-SIGHAN Joint Conference on Chinese Language Processing*.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of The 32nd International Conference on Machine Learning*.

Ozan İrsoy and Claire Cardie. 2014. Opinion mining with deep recurrent neural networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 720–728.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1681–1691.

Shafiq Joty, Preslav Nakov, Lluís Màrquez, and Israa Jaradat. 2017. Cross-language learning with adversarial neural networks: Application to community question answering. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 226–237, Vancouver, Canada.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Yiou Lin, Hang Lei, Jia Wu, and Xiaoyu Li. 2015. An empirical study on sentiment classification of chinese review using word embedding. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation: Posters*, pages 258–266.

Biao Liu, Minlie Huang, Jiashen Sun, and Xuan Zhu. 2015. Incorporating domain and sentiment supervision in representation learning for domain adaptation. In *International Joint Conference on Artificial Intelligence*.

Bin Lu, Chenhao Tan, Claire Cardie, and Benjamin K. Tsou. 2011. Joint bilingual sentiment classification with unlabeled parallel cor-

pora. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 320–330.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Rada Mihalcea, Carmen Banea, and Janyce Wiebe. 2007. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 976–983.

Saif M. Mohammad, Mohammad Salameh, and Svetlana Kiritchenko. 2016. How translation alters sentiment. *Journal of Artificial Intelligence Research*, 55(1):95–130.

Sinno J. Pan, Ivor W. Tsang, James T. Kwok, and Qiang Yang. 2011. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS 2017 Autodiff Workshop*.

Mohammad Salameh, Saif Mohammad, and Svetlana Kiritchenko. 2015. Sentiment after translation: A case-study on arabic social media posts. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 767–777.

Michael D Shapiro and Matthew B Blaschko. 2004. On hausdorff distance measures. Technical report, Technical Report UM-CS-2004-071.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, D. Christopher Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Hubert Soyer, Pontus Stenetorp, and Akiko Aizawa. 2015. Leveraging monolingual data for crosslingual compositional word representations. In *International Conference on Learning Representations*.

Sheng Kai Tai, Richard Socher, and D. Christopher Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394.

Cédric Villani. 2008. *Optimal transport: old and new*, volume 338. Springer Science & Business Media.

Xiaojun Wan. 2008. Using bilingual knowledge and ensemble techniques for unsupervised chinese sentiment analysis. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 553–561.

Xiaojun Wan. 2009. Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, pages 235–243, Stroudsburg, PA, USA.

Ruochen Xu and Yiming Yang. 2017. Cross-lingual distillation for text classification. In

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1415–1425.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28*, pages 649–657.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016. Cross-lingual sentiment classification with bilingual document representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1403–1412.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus. In *Language Resources and Evaluation (LREC'16)*.

Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398, Seattle, Washington, USA.