Knowl Inf Syst (2016) 47:27–44
DOI 10.1007/s10115-015-0849-0

REGULAR PAPER

Cross-lingual sentiment classification with stacked autoencoders

Guangyou Zhou[1] · Zhiyuan Zhu[2] ·
Tingting He[1] · Xiaohua Tony Hu[1,3]

Received: 8 May 2014 / Revised: 15 March 2015 / Accepted: 30 May 2015 /
Published online: 11 June 2015

Abstract Cross-lingual sentiment classification is a popular research topic in natural language processing. The fundamental challenge of cross-lingual learning stems from a lack of overlap between the feature spaces of the source language data and the target language data. In this article, we propose a new model which uses stacked autoencoders to learn language-independent high-level feature representations for the both languages in an unsupervised fashion. The proposed framework aims to force the aligned input bilingual sentences into a common latent space, and the objective function is defined by minimizing the input and output vector representations as well as the distance of the common representations in the latent space. Sentiment classifiers trained on the source language can be adapted to predict sentiment polarity of the target language with the language-independent high-level feature representations. We conduct extensive experiments on English–Chinese sentiment classification tasks of multiple data sets. Our experimental results demonstrate the efficacy of the proposed cross-lingual approach.

Keywords Sentiment classification · Cross-lingual · Stacked autoencoder

1 Introduction

With the development of Web 2.0, more and more user-generated sentiment data have been shared on the Web. They exist in the form of user reviews on shopping or opinion sites, in posts of blogs or customer feedback in different languages. These labelled user-generated sentiment data are considered as the most valuable resources for the sentiment classification task. Sentiment classification is the task of predicting the sentiment polarity of a given text.

✉ Guangyou Zhou
gyzhou@mail.ccnu.edu.cn

[1] School of Computer, Central China Normal University, Wuhan 430079, China

[2] Chinese Institute of Electronics, Beijing 100036, China

[3] College of Computing and Informatics, Drexel University, Philadelphia, PA 19104, USA

The sentiment polarity is usually positive or negative. Over the past years, sentiment classification has drawn much attention in the natural language processing (NLP) filed. However, these sentiment resources in different languages are very imbalanced. Manually labelling each individual language is a time-consuming and labour-intensive job, which makes cross-lingual sentiment classification essential for this application.

Cross-lingual sentiment classification aims to automatically predict sentiment polarity (e.g. positive or negative) of data in a label-scarce target language by exploiting labelled data from a label-rich language. The fundamental challenge of cross-lingual learning stems from a lack of overlap between the feature spaces of the source language data and that of the target language data. To address this challenge, previous work in the literature mainly relies on machine translation engines or bilingual lexicons to directly adapt labelled data from the source language to the target language [23,28,31,39,40,42,44]. Although the machine translation-based approaches are intuitive and have advanced the task of cross-lingual sentiment classification, they have certain limitations [47]. First, machine translation may change the sentiment polarity of the original data [17]. For example, the negative English sentence “it is too beautiful to be true” is translated to a positive sentence in Chinese “实在是太漂亮是真实的” by Google Translate (http://translate.google.com/), which literally means “it is too beautiful and true”. Second, many sentiment indicative words cannot be learned from the translated labelled data due to the limited coverage of vocabulary in the machine translation results. Recently, Duh et al. [5] report a low overlap between the vocabulary of English documents and the documents translated from Japanese to English, and the experiments also show that vocabulary coverage has a strong correlation with sentiment classification accuracy. Third, translating all the sentiment data in one language into the other language is a time-consuming and labour-intensive job in reality.

In this article, we propose a deep learning approach, which uses stacked autoencoders [45] to learn language-independent high-level semantic representations of data for cross-lingual sentiment classification. Our model is firstly trained on a large-scale bilingual parallel data and then projects the source language and the target language into a common space that fuses the two types of information together. The goal of our model is to learn multiple levels of representations through a hierarchy of network architectures, where higher-level representations (e.g. the representation $\hat{\mathbf{y}}$ in Fig. 1) can be used to bridge the gap between the source language and the target language. For example, if we have learned higher-level concepts for representing English and Chinese sentiment data, then a classifier trained on labelled English sentiment data can be used to classify Chinese sentiment data (provided we use the learned common higher-level concepts for representing documents in both languages).

The novelty of our approach lies in that we employ a deep learning approach to project the source language and the target language into a language-independent unified representations. Our work shares certain intuition with the mixture model [17] and the bilingual word embeddings used in cross-lingual sentiment classification [10] and phrase-based machine translation [50]. A common property of these approaches is that a word-level alignment (extracted using GIZA++ [21]) of bilingual parallel corpus is leveraged [10,14,17,50]. In this article, we only require aligned parallel documents and do not rely on word-level alignments of bilingual corpus during training, which simplifies the learning procedure.

To evaluate the effectiveness of the proposed approach, we conduct experiments on the task of English–Chinese cross-lingual sentiment classification as a case study. Our experimental results show that the proposed framework is better than the baseline methods and achieves the comparable performance with the state-of-the-art methods.

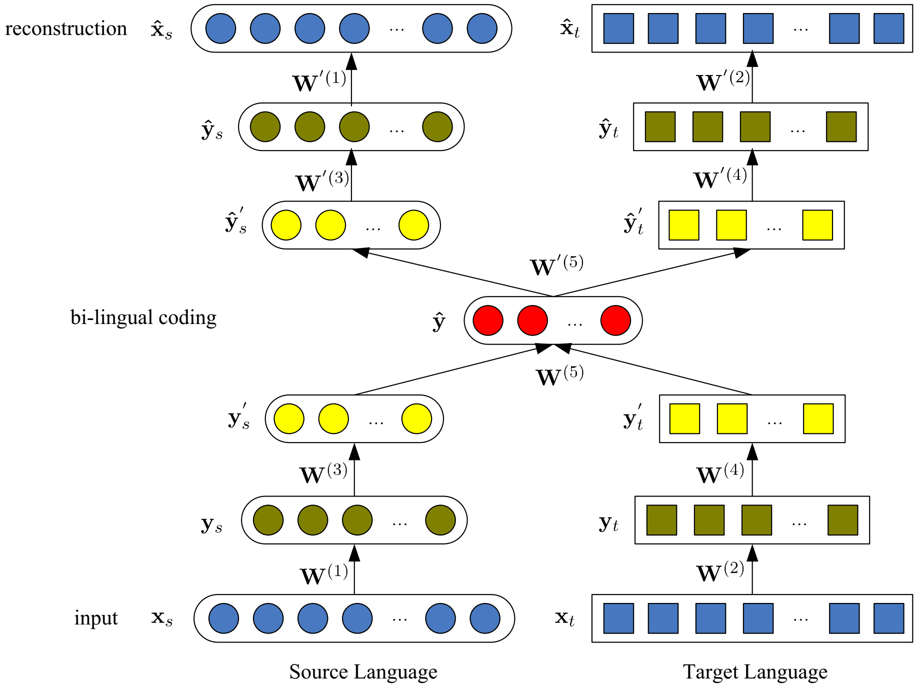

Fig. 1 Stacked autoencoder trained on large-scale parallel sentence pairs. Input to the model is binary bag-of-words vector representations obtained from the source language and the target language

The rest of this article is organized as follows. Section 2 introduces the related work. Section 3 describes the problem settings. Section 4 presents our proposed deep learning method for cross-lingual sentiment classification. Section 5 presents the experimental results. In Sect. 6, we conclude the article and discuss future research directions.

2 Related work

In this section, we present the related work on traditional monolingual sentiment classification and cross-lingual sentiment classification. We divide this part into two subsections because monolingual and cross-lingual sentiment classifications share certain properties, but they have their own methodology.

2.1 Monolingual sentiment classification

Sentiment classification has gained wide interest in NLP community. Methods for automatically classifying sentiment expressed in products and movie reviews can roughly be divided into supervised and unsupervised (or semi-supervised) sentiment analysis. Supervised techniques have been proved promising and widely used in sentiment classification [13,24,25]. Pang et al. [25]'s was the first paper to take this approach to classify movie reviews int two classes, positive and negative. It was shown that using bag-of-words as features in classification performed quite well with either naive Bayes or SVM, although the authors also tried a number of other features.

In subsequent research, many more features and learn algorithms were tried by a large number of researchers. In [26], the minimum cut algorithm working on a graph was employed to help sentiment classification. In [20], the classification was done by using some linguistic knowledge sources. In [43], syntactic relations were used together with traditional features. In [9], different IR term weighting schemes were studied and compared for sentiment classification. In [19], a dependency tree-based classification method was proposed, which used conditional random fields (CRF) [11] with hidden variables. In [15], the authors used word vectors which can capture some latent aspects of the words to help classification. In [2], sentiment classification was performed based on supervised latent n-gram analysis. In [16], the authors explored various feature definition and selection strategies. However, the performance of these methods relies on manually labelled training data. In some cases, the labelling work may be time-consuming and expensive. This motivates the problem of learning robust sentiment classification via unsupervised (or semi-supervised) paradigm.

The most representative way to perform semi-supervised paradigm is to employ partial labelled data to guide the sentiment classification [6,12,37,48]. The goal of the semi-supervised scheme is to learn from few labelled documents by making use of the unlabelled data. Empirical study showed that simultaneously attempting both these goals in a single model leads to improvements over models that focus on a single goal. Goldberg et al. [6] adapted semi-supervised graph-based methods for sentiment analysis but did not incorporate lexical prior knowledge in the form of features. Vikas and Prem [37] proposed a dual supervision model for semi-supervised sentiment analysis. In this model, bipartite graph regularization is used to diffuse label information along both sides of the term-document matrix. Li [12] proposed a active learning method to select documents. However, we do not have any labelled data at hand in many situations, which makes the unsupervised paradigm necessary. Zhou et al. [48] proposed a non-negative matrix factorization (NMF) framework with co-regularization for sentiment classification.

The most representative way to perform unsupervised paradigm is to use a sentiment lexicon to guide the sentiment classification [3,33,35] or learn sentiment orientation of a word from its semantically related words mined from the lexicon [27]. Sentiment polarity of a word is obtained from an existed sentiment lexicon; the overall sentiment polarity of a document is computed as the summation of sentiment scores of the words in the document. Turney [35] performed classification based on some fixed syntactic patterns that are likely to be used to express opinions. The syntactic patterns are composed based on part-of-speech (POS) tags. Taboada et al. [33] used a dictionary of sentiment words and phrases with their associated orientations and strength, and incorporates intensification and negation to compute a sentiment score for each document. Baccianella et al. [3] presented a lexical resource called SENTIWORDNET 3.0 for supporting sentiment classification. Peng and Park [27] presented an automatic sentiment dictionary generation method to assign polarity scores to each word in the dictionary on a large social media corpus. All these work focuses on monolingual sentiment classification; we point the readers to recent books [13,24] for an in-depth survey of literature on sentiment classification.

2.2 Cross-lingual sentiment classification

Cross-lingual sentiment classification aims to automatically predict sentiment polarity (e.g. positive or negative) of data in a label-scarce target language by exploiting labelled data from a label-rich language. The fundamental challenge of cross-lingual learning stems from a lack of overlap between the feature spaces of the source language data and that of the target language data.

To bridge the language gap, previous work in the literature mainly relies on machine translation engines or bilingual lexicons to directly adapt labelled data from the source language to the target language. Banea et al. [1] employed machine translation engines to bridge the language gap in different languages for multilingual subjectivity analysis. Wan [38,39] proposed to use ensemble methods to train Chinese sentiment classification model on English labelled data and their Chinese translations. English labelled data are first translated into Chinese, and then the bi-view (Chinese side and English side) sentiment classifiers are trained on English and Chinese labelled data, respectively. Pan et al. [23] proposed a bi-view (e.g. a view is from English data and another view is from Chinese data) joint non-negative matrix tri-factorization (BNMTF) model for cross-lingual sentiment classification problem. They employed machine translation engines so that both training and test data are able to have two representations, one in source language and the other in target language. The proposed model is derived from the NMF models in both languages in order to make more accurate prediction. Prettenhofer and Stein [28] proposed a cross-lingual structural correspondence learning (CL-SCL) method to induce language-independent features. Instead of using machine translation engines to translate labelled text, the authors first selected a subsect of pivot features in the source language to translate them into the target language, and then use these pivot pairs to induce cross-lingual representations by modelling the correlations between pivot features and non-pivot features in an unsupervised fashion. Recently, Xiao and Guo [44] performed representation learning in a semi-supervised manner by directly incorporating discriminative information with respect to the target prediction task. In this article, we propose a deep learning approach, which uses stacked autoencoders [45] to learn language-independent high-level semantic representations of data instead of machine translation engines.

Another group of works propose to use an unlabelled parallel corpus to induce language-independent representations [14,17,49]. They assume that parallel sentences in the corpus should have the same sentiment polarity and labelled data in both the source and target languages are available. However, this method requires labelled data in both the source and target language, which are not always readily available [17]. Meng et al. [17] proposed a generative cross-lingual mixture model (CLMM) to learn previously unseen sentiment words from the large bilingual parallel data. Zhou et al. [49] proposed a subspace learning framework with partially labelled parallel corpus for cross-lingual sentiment classification. A common property of this approach is that a word-level alignment (extracted using GIZA++) of bilingual parallel corpus is leveraged [17] or a shallow matrix factorization model is leveraged [49]. In this article, we only require aligned parallel documents and do not rely on word-level alignments of bilingual corpus during training, which simplifies the learning procedure.

3 Problem description and definitions

In this section, we first present some definitions and then give a formal definition of the problem we address in this article.

Definition 3.1 (*Vocabulary*) Let $\mathcal{V}_{\mathrm{s}}$ denote the vocabulary in the source language and $\mathcal{V}_{\mathrm{t}}$ denote the vocabulary in the target language. We also set $m_{\mathrm{s}} = |\mathcal{V}_{\mathrm{s}}|$ and $m_{\mathrm{t}} = |\mathcal{V}_{\mathrm{t}}|$, which denote the vocabulary size of the source language and the target language, respectively.

In this article, we focus on English–Chinese sentiment classification (e.g. English is the source language, while Chinese is the target language, or vice versa). However, our proposed approach is quite general and can be easily adapted to other language pairs (e.g. English–Japanese) sentiment classification problems. Because our framework can force the aligned sentences in both languages into the bilingual coding space, which is a language-independent latent space. Therefore, we can train a classification model on this latent space.

Definition 3.2 (*Sentiment document set*) Let $\mathcal{X}_{\mathrm{s}} = \{\mathbf{x}_{\mathrm{s}}^{(1)}, \ldots, \mathbf{x}_{\mathrm{s}}^{(n_{\mathrm{s}})}\}$ denote the sentiment document set in the source language, n_{s} is the number of documents in $\mathcal{X}_{\mathrm{s}}$, and $\mathbf{x}_{\mathrm{s}}^{(i)}$ is the binary bag-of-words representation of ith document. Similarly, let $\mathcal{X}_{\mathrm{t}} = \{\mathbf{x}_{\mathrm{t}}^{(1)}, \ldots, \mathbf{x}_{\mathrm{t}}^{(n_{\mathrm{t}})}\}$ denote the sentiment document set in the target language and n_{t} is the number of documents in $\mathcal{X}_{\mathrm{t}}$.

Definition 3.3 (*Sentiment label*) Let $\mathcal{Y}_{\mathrm{s}} = \{y_{\mathrm{s}}^{(1)}, \ldots, y_{\mathrm{s}}^{(n_{\mathrm{s}})}\}$ denote the sentiment label for the document set in the source language, In this article, we focus on binary sentiment classification, that is $y_{\mathrm{s}}^{(i)} \in \{+1, -1\}$. Similarly, we can define $\mathcal{Y}_{\mathrm{t}}$.

Besides positive and negative sentiment data described above, there are also neutral and mixed sentiment data in practical applications. Mixed sentiment data mean that the sentiment polarity of a document is positive in some aspects but negative in other ones. Neutral sentiment data mean that there is no sentiment expressed by users [22]. In this article, we only focus on positive and negative sentiment data, but it is not hard to extend the proposed method to address multi-class (e.g. positive, negative and neutral) sentiment classification problems.

Definition 3.4 (*Unlabelled parallel documents*) Let $\mathcal{X} = \{(\mathbf{x}_{\mathrm{s}}^{(1)}, \mathbf{x}_{\mathrm{t}}^{(1)}), \ldots, (\mathbf{x}_{\mathrm{s}}^{(n)}, \mathbf{x}_{\mathrm{t}}^{(n)})\}$ be a large amount of parallel documents; we would like to use it to learn language-independent representations in both languages that are aligned, such that pairs of translated words have similar representations.

Definition 3.5 *Problem definition* (*Cross-lingual sentiment classification*) Given the labelled sentiment data $\mathcal{X}_{\mathrm{s}}$ from a source language and the unlabelled sentiment data $\mathcal{X}_{\mathrm{t}}$ from a target language (sometimes, the partially labelled sentiment data $\mathcal{X}_{\mathrm{t}}$ from a target language can also be used), cross-lingual sentiment classification aims to learn a classifier on the source language $\mathcal{X}_{\mathrm{s}}$ and then predicts the polarity of unseen sentiment data of $\mathcal{X}_{\mathrm{t}}$.

In order to solve this problem, we propose a framework shown in Fig. 1 which targets to achieve two subtasks: (1) learning language-independent document representation and (2) projecting documents in the source language and the target language into a common space. In the first subtask, we aim to learn language-independent document representation using stacked autoencoders. We train the model on a large-scale parallel corpus in a unsupervised fashion. The goal of this subtask is to learn multiple levels of representations through a hierarchy of network architectures, where higher-level representations are expected to help define higher-level concepts. These learned language-independent representations are used as a bridge to reduce the language gap. In the second subtask, we aim to project the labelled sentiment data $\mathcal{X}_{\mathrm{s}}$ from the source language and the unlabelled data $\mathcal{X}_{\mathrm{t}}$ from the target language into a common space. With this unified representation, we can train a sentiment classifier on the labelled sentiment data and predict the sentiment polarity on the unseen data.

From the framework in Fig. 1, we can see that although language-independent representation learning computation is time-consuming, they can run once in advance. Meanwhile, the computational requirement of sentiment classifier training is limited. Here, the proposed framework can efficiently achieve the cross-lingual sentiment classification.

4 Our approach

4.1 Background

Our model aims to learn language-independent document representation in a unsupervised fashion. We first briefly describe autoencoders in this section with emphasis on aspects relevant to our model.

4.1.1 The basic autoencoder

An autoencoder is an unsupervised neural network which is trained to reconstruct a given input vector from its latent representation [46]. An autoencoder takes an input vector $\mathbf{x}^{(i)} \in [0, 1]^m$, and first maps it to a latent representation $\mathbf{y}^{(i)} \in [0, 1]^{m'}$ through a encoding function $\mathbf{y}^{(i)} = f_\theta(\mathbf{x}^{(i)}) = s(\mathbf{W}\mathbf{x}^{(i)} + \mathbf{b})$, parameterized by $\theta = \{\mathbf{W}, \mathbf{b}\}$. $\mathbf{W}$ is a $m' \times m$ weight matrix, and $\mathbf{b}$ is a bias vector. s is a nonlinear activation function, such as a sigmoid function or hyperbolic tangent. The latent representation $\mathbf{y}^{(i)}$ is then mapped back to a "reconstructed" vector $\hat{\mathbf{x}}^{(i)} \in [0, 1]^m$ in input space $\hat{\mathbf{x}}^{(i)} = g_{\theta'}(\mathbf{y}^{(i)}) = s(\mathbf{W}'\mathbf{y}^{(i)} + \mathbf{b}')$ with $\theta' = \{\mathbf{W}', \mathbf{b}'\}$. The weight matrix $\mathbf{W}'$ of the reverse mapping may optionally be constrained by $\mathbf{W}' = \mathbf{W}^T$, in which case the autoencoder is said to have tied weights [36,45]. The training objective is to learn the parameters $\hat{\theta} = \{\mathbf{W}, \mathbf{b}\}$ and $\hat{\theta}' = \{\mathbf{W}', \mathbf{b}'\}$ that minimize the average reconstruction error over a set of input vectors $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\}$:

$$\begin{aligned}\hat{\theta}, \hat{\theta}' &= \arg\max_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} L\left(\mathbf{x}^{(i)}, \hat{\mathbf{x}}^{(i)}\right) \\ &= \arg\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} L\left(\mathbf{x}^{(i)}, g_{\theta'}\left(f_\theta(\mathbf{x}^{(i)})\right)\right) \qquad (1)\end{aligned}$$

where L is a loss function, such as cross-entropy. Parameters θ and θ' can be optimized by stochastic or mini-batch gradient descent.

4.1.2 The denoising autoencoder

The procedure to train a deep network using the denoising autoencoder is similar to the basic autoencoder. The only difference is how each layer is trained. The training criterion with denoising autoencoders is to reconstruct a clean "repaired" input from a corrupted, partially destroyed one. This is done by first corrupting the initial input vector $\mathbf{x}^{(i)}$ to get a partially destroyed version $\widetilde{\mathbf{x}}^{(i)}$ [36]. The basic idea is that the learned latent representation is good if the autoencoder is capable of reconstructing the actual input from its corruption. The loss function of the reconstruction error for an input vector $\mathbf{x}^{(i)}$ becomes:

$$L\left(\mathbf{x}^{(i)}, g_{\theta'}\left(f_\theta\left(\widetilde{\mathbf{x}}^{(i)}\right)\right)\right) \qquad (2)$$

Following the literature [36], we consider one possible corrupting process, parameterized by the desired proportion ν of "destruction": for each input vector $\mathbf{x}^{(i)}$, a fixed number of νm of components are chosen at random, and their values are forced to 0, while the others are left untouched.

4.1.3 The stack autoencoder

Several autoencoders can be used as building blocks to form a deep neural network [36,45]. Once an autoencoder has been trained, one can stack another autoencoder on top of it, by training a second one which sees the latent representation of the first one as input. Once a stack autoencoder has been trained, their parameters describe multiple levels of representation for the input vector $\mathbf{x}^{(i)}$ and can be used to initialize a supervised deep neural network [32,46] or directly feed a classifier [7].

4.2 Learning language-independent semantic representation

To learn semantic representation of documents from the source language and the target language, we employ stacked autoencoders (SAEs) [36]. Documents from both the source language and the target language are binary bag-of-words representation. Assuming we have a large amount of parallel sentence pairs $\mathcal{X} = \{(\mathbf{x}_\mathrm{s}^{(1)}, \mathbf{x}_\mathrm{t}^{(1)}), \ldots, (\mathbf{x}_\mathrm{s}^{(n)}, \mathbf{x}_\mathrm{t}^{(n)})\}$, we first train SAEs with two hidden layers for each language separately. Then, we joint these two SAEs by feeding their respective second codings simultaneously to another autoencoder, whose hidden layer thus yields the fused semantic representation. Finally, we stack all layers and unfold them in order to fine-tune all SAEs. Figure 1 shows the model architecture.

Given a large-scale parallel sentence pairs $(\mathbf{x}_\mathrm{s}, \mathbf{x}_\mathrm{t})$, we would like to use it to learn representations in both languages that are aligned. For each sentence with binary bag-of-words representation $\mathbf{x}_\mathrm{s}$ in the source language and an associated binary bag-of-words representation $\mathbf{x}_\mathrm{t}$ in the target language for the same sentence in the target language, we use the hyperbolic tangent function as activation function for encoder f_θ and decoder $g_{\theta'}$. The weights of each autoencoder are tied, i.e. $\mathbf{W}'^{(1)} = \mathbf{W}^{(1)}$ in Fig. 1. We employ denoising autoencoders (DAEs) for pre-training the sentences in each language. For example in Fig. 1, let $\widetilde{\mathbf{x}}_\mathrm{s}$ and $\widetilde{\mathbf{y}}_\mathrm{s}$ denote the corrupted versions of the initial input vector $\mathbf{x}_\mathrm{s}$ and the first-layer output vector $\mathbf{y}_\mathrm{s}$, and we have the following high-level latent representations: $\mathbf{y}_\mathrm{s} = f_{\theta_\mathrm{s}}(\widetilde{\mathbf{x}}_\mathrm{s}) = s(\mathbf{W}^{(1)}\widetilde{\mathbf{x}}_\mathrm{s} + \mathbf{b}^{(1)})$, $\mathbf{y}'_\mathrm{s} = f_{\theta_\mathrm{s}}(\widetilde{\mathbf{y}}_\mathrm{s}) = s(\mathbf{W}^{(3)}\widetilde{\mathbf{y}}_\mathrm{s} + \mathbf{b}^{(3)})$. Essentially, the same steps repeat for the input vector $\mathbf{x}_\mathrm{t}$.

The bilingual autoencoder is fed with the concatenated final hidden codings of the source language and the target language as input and maps these inputs to a language-independent joint hidden layer $\hat{\mathbf{y}}$ with B units. We normalize both the source language and the target language input codings to unit length, i.e. $\mathbf{y}'_\mathrm{s} = \frac{\mathbf{y}'_\mathrm{s}}{\|\mathbf{y}'_\mathrm{s}\|}$. Again, we use tied weights for the bilingual autoencoder. Given the input codings $\mathbf{y}'_\mathrm{s}$ and $\mathbf{y}'_\mathrm{t}$, we force the autoencoder to detect dependencies between them while learning the mapping to the bilingual hidden layer. Therefore, we corrupt one input coding (e.g. $\mathbf{y}'_\mathrm{s}$) with a desired proportion ν, so that the corrupted representation from one language has to rely on the representation from the other language in order to reconstruct its missing input features. Let $\widetilde{\mathbf{y}}'_\mathrm{s}$ and $\widetilde{\mathbf{y}}'_\mathrm{t}$ denote the corrupted versions of $\mathbf{y}'_\mathrm{s}$ and $\mathbf{y}'_\mathrm{t}$, respectively, and we map the language-specific input codings to a language-independent joint hidden layer $\hat{\mathbf{y}}$:

$$\hat{\mathbf{y}} = s\left(\mathbf{W}^{(5)}\left[\widetilde{\mathbf{y}}'_\mathrm{s}, \widetilde{\mathbf{y}}'_\mathrm{t}\right] + \mathbf{b}^{(5)}\right) \tag{3}$$

where $[\widetilde{\mathbf{y}}'_\mathrm{s}, \widetilde{\mathbf{y}}'_\mathrm{t}]$ refers to the concatenation of two vectors $\widetilde{\mathbf{y}}'_\mathrm{s}$ and $\widetilde{\mathbf{y}}'_\mathrm{t}$ and then multiplied with $\mathbf{W}^{(5)}$.

During the decoding phase, we want to be able to perform a reconstruction of the original sentence in any of the languages. In particular, given a representation in any language, we would like a decoder $g_{\theta'_\mathrm{s}}$ that can perform a reconstruction in the source language and another decoder $g_{\theta'_{\mathrm{t}}}$ that can perform a reconstruction in the target language. Given the reconstruction layers, we have $\hat{\mathbf{y}}_{\mathrm{s}}{}' = g_{\theta'_{\mathrm{s}}} = s(\mathbf{W}'^{(5)}\hat{\mathbf{y}} + \mathbf{b}'^{(5)})$, $\hat{\mathbf{y}}_{\mathrm{s}} = g_{\theta'_{\mathrm{s}}} = s(\mathbf{W}'^{(3)}\hat{\mathbf{y}}' + \mathbf{b}'^{(3)})$, and $\hat{\mathbf{x}}_{\mathrm{s}} = g'_{\theta_{\mathrm{s}}} = s(\mathbf{W}'\hat{\mathbf{y}}_{\mathrm{s}} + \mathbf{b}'^{(1)})$. Essentially, the same steps repeat for the reconstruction process of $\hat{\mathbf{x}}_{\mathrm{t}}$.

The encoder/decoder decomposition allows us to learn a mapping within each language and across the languages. Specially, for a given parallel sentence pair $(\mathbf{x}_{\mathrm{s}}, \mathbf{x}_{\mathrm{t}})$, we can train the model to (1) reconstruct $\mathbf{x}_{\mathrm{s}}$ from itself [loss $L(\mathbf{x}_{\mathrm{s}}, \hat{\mathbf{x}}_{\mathrm{s}})$]; (2) reconstruct $\mathbf{x}_{\mathrm{t}}$ from itself [loss $L(\mathbf{x}_{\mathrm{t}}, \hat{\mathbf{x}}_{\mathrm{t}})$]; (3) construct $\hat{\mathbf{x}}_{\mathrm{s}}$ from $\mathbf{x}_{\mathrm{t}}$ [loss $L(\mathbf{x}_{\mathrm{t}}, \hat{\mathbf{x}}_{\mathrm{s}})$]; and (4) construct $\hat{\mathbf{x}}_{\mathrm{t}}$ from $\mathbf{x}_{\mathrm{s}}$ [loss $L(\mathbf{x}_{\mathrm{s}}, \hat{\mathbf{x}}_{\mathrm{t}})$]. We minimize the following objective function on a set of binary bag-of-words input vectors $\{(\mathbf{x}_{\mathrm{s}}^{(1)}, \mathbf{x}_{\mathrm{t}}^{(1)}), \ldots, (\mathbf{x}_{\mathrm{s}}^{(n)}, \mathbf{x}_{\mathrm{t}}^{(n)})\}$:

$$\begin{aligned} J = \sum_{i=1}^{n} \Big\{ & L\left(\mathbf{x}_{\mathrm{s}}^{(i)}, \hat{\mathbf{x}}_{\mathrm{s}}^{(i)}\right) + L\left(\mathbf{x}_{\mathrm{t}}^{(i)}, \hat{\mathbf{x}}_{\mathrm{t}}^{(i)}\right) + L\left(\mathbf{x}_{\mathrm{t}}^{(i)}, \hat{\mathbf{x}}_{\mathrm{s}}^{(i)}\right) \\ & + L\left(\mathbf{x}_{\mathrm{s}}^{(i)}, \hat{\mathbf{x}}_{\mathrm{t}}^{(i)}\right) \Big\} + \frac{\lambda}{2}\left(\|\theta\|_2 + \|\theta'\|_2\right) \end{aligned} \tag{4}$$

where J is the overall objective function, L is a loss function (e.g. cross-entropy), and λ is the weighing parameter. $\theta = \{\theta_{\mathrm{s}}, \theta_{\mathrm{t}}\} = \{\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{W}^{(3)}, \mathbf{W}^{(4)}, \mathbf{W}^{(5)}, \mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \mathbf{b}^{(3)}, \mathbf{b}^{(4)}, \mathbf{b}^{(5)}\}$ and $\theta' = \{\theta'_{\mathrm{s}}, \theta'_{\mathrm{t}}\} = \{\mathbf{W}'^{(1)}, \mathbf{W}'^{(2)}, \mathbf{W}'^{(3)}, \mathbf{W}'^{(4)}, \mathbf{W}'^{(5)}, \mathbf{b}'^{(1)}, \mathbf{b}'^{(2)}, \mathbf{b}'^{(3)}, \mathbf{b}'^{(4)}, \mathbf{b}'^{(5)}\}$ can be optimized by gradient descent methods. Note that we use tied weights for the stacked autoencoder, i.e. $\mathbf{W}^{(1)} = \mathbf{W}'^{(1)}$. In our experiments, we also add the constraints $\mathbf{b}^{(1)} = \mathbf{b}^{(2)}$, $\mathbf{b}^{(3)} = \mathbf{b}^{(4)}$, $\mathbf{b}'^{(1)} = \mathbf{b}'^{(2)}$ and $\mathbf{b}'^{(3)} = \mathbf{b}'^{(4)}$ before the nonlinearity across encoders, to encourage the encoders in both languages to produce representations on the same scale.

4.3 Document representation for cross-lingual sentiment classification

Once we have learned the parameters θ and θ', we can use them to transform the binary bag-of-words representation of the original documents into the high-level latent representation. Now, given the encoding parameters θ and a document $\mathbf{d}$ from the labelled training data $\mathcal{X}_{\mathrm{s}} = \{\mathbf{x}_{\mathrm{s}}^{(1)}, \ldots, \mathbf{x}_{\mathrm{s}}^{(n_{\mathrm{s}})}\}$ written in source language or the sentiment data $\mathcal{X}_{\mathrm{t}} = \{\mathbf{x}_{\mathrm{t}}^{(1)}, \ldots, \mathbf{x}_{\mathrm{t}}^{(n_{\mathrm{t}})}\}$ written in target language, we represent it as the language-independent high-level bilingual encoding representation $\hat{\mathbf{y}}_d$ by repeating the same encoding steps as in the training steps. Then, we train a simple sentiment classification model using SVM [8] on the labelled bilingual encoding data from the source language and predict the sentiment labels on the unlabelled bilingual encoding data from the target language.

5 Experiments

5.1 Experimental set-up

In this section, we conduct experiments for cross-lingual sentiment classification. We focus on the two common cross-lingual sentiment classification settings. In the first setting, no labelled data in the target language are available. This task has realistic significance, since in some situations, we need to quickly develop a sentiment classifier for languages that we do not have labelled data in hand. In this case, we classify documents in the target language using only labelled data in the source language. In the second setting, we have some labelled data in the target language. In this case, a more reasonable method is to make full use the both labelled data in the source language and the target language to build the sentiment classification model for the target language. In our experiments, for each setting, we consider two cases, one is English as the source language and Chinese as the target language, and another is Chinese as the source language and English as the target language.

5.2 Data set

For cross-lingual sentiment classification, we use the benchmark data set described in Lu et al. [14] and Meng et al. [17]. The labelled data sets consist of two English data sets and one Chinese data set.

MPQA-EN (*Labelled English Data*) The multi-perspective question answering (MPQA-EN) corpus [41] consists of newswire documents manually labelled with subjectivity information. We extract all sentences containing strong (e.g. intensity is medium or higher), sentiment-bearing (e.g. polarity is positive or negative) expressions following Choi and Cardie [4]. Sentences with both positive and negative strong expressions are then discarded, and the polarity of each remaining sentence is set to that of its sentiment-bearing expressions.

NTCIR-EN (*Labelled English Data*) *and NTCIR-CH (Labelled Chinese Data)* The NTCIR opinion analysis task [29,30] provides sentiment labelled news data in Chinese and English. The sentences with a sentiment polarity agreed to by at least two annotators are extracted. In this article, we use the Chinese data from NTCIR-6 as our Chinese labelled data. Since far fewer sentences in the English data pass the annotator agreement filter, we combine the English data from NTCIR-6 and NTCIR-7. The Chinese sentences are segmented using the Stanford Chinese word segmenter [34].

The statistics of the data sets are shown in Table 1. In our experiments, we evaluate four settings of the data: (1) MPQA-EN → NTCIR-CH; (2) NTCIR-EN → NTCIR-CH; (3) NTCIR-CH → MPQA-EN; and (4) NTCIR-CH → NTCIR-EN, where the word before an arrow corresponds with the source language and the word after an arrow corresponds with the target language.

To learn the parameters θ and θ', we use the Chinese–English parallel corpus [18], which contains about 10.5M sentence pairs. As mentioned earlier, unlike the previous work [10, 14,17], we do not use any word alignment between these parallel corpora. Specifically, we segment the Chinese sentences using the Stanford Chinese word segmenter [34] and remove all punctuations from the parallel corpus.

5.3 Model architecture

Our model has many hyper-parameters; we set these parameters empirically as follows: the source language autoencoder (see Fig. 1, left side) and the target language autoencoder (see Fig. 1, right side) consist of 1000 hidden units which are then mapped to the second hidden layer with 500 units (the corruption parameter is set to $\nu = 0.5$). The 500 source language and the 500 target language hidden units are fed to a bilingual autoencoder containing 500 latent units (the corruption parameter is set to $\nu = 0.5$). We use the model described above

Table 1 Statistics of data sets used in this article

	MPQA	NTCIR-EN	NTCIR-CH
Positive	1471 (30 %)	528 (30 %)	2378 (55 %)
Negative	3487 (70 %)	1209 (70 %)	1916 (44 %)
Total	4958	1737	4294

and the language-independent representations obtained from the output of the bilingual latent layer for the cross-lingual task. Note that some performance gains could be expected if these parameters are optimized on the development set.

5.4 Baseline methods

In our experiments, we compare our proposed SAEs with the following baseline methods:

SVM This method learns a SVM classifier for each language given the monolingual labelled data. In this article, SVM light [8] is used for all the SVM-related experiments.

MT-SVM This method employs Google Translate (http://translate.google.com) to translate the labelled data from the source language (e.g. English) to the target language (e.g. Chinese) and uses the translated results to train a SVM classifier for the target language.

MT-Cotrain This method is based on a co-training framework described in Wan [39]. For easy description, we assume that the source language is English while the target language is Chinese. First, two monolingual SVM classifiers are trained on English labelled data and Chinese data translated from English labelled data. Second, the two classifiers make prediction on Chinese unlabelled data and their English translation, respectively. Third, the most confidently predicted English and Chinese documents are added to the training set, and the two monolingual SVM classifier are re-trained on the expanded training set. Following the literature [17], we repeat the second and third steps 100 times to obtain the final classifiers.

Joint-Train This method uses English labelled data and Chinese labelled data to obtain initial parameters for two maximum entropy classifiers and then conducts EM-iterations to update the parameters to gradually improve the agreement of the two monolingual classifiers on the unlabelled parallel data [14].

CLMM This is the state-of-the-art method for cross-lingual sentiment classification described in Meng et al. [17]. This method proposes a generative cross-lingual mixture model (CLMM) and learns previously unseen sentiment words from the large-scale bilingual parallel data to improve the vocabulary coverage.

5.5 Cross-lingual sentiment classification only using source labelled data

In this section, we investigate cross-lingual sentiment classification towards the case that we have only labelled data from the source language. The first set of experiments are conducted on using only English labelled data to build sentiment classifier for Chinese sentiment classification. This is a challenging task since we do not have any Chinese labelled data in hand.

Table 2 shows the accuracy of the baseline systems as well as the proposed model (SAEs). As seen from the table, our proposed approach SAEs outperforms all baseline methods for Chinese sentiment classification only using the labelled English data. Specifically, our proposed approach improves the accuracy, compared to MT-SVM, by 18.58 and 10.01 % (row 2 vs. row 6) on Chinese in the first setting and in the second setting, respectively. Meanwhile, the accuracy of MT-SVM on NTCIR-EN → NTCIR-CH is much better than that on MPQA-EN → NTCIR-CH. The reason may be that NTCIR-EN and NTCIR-CH cover similar topics. Besides, we also observe that using a parallel corpus instead of machine translations can improve the classification accuracy (row 2 and row 3 vs. row 5 and row 6). Moreover, our proposed SAEs outperforms CLMM (row 5 vs. row 6). The reason may be that our method can learn language-independent high-level features without using the off-the-shelf word alignment tool (e.g. GIZA++).

Table 2 Sentiment classification accuracy for Chinese only using English labelled data

	Method	MPQA-EN → NTCIR-CH	NTCIR-EN → NTCIR-CH
1	SVM	N/A	N/A
2	MT-SVM	54.33	62.34
3	MT-Cotrain	59.11 (+4.78)	65.13 (+2.79)
4	Joint-Train	N/A	N/A
5	CLMM	71.52 (+17.19)	70.96 (+8.62)
6	SAEs	**72.91** (+**18.58**)	**72.35** (+**10.01**)

Improvements in different methods over baseline MT-SVM are shown in parentheses
Bold values indicate that our proposed approach is statistically significant with the baseline methods by using t-test (p value < 0.5)

Table 3 Sentiment classification accuracy for English only using Chinese labelled data

	Method	NTCIR-CH → MPQA-EN	NTCIR-CH → NTCIR-EN
1	SVM	N/A	N/A
2	MT-SVM	52.47	58.51
3	MT-Cotrain	58.63 (+6.16)	63.72 (+5.21)
4	Joint-Train	N/A	N/A
5	CLMM	68.29 (+15.82)	69.15 (+10.64)
6	SAEs	**71.55** (+**19.08**)	**73.57** (+**15.06**)

Improvements in different methods over baseline MT-SVM are shown in parentheses
Bold values indicate that our proposed approach is statistically significant with the baseline methods by using t-test (p value < 0.5)

The second set of experiments are conducted on using only Chinese labelled data to build sentiment classifier for English sentiment classification. Table 3 shows the sentiment classification accuracy for English using only Chinese labelled data. From this table, we have the similar observations as in Table 2.

5.6 Cross-lingual sentiment classification using source language and target language labelled data

The third set of experiments are conducted on using both English labelled data and Chinese labelled data to build the Chinese sentiment classifier. We conduct fivefold cross-validation on Chinese labelled data and use the similar settings with Meng et al. [17].

Table 4 shows the average accuracy of baseline systems as well as our proposed SAEs. From this table, we can see that SVM performs significantly better than MT-SVM. The reason may be that we use the original Chinese labelled data instead of translated Chinese labelled data. We also find that all four methods which employ the unlabelled parallel corpus, namely MT-Cotrain, Joint-Train, CLMM and SAEs, still show improvements over the baseline SVM. Moreover, our proposed SAEs obtain the state-of-the-art accuracy of 83.87 on MPQA-EN → NTCIR-CH and 83.68 on NTCIR-EN → NTCIR-CH. This again validates that learning language-independent high-level latent representation is better than using word alignment tool for cross-lingual sentiment classification.

Table 4 Sentiment classification accuracy for Chinese using English and Chinese labelled data

	Method	MPQA-EN → NTCIR-CH	NTCIR-EN → NTCIR-CH
1	SVM	80.58	80.58
2	MT-SVM	54.33 (−26.25)	62.34 (−18.24)
3	MT-Cotrain	80.93 (+0.35)	82.28 (+2.79)
4	Joint-Train	83.42 (+2.84)	83.11 (+2.53)
5	CLMM	83.02 (+2.44)	82.73 (+2.15)
6	SAEs	**83.87** (+**3.29**)	**83.68** (+**3.10**)

Improvements in different methods over baseline SVM are shown in parentheses
Bold values indicate that our proposed approach is statistically significant with the baseline methods by using t-test (p value < 0.5)

Table 5 Sentiment classification accuracy for English using Chinese and English labelled data

	Method	NTCIR-CH → MPQA-EN	NTCIR-CH → NTCIR-EN
1	SVM	75.06	74.87
2	MT-SVM	52.47 (−22.59)	58.51 (−16.36)
3	MT-Cotrain	77.52 (+2.46)	80.34 (+5.47)
4	Joint-Train	78.26 (+3.20)	82.75 (+7.88)
5	CLMM	78.11 (+3.05)	82.23 (+7.36)
6	SAEs	**79.32** (+**4.26**)	**84.01** (+9.14)

Improvements in different methods over baseline SVM are shown in parentheses
Bold values indicate that our proposed approach is statistically significant with the baseline methods by using t-test (p value < 0.5)

The fourth set of experiments are conducted on using both English labelled data and Chinese labelled data to build the English sentiment classifier. We conduct fivefold cross-validation on English labelled data and report the average accuracy of sentiment classification. Table 5 shows the average accuracy of baseline systems as well as our proposed SAEs. As is seen, our proposed SAEs can significantly outperform other baseline systems and gains the state-of-the-art performance for English sentiment classification by using Chinese and English labelled data.

5.7 The influence of unlabelled data

We investigate how the size of the unlabelled parallel data affects the sentiment classification in this subsection. We vary the number of documents in the unlabelled corpus from 5000 to 50,000. We use only Chinese labelled data or only English labelled data in these experiments, since this more directly reflects the effectiveness of the proposed approach in utilizing unlabelled parallel data. From Figs. 2 and 3, we can see that when more unlabelled parallel data are used to learn the parameters, the accuracy of SAEs consistently improves. The performance of SAEs is remarkably superior than MT-Cotrain and CLMM.

6 Conclusion and future work

In this article, we present a model that uses stacked autoencoders to learn language-independent semantic representations for cross-lingual sentiment classification. Specially,

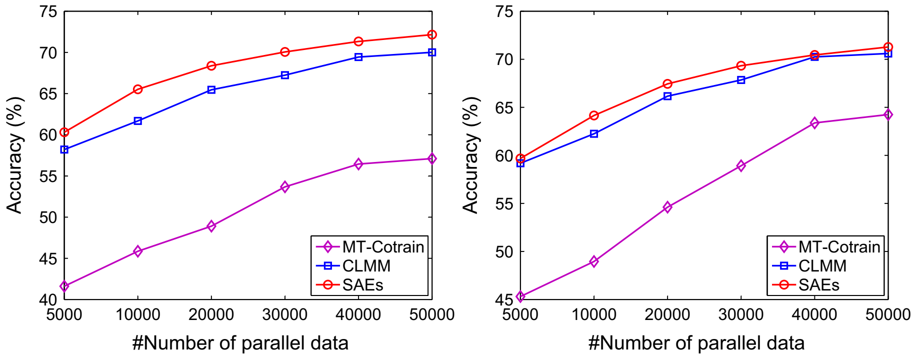

Fig. 2 Sentiment classification accuracy with different number of unlabelled parallel data for Chinese only using English labelled data. The *left figure* shows the performance on MPQA-EN → NTCIR-CH, while the *right figure* shows the performance on NTCIR-EN → NTCIR-CH

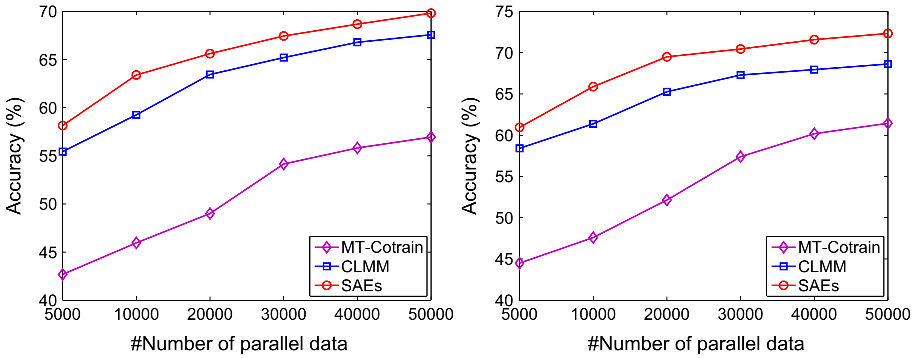

Fig. 3 Sentiment classification accuracy with different number of unlabelled parallel data for English only using Chinese labelled data. The *left figure* shows the performance on MPQA-EN → NTCIR-CH, while the *right figure* shows the performance on NTCIR-EN → NTCIR-CH

our model is firstly trained on a large-scale bilingual parallel data and then projects the source language and the target language into a common space that fuses the two types of information together. The goal of our model is to learn multiple levels of representations through a hierarchy of network architectures, where higher-level representations are expectes to help define higher-level concepts. The learned higher-level concepts can be used to bridge the gap between the source language and the target language. To the best of our knowledge, our model is novel in its use of bag-of-words vector input in a deep neural network. To evaluate the effectiveness of the proposed approach, we conduct experiments on the task of English–Chinese cross-lingual sentiment classification. The empirical results show that the proposed approach is very effective for cross-lingual sentiment classification and outperforms other comparison methods.

For future work, we would like to investigate extensions of our bag-of-words bilingual autoencoder to bag-of-n-grams, where the model would also have to learn representations for phrases. Such a model should be particularly useful in the context of a cross-lingual information retrieval. Besides, we would also like to explore the possibility of converting our bilingual model to a multilingual model which can learn common representations for multiple languages given different amount of parallel data between these languages.

Acknowledgments This work was supported by the National Natural Science Foundation of China (Nos. 61303180, 61272332, 61402191), the Beijing Natural Science Foundation (No. 4144087), the Major Project of National Social Science Found (No. 12&2D223), the Fundamental Research Funds for the Central Universities (No. CCNU15ZD003) and also sponsored by CCF-Tencent Open Research Fund. We thank the anonymous reviewers for their insightful comments.

References

1. Banea C, Mihalcea R, Wiebe J, Hassan S (2008) Multilingual subjectivity analysis using machine translation. In: Proceedings of the conference on empirical methods in natural language processing, Honolulu, Hawaii, pp 127–135
2. Bespalov D, Bai B, Qi Y, Shokoufandeh A (2011) Sentiment classification based on supervised latent N-gram analysis. In: Proceedings of the 20th ACM international conference on information and knowledge management, Glasgow, Scotland, UK, pp 375–382
3. Baccianella S, Esuli A, Sebastiani F (1996) SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: Language resources and evaluation
4. Choi Y, Cardie C (2008) Learning with compositional semantics as structural inference for subsentential sentiment analysis. In: Proceedings of the conference on empirical methods in natural language processing, Honolulu, Hawaii, pp 793–801
5. Duh K, Fujino A, Nagata M (2011) Is machine translation ripe for cross-lingual sentiment classification? In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, Portland, OR, pp 429–433
6. Goldberg B, Zhu X (2006) Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization? In: Proceedings of the first workshop on graph based methods for natural language processing, Stroudsburg, PA, USA, pp 45–52
7. Glorot X, Bordes A, Bengio Y (2011) Domain adaptation for large-scale sentiment classification: a deep learning approach. In: Proceedings of the twenty-eight international conference on machine learning
8. Joachims T (1999) Making large-scale support vector machine learning practical. In: Advances in kernel methods, Cambridge, MA, pp 169–184
9. Kim J, Li J, Lee J (2009) Discovering the discriminative views: measuring term weights for sentiment analysis. 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, Suntec, Singapore, pp 253–261
10. Klementiev A, Titov I, Bhattarai B (2012) Inducing crosslingual distributed representations of words. In: Proceedings of the international conference on computational linguistics, Bombay, India
11. Lafferty J (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: ICML, Morgan Kaufmann, pp 282–289
12. Li S, Wang Z, Zhou G, Lee S (2011) Semi-supervised learning for imbalanced sentiment classification. In: Proceedings of the twenty-second international joint conference on artificial intelligence. Catalonia, Spain, Barcelona, pp 1826–1831
13. Liu B (2012) Sentiment analysis and opinion mining. In: Synthesis lectures on human language technologies
14. Lu B, Tan C, Cardie C, Tsou K (2011) Joint bilingual sentiment classification with unlabeled parallel corpora. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, Portland, OR, pp 320–330
15. Maas L, Daly E, Pham T, Huang D, Ng Y, Potts C (2011) Learning word vectors for sentiment analysis. In: Proceedings of the 49th annual meeting of the association for computational linguistics, Portland, OR, pp 142–150
16. Mejova Y, Padmini S (2011) Exploring feature definition and selection for sentiment classifiers. In: Proceedings of the fifth international AAAI conference on webblogs and social media
17. Meng X, Wei F, Liu X, Zhou M, Xu G, Wang H (2012) Cross-lingual mixture model for sentiment classification. In: Proceedings of the 50th annual meeting of the association for computational linguistics, Jeju Island, Korea, pp 572–581
18. Munteanu S, Marcu D (2005) Improving machine translation performance by exploiting non-parallel corpora. Comput Linguist 31(4):477–504
19. Nakagawa T, Inui K, Kurohashi S (2010) Dependency tree-based sentiment classification using CRFs with hidden variables. In: The 2010 annual conference of the North American chapter of the association for computational linguistics, Los Angeles, CA, pp 786–794

20. Ng V, Dasgupta S, Arifin S (2006) Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In: Proceedings of the COLING/ACL on main conference poster sessions, Sydney, Australia, pp 611–618
21. Och F, Ney H (2000) Improved statistical alignment models. In: Proceedings of the 38th annual meeting on association for computational linguistics, Hong Kong, pp 440–447
22. Pan S, Ni X, Sun J, Yang Q, Chen Z (2010) Cross-domain sentiment classification via spectral feature alignment. In: Proceedings of the 19th international conference on World Wide Web, Raleigh, NC, USA, pp 751–760
23. Pan J, Xue G, Yu Y, Wang Y (2011) Cross-lingual sentiment classification via Bi-view non-negative matrix tri-factorization. In: Proceedings of the 15th Pacific-Asia conference on advances in knowledge discovery and data mining, Shenzhen, China, pp 289–300
24. Pang B, Lee L (2008) Opinion mining and sentiment analysis. Found Trends Inf Retr 2(1):1–135
25. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on empirical methods in natural language processing, Stroudsburg, PA, USA, pp 79–86
26. Pang B, Lee L (2004) A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on association for computational linguistics, Barcelona, Spain
27. Peng W, Park D (2011) Generate adjective sentiment dictionary for social media sentiment analysis using constrained nonnegative matrix factorization. In: The international conference on weblogs and social media, Barcelona, Spain. The AAAI Press
28. Prettenhofer P, Stein B (2010) Cross-language text classification using structural correspondence learning. In: Proceedings of the 48th annual meeting of the association for computational linguistics, Uppsala, Sweden, pp 1118–1127
29. Seki Y, Evans D, Ku L, Chen H, Kando N, Lin C (2007) Overview of opinion analysis pilot task at NTCIR-6. In: Proceedings of the workshop meeting of the national institute of informatics test collection for information retrieval systems (NTCIR)
30. Seki Y, Evans D, Ku L, Chen H, Kando N, Lin C (2007) Overview of multilingual opinion analysis task at NTCIR-7. In: Proceedings of NTCIR-7
31. Seki Y, Evans D, Ku L, Chen H, Kando N, Lin C (2004) Mining multilingual opinions through classification and translation. In: AAAI Spring symposium on exploring attitude and affect in text
32. Silberer C, Lapata M (2014) Learning grounded meaning representations with autoencoders. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, Baltimore, MD, pp 721–732
33. Taboada M, Brooke J, Tofiloski M, Voll K, Stede M (2011) Lexicon-based methods for sentiment analysis. Comput Linguist 37(2):267–307
34. Tseng H (2005) A conditional random field word segmenter. In: Fourth SIGHAN workshop on Chinese language processing
35. Turney D (2002) Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th annual meeting on association for computational linguistics, Philadelphia, PA, pp 417–424
36. Vincent P, Larochelle H, Bengio Y, Manzagol P (2008) Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on machine learning, Helsinki, Finland, pp 1096–1103
37. Vikas S, Prem M (2008) Document-word co-regularization for semi-supervised sentiment analysis. In: Proceedings of the international conference on data mining
38. Wan X (2008) Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis. In: Proceedings of the conference on empirical methods in natural language processing, Honolulu, Hawaii, pp 553–561
39. Wan X (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, Suntec, Singapore, pp 235–243
40. Wan X (2011) Bilingual co-training for sentiment classification of Chinese product reviews. Comput Linguist 37(3):587–616
41. Wiebe J, Cardie C (2005) Annotating expressions of opinions and emotions in language. In: Language resources and evaluation, language resources and evaluation (formerly computers and the humanities)
42. Wu K, Wang X, Lu B (2008) Cross language text categorization using a bilingual lexicon. In: Proceedings of the third international joint conference on natural language processing

43. Xia R, Zong C (2010) Exploring the use of word relation features for sentiment classification. In: Proceedings of the 23rd international conference on computational linguistics: posters, Beijing, China, pp 1336–1344
44. Xiao M, Guo Y (2013) Semi-supervised representation learning for cross-lingual text classification. In: Proceedings of the conference on empirical methods on natural language processing, Seattle, USA, pp 1465–1475
45. Yoshua B, Pascal L, Dan P, Hugo L (2011) Greedy layer-wise training of deep networks. In: Proceedings of the NIPS
46. Yoshua B (2011) Learning deep architectures for AI. In: Foundations and trends in machine learning, Hanover, MA, USA, pp 1–127
47. Zhou G, He T, Zhao J (2014) Bridge the language gap: learning distributed semantics for cross-lingual sentiment classification. In: Proceedings of the 3rd international conference on natural language processing and Chinese computing, Shenzhen, China, pp 138–149
48. Zhou G, Zhao J, Zeng D (2014) Sentiment classification with graph co-regularization. In: Proceedings of COLING 2014, the 25th international conference on computational linguistics: technical papers, Dublin, Ireland, pp 1331–1340
49. Zhou G, He T, Zhao J, Wu W (2015) A subspace learning framework for cross-lingual sentiment classification with partial parallel data. In: Proceedings of the international joint conference on artificial intelligence, Buenos Aires
50. Zou Y, Socher R, Cer M, Manning D (2013) Bilingual word embeddings for phrase-based machine translation. In: Proceedings of the conference on empirical methods on natural language processing, Seattle, USA, pp 1393–1398

Guangyou Zhou received his Ph.D. degree from National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (IACAS) in 2013. Before that, he received his M.S. degree from Northeast Normal University in 2008. Currently, he worked as an Associate Professor at the School of Computer, Central China Normal University. His research interests include natural language processing and information retrieval. Now he has served several programme committees of the major international conferences in the field of natural language processing and knowledge engineering and also served as reviewers for several journals. He has won the best paper award in COLING 2014. In the past 5 years, he has published more than 20 papers in the leading journals and top conferences, such as ACM TWEB, ACM TIST, IEEE TKDE, ACL, IJCAI, CIKM, COLING.

Zhiyuan Zhu is a research staff in Chinese Institute of Electronics. He received his Ph.D. degree from Institute of Automation Chinese Academy of Sciences in 2013. Before that, he received his B.S. degree from Hunan Normal University in 2008. His research interests include network communication, cloud computing and big data application. He has published several peer-reviewed research papers in various journals and conferences.

Tingting He is a full professor and the founding director of the natural language processing laboratory at the School of Computer, Central China Normal University. She received her Ph.D. degree from Central China Normal University in 2003. Before that, she received her M.S. degree and B.S. degree from Wuhan University in 1985 and 1988, respectively. Her research interests include natural language processing, information retrieval and database application. She has published more than 100 peer-reviewed research papers in various journals, conferences and books. Her research projects are funded by the National Science Foundation (NSF).

Xiaohua Tony Hu is a full professor and the founding director of the data mining and bioinformatics laboratory at the College of Computing and Informatics (the former College of Information Science and Technology, one of the best information science schools in the USA, ranked as #1 in 1999 and #6 in 2010 in information systems by U.S. News & World Report). He is also serving as the founding Co-Director of the NSF Center (I/U CRC) on Visual and Decision Informatics (NSF CVDI), IEEE Computer Society Bioinformatics and Biomedicine Steering Committee Chair, and IEEE Computer Society Big Data Steering Committee Chair. Tony is a scientist, teacher and entrepreneur. He joined Drexel University in 2002. He founded the International Journal of Data Mining and Bioinformatics (SCI indexed) in 2006, International Journal of Granular Computing, Rough Sets and Intelligent Systems in 2008. Earlier, he worked as a research scientist in the world-leading R&D centres such as Nortel Research Center, and Verizon Lab (the former GTE Labs). In 2001, he founded the DMW Software in Silicon Valley, California. He has a lot of experience and expertise to convert original ideas into research prototypes and eventually into commercial products, many of his research ideas have been integrated into commercial products and applications in data mining fraud detection, database marketing.