

Article

# Cross-Domain Text Sentiment Analysis Based on CNN\_FT Method

Jiana Meng, Yingchun Long, Yuhai Yu \* , Dandan Zhao and Shuang Liu

School of Computer Science & Engineering, Dalian Minzu University, Dalian 116600, China; mengjn@dlnu.edu.cn (J.M.); yingchunlong31@gmail.com (Y.L.); zhaodd@dlnu.edu.cn (D.Z.); liushuang@dlnu.edu.cn (S.L.)

\* Correspondence: yuyh@dlnu.edu.cn; Tel.: +86-411-8765-6256

Received: 25 February 2019; Accepted: 22 April 2019; Published: 1 May 2019



**Abstract:** Transfer learning is one of the popular methods for solving the problem that the models built on the source domain cannot be directly applied to the target domain in the cross-domain sentiment classification. This paper proposes a transfer learning method based on the multi-layer convolutional neural network (CNN). Interestingly, we construct a convolutional neural network model to extract features from the source domain and share the weights in the convolutional layer and the pooling layer between the source and target domain samples. Next, we fine-tune the weights in the last layer, named the fully connected layer, and transfer the models from the source domain to the target domain. Comparing with the classical transfer learning methods, the method proposed in this paper does not need to retrain the network for the target domain. The experimental evaluation of the cross-domain data set shows that the proposed method achieves a relatively good performance.

**Keywords:** cross-domain; sentiment classification; transfer learning; convolutional neural network; word2vec

## 1. Introduction

As is well known, the performance of training and update on machine learning models depends on the labeled data, although one can get huge amounts of data, but can rarely correct the manually labeled data. However, the labeling work may be time-consuming and expensive, which presents a great challenge to machine learning models.

To solve this problem, Yang et al. propose a transfer learning method [1], the core idea of which is to find the similarity between the source and target domains and to transfer the model or the labeled data used in the source domain to those in the target domain in the view of similarity. Thereafter, one can perform the new training based on the existing similarity. In fact, the ability to transfer knowledge is inherent, for example, if you can play tennis, you can learn to play badminton easily. Because these activities often have a very high degree of similarity, one can solve new problems based on the existing learning methods and modify them gradually.

Transfer learning has many applications in the field of text sentiment analysis [2,3]. Blitzer et al. propose the SCL method [4] to find the common characteristics between the source and target domains. Pan et al. [5] propose the SFA algorithm to align the domain-specific words from different domains into the unified clusters with the help of domain independent words as a bridge. At present, with the rapid development of deep learning, some relevant methods are applied to transfer learning, and obtain many significant results. Ganin et al. [6] propose the DANN algorithm. The authors embed the domain adaptive learning into the feature representation process, interestingly, the resulting feedforward neural network can be directly applied to the target domain. The review article contributed by Zhang et al. [7] illustrates that deep learning can effectively solve the problem of sentiment analysis. Lai et al. [8]

propose the use of word vector and convolutional neural network for solving the Chinese text sentiment analysis problem. Li et al. [9] propose an end-to-end adversarial memory network (AMN) consisting of two parameter-shared memory networks for cross-domain sentiment classification. The two networks are jointly trained so that the selected features can minimize the sentiment classification error and can make the domain classifier indiscriminative between the representations from the source domain or the target domain simultaneously. In the view of methodological point, Fortuna et al. [10] present the paradigm of cellular neural networks and achieve a complete maturity.

In this paper we propose an inductive transfer learning method based on the word vector and convolutional neural network model for solving the cross-domain text sentiment analysis. Interestingly, this method employs a two-stage training procedure. First, we pre-train the embeddings that capture the additional context via another domain in the embedding level and share the weights in the convolutional layer and the pooling layer between the source and target domain texts. Subsequently, we fine-tune the weights in the last layer of the fully connected layer. Experiments on the Chinese and English corpora demonstrate that the proposed method can achieve good performance.

## 2. Related Works

### 2.1. Transfer Learning

Transfer learning mainly solves the distribution differences for cross-domain problems. There are four main strategies of implementation, namely sample-based, feature-based, model-based and relationship-based.

The sample-based transfer learning method transfers the samples of the source and target domains through weight reuse, that is to say, different weights are given directly to different samples. Dai et al. [11] propose the TrAdaboost method to improve the sample weights that are beneficial to the target classification task and to reduce the sample weights that are not conducive to the target classification task. By using the joint matrix decomposition and neural networks, Tan et al. propose two transfer learning methods, transitive transfer learning (TTL) [12] and distant domain transfer learning (DDTL) [13], to share multiple-knowledge between similar domains. The feature-based transfer learning method assumes that the features of the source and target domains are not in a common space, so one should transform them into a similar space. Blitzer et al. [14] propose a learning method based on the structural correspondence learning, in which the algorithm can transform some unique features in a space to the features in all other spaces through mapping. Particularly, some feature-based transfer learning methods are combined with neural networks [15–17]. The model-based transfer learning method transfers the shared parameters by building a model, which is often used in neural networks since the structure of neural networks can be transferred directly. Compared with the traditional non-deep transfer learning methods, deep transfer learning improves the learning effect on different tasks. Glorot et al. [18] use a deep learning model, i.e., SD Ae, to pre-train the unlabeled data in multiple domains, and combine the pre-training model with the labeled texts of source domains to train the sentiment classification model. The classification performance of this model is superior to that of the SCL model [4] and the SFA model [5] in 22 domains. Since the model-based transfer learning method reduces the dependence of the pivot vocabulary and the source domain labeled data, the model-based transfer combined with deep learning is superior to the traditional feature-based and sample-based transfer learning method in the overall efficiency of the cross-domain text sentiment analysis. The relationship-based transfer learning method mainly focuses on the relation between the source and target domain samples. Davis et al. [19] use the Markov logic network to explore the relation similarity between different domains.

### 2.2. Cross-Domain Text Sentiment Analysis

Word vector mentioned in this paper is a low-dimensional real vector and is another related research field of text, which refers to the distributed expression of words. The word vector is the point

in an N-dimensional real space and represents the relation between the points reflecting the potential semantic associated with words. The neural network's ability to perceive structural information brings about the development of word vectors [20–23] and applications, moreover, it also broadens the way for sentiment analysis [24].

In general, sentiment analysis can be divided into positive and negative, namely binary sentiment classification. Many detailed classification methods can also be formulated as a three-class classification problem (positive, negative and neutral) and a multiple sentiment classification problem (anger, sorrow, joy and other emotions). Text sentiment analysis methods are mainly divided into two categories, namely, the lexicon-based method and the machine learning-based method. The lexicon-based method mainly uses the emotional lexicon to extract the emotional expression keywords in the corpus. In a recent study, Mohammad et al. [25] use mass crowdsourcing to build emotional annotations and then utilize the features of the most advanced sentiment analysis systems to predict emotions in new tweets. In addition, Xing et al. [26] propose a new lexicons-based approach to simultaneously train an emotion classifier and adapt the word polarity to the target domain. The method tracks the incorrectly predicted sentences and uses them as supervision to mimic the lifelong cognitive process of dictionary learning. Machine learning-based method is a popular research direction in recent years. The test data is identified by training data, and then feature extraction is performed. The text sentiment analysis model is generated through model training, and then the text sentiment analysis is carried out. According to different sentiment classification algorithms, it can be divided into Naive Bayes (NB), Maximum Entropy (ME) and support vector machine (SVM). Pang et al. [27] use NB, ME and SVM to classify text emotions and examine the effectiveness of applying machine learning techniques to the sentiment classification problem. Desai et al. [28] discuss in detail various techniques for sentiment analysis of twitter data, including machine learning-based methods.

However, traditional machine learning method requires training and testing the data to follow the same distribution. Existing studies have shown that the supervised classification method is effective for sentiment analysis. For text sentiment analysis, it is regarded as a special classification, that is, the text is classified in the view of a certain subject in the text. According to different granularities of text, text sentiment analysis is usually divided into three classes, i.e., phrase level sentiment analysis, sentence level sentiment analysis and document level sentiment analysis. In most cases, the existing labeled data do not belong to the same domain as the data to be judged, so the performance of the supervised classification algorithm is obviously reduced, which leads to the cross-domain text sentiment analysis problem.

Cross-domain text sentiment analysis involves the knowledge transfer problem between different domains. Meng et al. [29] propose a cross-domain text sentiment analysis method using the machine learning from the perspectives of features and samples. Huang et al. [30] compare the applications of naive Bayes (NB), support vector classification (SVC) and expectation maximization (EM) in cross-domain text sentiment analysis. The experimental results show that the EM method is slightly better than the NB method and the SVC method. Xia et al. [31] use the ensemble features such as part-of-speech tagging and word relation to establish an ensemble model for the NB, SVC and EM methods, and the experimental results are better than the traditional single machine learning models. Deshmuke et al. [32] combine the improved maximum entropy model with the binary graph clustering model and achieve relatively high accuracy for the classification of affective words. Tang et al. [33] analyze the applications of deep learning methods in sentiment analysis earlier and find that it is superior to traditional methods in sentiment classification, viewpoint extraction and emotion dictionary construction. Yu et al. [34] use the deep learning method to model sentences and the experimental results show that the deep learning model is superior to the traditional structure corresponding to learning models. Because deep learning can better reveal and obtain the internal semantic representation of text information in different domains, it is generally superior to the traditional graph model method and the statistical learning method.

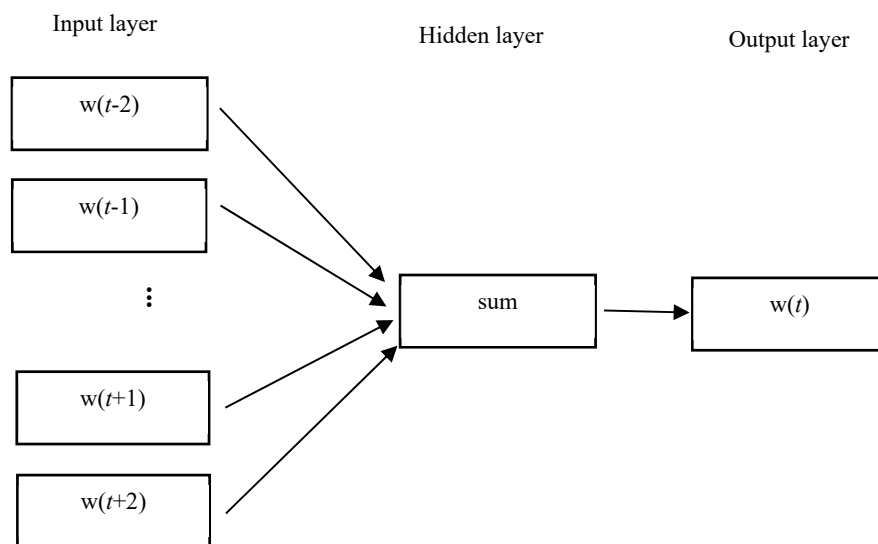
### 3. Transfer Learning Method Based on Deep Learning Model

The formulated problem in this paper mainly focuses on transfer learning, in which the major difficulty is that the source and target domain data are not likely to be drawn from the same distribution. This paper proposes a transfer learning method based on the multi-layer convolutional neural network model, called CNN\_FT, to solve the cross-domain text sentiment analysis problem. We train the neural network model through the source domain and share the weights of the convolution kernel in the model. We use the convolution kernel weights of the source domain to extract the corresponding features in the target domain. Finally, we retrain a small part of the target domain data so as to adjust the parameters of the fully connected layer weights of the model, where  $k$  is the number of target domain samples included in training, and  $k = 200, 500, 1000$  and  $2000$  out of  $4000$ , respectively.

#### 3.1. Feature Representation

The current text representation usually uses the bag-of-words model since the model is simple to construct and can reduce the complexity of vector calculation. However, this model has many shortcomings, for example, when the sample data is large and contains rich keywords, the feature dimension of the text will be very high and may lead to a dimensional explosion. The word vector matrix is particularly sparse, and it is likely to yield overfitting. In order to solve the problem, Mikolov et al. [22] introduce the word embedding model, an efficient method for learning high quality vector representations of words from large amounts of unstructured text data, to capture the syntactic and semantic information, which are very important to sentiment analysis.

Word2vec trains word vectors according to the relation between contexts. There are two training modes, i.e., Skip Gram and CBOW, in which Skip Gram predicts contexts based on words and CBOW predicts words according to contexts. This paper should train words based on the CBOW neural network model, as shown in Figure 1.



**Figure 1.** Training the word vectors using the CBOW model.

Let  $V$  be the vocabulary size,  $D$  be the dimension of word vectors, and  $x_1, x_2, \dots, x_m$  be the context word vectors. The average value of the context word vectors is given by

$$\hat{x} = \frac{1}{m} \sum_{i=1}^m x_i \quad (1)$$

The score vector  $z$  is then defined in terms of Equation (1), the formula is as follows,

$$z = U \hat{x} \quad (2)$$

where  $U$  is the weight matrix associated with the output words.

To obtain the maximum of the conditional probability of the target word vector  $w_t$ , i.e.,  $p(w_t | w_1, w_2, \dots, w_m)$ , we define the loss function according to the minimization of the negative log-likelihood function, namely,

$$L = -\log p(w_t | w_{t-m}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+m}) \quad (3)$$

Equation (3) can also be expressed as

$$L = \text{softmax}(z) \quad (4)$$

According to the gradient descent algorithm, the iterative formula is given by

$$w_t = w_t - \alpha(\hat{x} - x_i), i \in (1, m) \quad (5)$$

where  $\alpha$  is the step describing the learning speed.

### 3.2. Convolutional Neural Network Structure

The convolutional neural network (CNN) is a model based on the deep neural network [35], which is generally composed of three layers, i.e., a convolutional layer, a pooling layer and a fully connected layer. Meanwhile, each of the convolution kernels represents different features, and multiple convolution kernels extract the features and combine them.

In this paper, we propose a multi-convolutional neural network model to solve the cross-domain sentiment analysis problem. The overall flow chart is shown in Figure 2. The neural network model is firstly trained via using the source domain data.

The first layer is an input layer, denoted by  $xs \in R_s^{n \times 1}$ , in which  $xs$  representing the sentence in the source domain is stored in a matrix. The fixed length of the sentence is set to be  $n$  and to be 0 if the length is less than  $n$ . The embedded layer transforms the original input layer to  $xs \in R_s^{n \times k}$  by using Word2vec, where  $k$  is the dimension of a word vector. Let  $xs_i$  ( $xs_i \in R_s^k$ ) represent the  $i$ -th word in the sentence, the sentence is then expressed as

$$xs_{1:n} = xs_1 \oplus xs_2 \oplus \dots \oplus xs_n \quad (6)$$

where  $\oplus$  is a join operator.

The second layer is a convolutional layer, which is used to extract the features of sentences. To obtain these features, we should slide the  $h \times k$  convolution kernel  $w_s \in R_s^{h \times k}$  in the input layer from top to bottom to complete the convolution operation, and then obtain a feature map, where the feature map column is 1, the line is  $n - h + 1$ , i.e.,

$$c_s = (c_s^1, c_s^2, \dots, c_s^{n-h+1}) \quad (7)$$

where

$$c_s^i = f(w_s \cdot xs_{i:i+h-1} + b_s) \quad (8)$$

In Equation (8),  $f$  is a nonlinear activation function and  $b_s$  is a bias term.

In the training model, over-fitting may occur. In order to improve the generalization ability of the model, we use the Dropout method proposed by Hinton et al. [36] to improve the structural performance of the neural network. The main function is to randomly ignore the neurons in the convolutional layer and to reduce the interaction between hidden layer neurons.

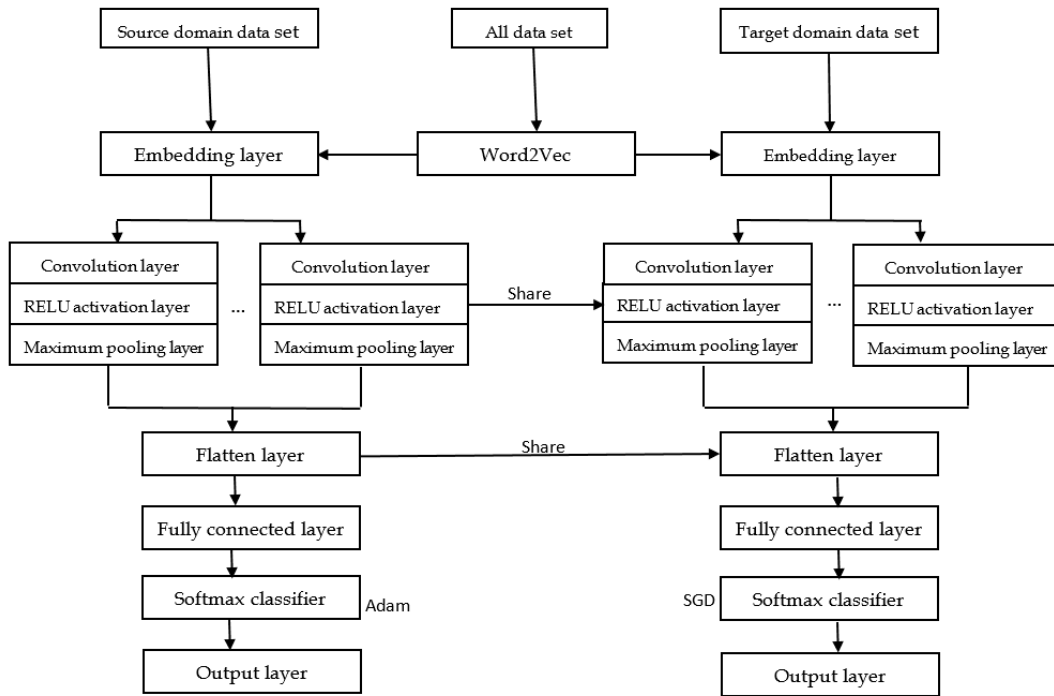


Figure 2. Framework of CNN-based transfer model.

The third layer is a pooling layer, whose role is to extract the most important features. In this paper we use the maximum pooling operation, that is, the maximum value of the feature values is set to be the main feature, defined by

$$\max(c_s) = (\max(c_s^1), \max(c_s^2), \dots, \max(c_s^{n-h+1})) \tag{9}$$

The last layer is a fully connected layer, whose purpose is to obtain the probability of each class by using the softmax classifier and to judge the class by the probability value. The corresponding formulae are given as follows,

$$\hat{y} = w \cdot \max(c_s) + b \tag{10}$$

$$P(\hat{y}_s^i) = \frac{\exp(\hat{y}_s^i)}{\sum_{i=0}^{Label} \exp(\hat{y}_s^i)} \tag{11}$$

where  $w$  is the weight of the fully connected layer,  $b$  the bias coefficient.

Next, we use a small part of labeled data fields for the target domain to fine-tune. The same first-level input layer uses Word2vec to represent the input layer as  $xt \in R_t^{n \times k}$ , where  $xt$  represents sentences in the target domain, so the input sentence is represented as

$$xt_{1:n} = xt_1 \oplus xt_2 \oplus \dots \oplus xt_n \tag{12}$$

The second layer, named the convolutional layer, uses the weight  $w_s$  of the trained convolution kernel, and obtains the feature map via using the forward propagation algorithm, the formula is given by

$$c_t = (c_t^1, c_t^2, \dots, c_t^{n-h+1}) \tag{13}$$

where

$$c_t^i = f(w_s \cdot xt_{i:i+h-1} + b_s) \tag{14}$$

Then we choose the maximum of each column of  $c_t$  by using the max pooling, and get the feature vector, denoted by  $\max(c_t)$ , i.e.,

$$\max(c_t) = (\max(c_t^1), \max(c_t^2), \dots, \max(c_t^{n-h+1})) \quad (15)$$

The weights in the last layer of the fully connected layer are fine-tuned by using the stochastic gradient descent method, the formula is given by

$$\hat{y} = w \cdot \max(c_t) + b \quad (16)$$

where  $w$  is the weight of the fully connected layer,  $b$  the bias coefficient.

Let  $P(\hat{y}_t^i)$  represent the probability of a target domain sample in the  $i$ -th class, and let the symbol label be the label of the sample, and the formula is

$$P(\hat{y}_t^i) = \frac{\exp(\hat{y}_t^i)}{\sum_{i=0}^{Label} \exp(\hat{y}_t^i)} \quad (17)$$

Above all, we use the convolutional neural network with three convolutional layers to train the source domain dataset and save the trained model structure and the weights of each layer. The first three layers do not change when training the target domain data, and only the weights of the fully connected layer are fine-tuned. The stochastic gradient descent method is used to adjust the weight value, and then the sentiment analysis is performed on the target domain.

## 4. Experiment Results and Analysis

### 4.1. Datasets

In the experiments, we select the Chinese corpus [3] and the English corpus [4] to verify the model. The Chinese corpus is the Chinese comment data sets, including Book reviews, Computer products reviews and Hotel reviews. Each domain review contains 2000 positive texts and 2000 negative texts, respectively. The detailed statistics are shown in Table 1. The English corpus is namely the Amazon review. We use the Amazon Review data set to select Amazon product reviews for four different product types: Books, DVDs, Electronics and Kitchen appliances. The setup of the sentiment classification data set is the same as [4]. Detailed statistics are shown in Table 2.

**Table 1.** Chinese experimental dataset.

Dataset	Positive Sample Number	Negative Sample Number
Book	2000	2000
Computer	2000	2000
Hotel	2000	2000

**Table 2.** English experimental dataset.

Dataset	Positive Sample Number	Negative Sample Number
Book	2000	2000
Kitchen	2000	2000
DVD	2000	2000
Electronic	2000	2000

### 4.2. Parameter Setting

The word vector is constructed with the word as a basic unit. In the training, we preprocess the datasets and take the first 100 text units. Word vector is constructed, so a sentence is converted into

a fixed size of  $100 \times 64$ . The convolutional neural network is composed of three convolution layers, a pooling layer and a fully connected layer. The specific parameters are given, i.e., the convolution kernel filter has a word length of 3, 4, 5, a width of 64, a dropout parameter of 0.2, a batch size of 32, and an epoch of 20. The parameter  $k$  is the number of the target domain labeled data for fine-tuning. Table 3 lists the parameter settings for CNN in the experiment.

**Table 3.** CNN parameter settings.

Parameter Name	Parameter Value
Sentence length	100
Number of convolution kernels	256
Convolution kernel filter word length	3, 4, 5
Word vector dimension	64
Batch size	32
Dropout	0.2
Number of iterations (epoch)	20
Target domain training data size (k)	0, 200, 500, 1000, 2000

#### 4.3. Baseline Methods

Several baseline methods are used in the experiments, such as SVM, LR, NB, SCL [4], SCL-MI [4], SGD [1] and DANN(domain-adversarial neural network) [6]. The baseline methods are based on supervised learning, such as SVM, NB and LR, directly adapt the classifiers trained from a single source to the target domain, while the gold standard is a source domain classifier which is trained on the same domain as it is tested. Table 4 shows the classification accuracy by using machine learning methods, where the source and target domains are the same. We use 80% samples for training and 20% samples for testing.

Moreover, when SCL, SCL-MI, SGD and DANN are applied, we choose the best performance as the final baseline among the possible results where a source domain is applied for training.

**Table 4.** Machine learning classification results.

	NB	LR	SVM
Book	0.8301	0.8900	0.8850
Computer	0.8550	0.8860	0.8530
Hotel	0.6062	0.8575	0.7598

#### 4.4. Experimental Results and Analysis

In this paper, the accuracy, precision, recall and F1-score are used as evaluation indicators.

##### 4.4.1. Experimental Results and Analysis of Chinese Corpus

In order to compare with our transfer learning method, we perform some machine learning methods that no target domain data are used. Table 5 shows the classification accuracy where Book→Hotel indicates that the source domain is Book, the target domain is Hotel, other rows of data are similar, and so on.

Different from the English text, the Chinese text needs to be divided into words in advance. This experiment uses the jieba token for word segmentation. In this experiment, the target domain labelled data are fine-tuned, where  $k$  is set to be 500 (250 positive examples, 250 negative examples), and the accuracy index is selected via using the 10-fold cross-validation method, as shown in Table 6.



**Table 5.** Machine learning classification results.

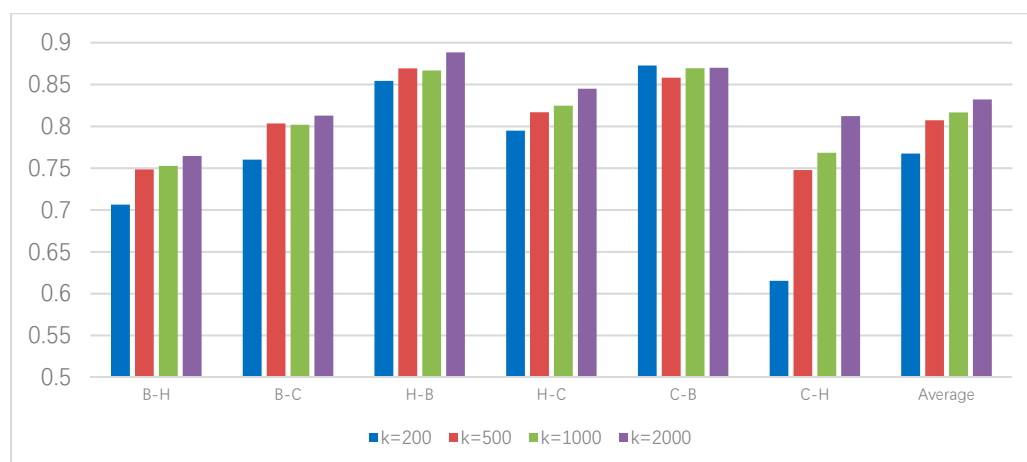
	NB	LR	SVM	CNN
Book→Hotel	0.5822	0.7750	0.7897	0.6080
Book→Computer	0.7632	0.7360	0.7312	0.6787
Hotel→Book	0.4897	0.6935	0.7107	0.7120
Hotel→Computer	0.1557	0.6992	0.7152	0.8182
Computer→Book	0.5212	0.5397	0.538	0.8267
Computer→Hotel	0.4247	0.5507	0.5412	0.6232
Average	0.4895	0.6657	0.671	0.7111

**Table 6.** Results of CNN\_FT method at  $k = 500$ .

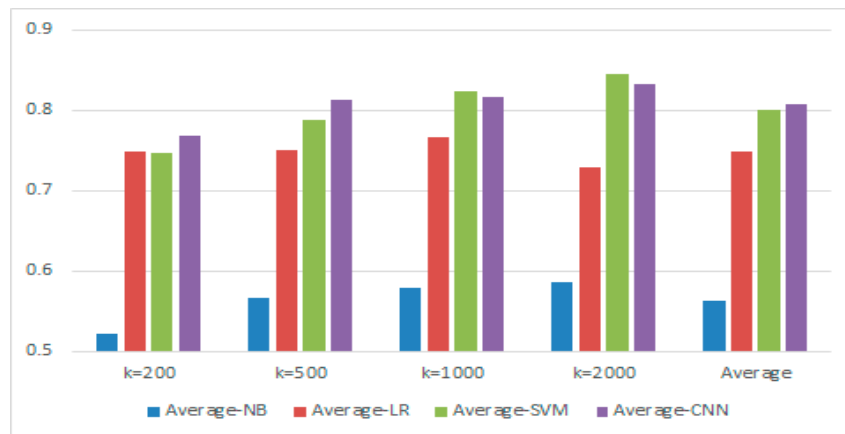
	Accuracy	Precision	Recall	F1-Score
Book→Hotel	0.7485	0.7645	0.7180	0.7407
Book→Computer	0.8034	0.8153	0.7845	0.7996
Hotel→Book	0.8691	0.8831	0.8508	0.8667
Hotel→Computer	0.8168	0.8123	0.8240	0.8180
Computer→Book	0.8580	0.8683	0.8440	0.8559
Computer→Hotel	0.7477	0.7533	0.7365	0.7448
Average	0.8072	0.8161	0.7929	0.8042

In order to indicate the impact of target domain samples for model transferring operation, we compare the size of the target domain samples added in the second training stage, where the training data size  $k$  of the target domain is set to be 0, 200, 500, 1000 and 2000, respectively. The results are shown in Figure 3.

In Figure 3, the values of  $k$  given by 200, 500, 1000 and 2000 indicate that the CNN\_FT weight parameters are adjusted by using the 200, 500, 1000 and 2000 labeled data of the target domain, respectively. Using the convolutional neural network model for transferring, as shown in Figure 3, the transferring results from the Hotel domain to the Book domain are the best, the accuracy can reach 86.91%, and the transferring results from the computer product domain data to the Hotel domain is generally, namely, 74.57%. Joining the target domain data for training, the accuracy is improved for the most training datasets. From the average results shown in the last column in Figure 3, the accuracy gradually improves after adding the labeled data of the target domain. Let  $k = 200$ , the average accuracy is improved higher, with an average increase of 5%.

**Figure 3.** Accuracy results obtained by adding different numbers of target domain samples.

In order to highlight the advantages of the CNN model, we use the traditional machine learning models for comparison. Figure 4 shows the average of the experimental results for different values of  $k$  combined with different models. Let  $k = 500$ , the comparison results are shown in Table 7.



**Figure 4.** Comparison of different models under different  $k$  values.

**Table 7.** Comparison with traditional machine learning methods ( $k = 500$ ).

	NB	LR	SVM	CNN
Book→Hotel	0.6180	0.7426	0.7317	0.7508
Book→Computer	0.6931	0.7951	0.7991	0.8014
Hotel→Book	0.7291	0.7411	0.7237	0.8788
Hotel→Computer	0.7431	0.8043	0.8103	0.8425
Computer→Book	0.7663	0.7520	0.7477	0.8625
Computer→Hotel	0.7583	0.7366	0.7463	0.7457
Average	0.7180	0.7620	0.7598	0.8136

As shown in Table 7, the experimental results show that the effect of using the CNN model for transfer learning is better than the traditional machine learning. Let  $k = 500$ , the transferring result from the Hotel domain to the Book domain is improved about 15%. The transferring result from the computer reviews to the hotel reviews is not improved obviously, and the overall average is about 5%.

According to the above experimental results, the transfer method based on the convolutional neural network can effectively improve the accuracy and can effectively solve the transfer learning problem.

#### 4.4.2. Experimental Results and Analysis of English Corpus

In this subsection, we use the Amazon corpus and compare with the experimental results of the currently published SGD [1], SCL [4], SCL-MI [4] and DANN [6]. In the case of the same model, we use Google News that is the google publicly released text to pre-train the Word2vec model, in which the word vector dimension is set to be 300 dimensions, including three million vocabulary bases, the model size reaches 4 G. The number  $k$  is set to be 50, that is to say, the number of the positive and negative texts in the target domain are 50. The experimental results are shown in Figure 5.

According to Figure 5, the CNN\_FT method achieves better results in cross-domain sentiment analysis. Clearly, the proposed CNN\_FT performs better than other methods including the state-of-the-art method SCL in most tasks except “K-E” data sets. The result of transferring from the Book domain to the DVD domain is significantly improved by 5%. The result of transferring from the Electronic domain to the Kitchen domain can reach 85.35%.



Figure 5. CNN\_FT vs. other methods.

It can be seen from the experiments that the cross-domain transfer learning method based on the convolutional neural network model is effective. The average accuracy in Chinese can reach 80.72%, and the F1 value can reach 80.42%. The improvement is quite obvious when the target domain provides about 1/10 of the labelled data. Fine-tuning the similar model in 1/10 of the annotation data yields a better performance, which can reduce the cost of the target data annotation. The CNN\_FT method proposed in this paper is much better than the traditional machine learning methods in the Chinese text. The accuracy between the Hotel and Book reviews is improved by 15%, with an average improvement is about 5%. Experimental results on the English corpus show that the proposed method is better than the traditional SGD, SCL, SCL-MI and DANN methods, and this method does not need manual query pivots and other manual operations like SCL, and thus it is convenient, simple, fast and accurate. It is not surprising to find that CNN\_FT offers a significant improvement compared to other methods. The reason is that fine-tuning can reduce the gap between domains and thus find a reasonable representation for cross-domain sentiment analysis. Experiments show that word2vec+CNN can be used to fine-tune the source domain model transfer in the full connected layer and can achieve better results in the cross-domain text sentiment analysis.

## 5. Conclusions

In this paper we explore the methods of cross-domain text sentiment analysis and propose a transfer learning method based on the multi-layer convolutional neural network model. In addition, we train a CNN model with the available labeled source data, and then fine-tune the full connected layer with a small part of target samples to solve the cross-domain sentiment analysis problems. We show that this method provides an improvement on two benchmark datasets, the Amazon reviews and the Chinese product reviews. We demonstrate through multiple experiments that CNN\_FT can better leverage the small part of labeled data in the target domain and can achieve improvements over baseline methods, that is to say, this method initially solves the problem of the absence of the labeled data in the target domain.

Next, from the perspective of adaptive learning in different domains, we will consider how to automatically extract the common features from different domains and will apply the sample-based transfer learning method to the model.

**Author Contributions:** J.M. wrote the paper and supervised the work; Y.L. performed the experiments; Y.Y. conceived and designed the experiments; D.Z. and S.L. investigated the literature and discussed the results.

**Funding:** This research was funded by the National Natural Science Foundation of China (No. 61876031), Natural Science Foundation of Liaoning Province, China (20180550921).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
2. Tahmoresnezhad, J.; Hashemi, S. Visual domain adaptation via transfer feature learning. *Knowl. Inf. Syst.* **2017**, *50*, 585–605. [CrossRef]
3. Wu, Q.; Tan, S. A two-stage framework for cross-domain sentiment classification. *Expert Syst. Appl.* **2011**, *38*, 14269–14275. [CrossRef]
4. Blitzer, J.; Dredze, M.; Pereira, F. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic, 25–27 June 2007.
5. Pan, S.J.; Ni, X.; Sun, J.T.; Yang, Q.; Chen, Z. Cross-domain sentiment classification via spectral feature alignment. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010.
6. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2015**, *17*, 2096–2030.
7. Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1253. [CrossRef]
8. Lai, W.H.; Qiao, Y.P. Spam messages recognizing methods based on word embedding and convolutional neural network. *J. Comput. Appl.* **2018**, *38*, 2469–2476.
9. Li, Z.; Zhang, Y.; Wei, Y.; Wu, Y.; Yang, Q. End-to-end adversarial memory network for cross-domain sentiment classification. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017.
10. Fortuna, L.; Arena, P.; Balya, D.; Zarandy, A. Cellular neural networks: A paradigm for nonlinear spatio-temporal processing. *IEEE Circuits Syst. Mag.* **2001**, *1*, 6–21. [CrossRef]
11. Dai, W.; Yang, Q.; Xue, Gu.; Yu, Y. Boosting for transfer learning. In Proceedings of the International Conference on Machine Learning, Corvallis, OR, USA, 20–24 June 2007.
12. Tan, B.; Song, Y.; Zhong, E.; Yang, Q. Transitive transfer learning. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, Australia, 10–13 August 2015.
13. Tan, B.; Zhang, Y.; Pan, S.J.; Yang, Q. Distant domain transfer learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
14. Blitzer, J.; McDonald, R.; Pereira, F. Domain adaptation with structural correspondence learning. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, 22–23 July 2006.
15. Long, M.; Wang, J.; Cao, Y.; Sun, J.; Philip, S.Y. Deep learning of transferable representation for scalable domain adaptation. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 2027–2040. [CrossRef]
16. Long, M.; Zhu, H.; Wang, J.; Jordan, M.I. Deep transfer learning with joint adaptation networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017.
17. Sener, O.; Song, H.O.; Saxena, A.; Savarese, S. Learning transferrable representations for unsupervised domain adaptation. In Proceedings of the 13th Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016.
18. Glorot, X.; Bordes, A.; Bengio, Y. Domain adaptation for large-scale sentiment classification: A deep learning approach. In Proceedings of the 28th International conference on machine learning, Bellevue, WA, USA, 28 June–2 July 2011.
19. Davis, J.; Domingos, P. Deep transfer via second-order markov logic. In Proceedings of the 26th International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009.
20. Bengio, Y.; Ducharme, R.; Vincent, P.; Jauvin, C. A neural probabilistic language model. *J. Mach. Learn. Res.* **2003**, *3*, 1137–1155.
21. Mikolov, T. Word2vec Project [DB/OL]. Available online: <http://code.google.com/p/word2vec/> (accessed on 20 March 2019).

22. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 3111–3119.
23. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vectorspace. In Proceedings of the International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013.
24. Severyn, A.; Moschitti, A. Twitter sentiment analysis with deep convolutional neural networks. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015.
25. Mohammad, S.M.; Zhu, X.; Kiritchenko, S.; Martin, J. Sentiment, emotion, purpose, and style in electoral tweets. *Inf. Process. Manag.* **2015**, *51*, 480–499. [[CrossRef](#)]
26. Xing, F.; Pallucchini, F.; Cambria, E. Cognitive-inspired domain adaptation of sentiment lexicons. *Inf. Process. Manag.* **2019**, *56*, 554–564. [[CrossRef](#)]
27. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs up: Sentiment classification using machine learning techniques. In Proceedings of the International Conference on Experience Methods in Natural Language Processing, Philadelphia, PA, USA, 6–7 July 2002.
28. Desai, M.; Mehta, M.A. Techniques for sentiment analysis of twitter data: A comprehensive survey. In Proceedings of the International Conference on Computing, Communication and Automation, Greater Noida, India, 29–30 April 2016.
29. Meng, J.N.; Yu, Y.H.; Zhao, D.D.; Sun, S. Cross-domain sentiment analysis based on combination of feature and instance-transfer. *J. Chin. Inf. Process.* **2015**, *79*, 74–79, 143.
30. Huang, R.Y.; Kang, S.Z. Improved EM-based cross-domain sentiment classification method. *Appl. Res. Comput.* **2017**, *34*, 2696–2699.
31. Xia, R.; Zong, C.Q.; Hu, X.L.; Cambria, E. Feature ensemble plus sample selection: Domain adaptation for sentiment classification. *IEEE Intell. Syst.* **2013**, *28*, 10–18. [[CrossRef](#)]
32. Deshmukha, J.S.; Tripathy, A.K. Entropy based classifier for cross-domain opinion mining. *Appl. Comput. Inform.* **2017**, *14*, 55–64. [[CrossRef](#)]
33. Tang, D.Y.; Qin, B.; Liu, T. Deep learning for sentiment analysis: Successful approaches and future challenges. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2015**, *5*, 292–303. [[CrossRef](#)]
34. Yu, J.F.; Jiang, J. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In Proceedings of the Conference on Empirical Methods in Natural Language, Stroudsburg, PA, USA, 1–4 November 2016.
35. Santos, C.N.; Gattit, M. Deep convolutional neural networks for sentiment analysis of short texts. In Proceedings of the 25th International Conference on Computational Linguistics, Dublin, Ireland, 23–29 August 2014.
36. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.R. Improving neural networks by preventing co-adaptation of feature detectors. *Comput. Sci.* **2012**, *3*, 212–223.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).