Combining Convolutional Neural Network and Support Vector Machine for Sentiment Classification

Conference Paper · November 2015

DOI: 10.1007/978-981-10-0080-5_13

Ruifeng Xu
Harbin Institute of Technolog , Shenzhen Graduate School

Combining Convolutional Neural Network and Support Vector Machine for Sentiment Classification

Yuhui Cao, Ruifeng Xu(✉), and Tao Chen

Shenzhen Engineering Laboratory of Performance Robots at Digital Stage,
Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China
{caoyuhuiszu,xuruifeng.hitsz,chentao1999}@gmail.com

Abstract. In recent years, the classifiers based on convolutional neural network (CNN) and word embedding achieved good performances in sentiment classification tasks. However, the CNN-based model simply uses a fully connected layer for classification and it cannot perform a non-linear classification efficiently compared to the support vector machine (SVM) classifier. Target to this problem, in this paper, we combine CNN and SVM for sentiment classification. Firstly, continuous bag of word (CBOW) model is applied to construct word embedding. CNN is then utilized to learn feature vector representation corresponding to each sentence. The learned vector representations are fed to a SVM classifier as features for sentiment classification. Evaluations on the NLPCC2014 Sentiment Classification with Deep Learning Technology Task datasets (in short, NLPCC-SCDL) show that our model outperforms the top system in the NLPCC 2014 evaluation, on both English and Chinese sides.

Keywords: Sentiment analysis · Convolutional neural network · Support vector machine

1 Introduction

With the rapid development of E-commerce, more and more people purchase items and share their reviews online. These increasing reviews are valuable to new customer for making purchase decision and to the manufactory for improving their products. Thus, the sentiment analysis techniques, which identifies the subjective comments from product review texts and determines their polarities, attracts much research interests in recent years. The majority of sentiment analysis may be divided into subjectivity classification [14] and sentiment polarity classification [16,23]. More specifically, sentiment polarity classification contains binary classification (positive and negative) [16] and multivariate classification [15,23].

Generally speaking, existing works on sentiment classification may be camped into two major approaches, namely sentiment knowledge based approach [17] and machine learning based approach [16]. The former approach mainly utilizes the sentiment lexicon, rules and pattern matching for sentiment analysis [19,20].

X. Zhang et al. (Eds.): SMP 2015, CCIS 568, pp. 144–155, 2015.
DOI: 10.1007/978-981-10-0080-5_13

Attributes to the increasing popularity of network informal language and flexible use of regular words in online product reviews, this approach has shown their limitations. The machine learning based approaches attracted increasing attentions in recent years. This approach normally uses words, Bi-grams, sentiment words, part-of-speech and information gain as features to represent the training samples. The machine learning based algorithms, such as support vector machines (SVMs), Naive Bayes (NB) and Maximum Entropy (ME), are applied to the feature vectors corresponding to the labeled dataset for training the classifier. Thus, feature engineering, a method learning features from texts, plays an important role. In recent years, the classifier based on convolutional neural network (CNN) and word embedding achieved good performances in sentiment classification. Unlike the bag of words representation, word embedding is distributed representation based on a neural probabilistic language model, which is expected to alleviate data sparseness because they are low dimensional, dense and continuous. However, the CNN-based model simply uses a fully connected layer for classification and it cannot perform a non-linear classification efficiently.

In this study, we present a CNN-SVM combined model for sentiment classification. In this model, continuous bog of word (CBOW) model is employed to construct word embedding. The CNN model is then applied to learn feature vector representations for the labeled training data. The learned feature vectors are fed to train the SVM classifier. Such a combined model is expected to combine the advantages of CNN model on feature learning and SVM model on efficient non-linear classification. Evaluations on NLPCC2014 Sentiment Classification with Deep Learning Technology Task datasets (NLPCC-SCDL) which is a sentiment labeled product review dataset, show that our combined model outperforms the CNN model and the top submitted system in the NLPCC 2014 evaluation, on both Chinese and English side. Our model achieves the highest known performance on this dataset, based on our knowledge, which shows the effectiveness of our CNN-SVM combined sentiment classification model.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work. Section 3 presents the design and implementation of our CNN-SVM combined model for sentiment analysis. Section 4 gives the evaluation results and discussions. Finally, Sect. 5 gives the conclusions.

2 Related Work

2.1 Sentiment Analysis

Sentiment knowledge based approach uses universal [10] and domain specific [21] sentiment word lexicon, or sentiment rules and patterns to discriminate sentiment polarity. Kim et al. [10] used WordNet and HowNet as sentiment knowledge to classify sentiment polarity. Tong [21] constructed a domain specific sentiment lexicon for movie reviews by manually choosing sentiment phrases. Besides, D. Tang et al. [19] proposed to build a large-scale sentiment lexicon from Twitter by following a representation learning approach and cast sentiment lexicon learning as a phrase-level sentiment classification. Strfano Baccianella et al. [1]

constructed SentiWordNet by enriching WordNet with sentiment information for sentiment classification. As for Chinese sentiment lexicons, Minlie Huang et al. [6] proposed a method for detecting new sentiment words by exploring the frequent sentiment word patterns.

Machine learning based approach employs the machine learning models to learn features of texts and use discriminate methods to classify sentiment polarity. Pang et al. [16] proposed to classify movie reviews into positive/negative by using three different classifier including Naive Bayes, Maximum Entropy and SVM. They tested different feature combinations including unigrams, unigrams+bigrams and unigrams+POS (part-of-speech) tags. SVM can be used for both binary and multiple category classification [7]. Yang and Liu [24] compared SVM with linear Least-squares, Neural Network, Naive Bayes and k-nearest neighbors for sentiment classification. They found that SVM achieved an equal performance like other classifiers in their experiments.

2.2 Deep Neural Networks

Deep neural networks models have achieved remarkable results in computer vision [12] and speech recognition [5] area. In natural language processing area, many researchers used deep neural networks to learn word embedding [13] and perform composition over the learned word embeddings for classification [4]. Bengio et al. proposed a feed-forward neural network with a linear projection layer and a non-linear hidden layer to construct a neural language model [2]. The Collobert and Weston (C&W) model [3] is another neural language model based on the syntactic context of words. It substitutes the center word of a sentence by a random word to generate a corrupted sentence as a negative sample. Mikolov et al. [13] proposed a neural language model to learn word distribution representations including words semantics.

CNN model, which is initially invented for computer vision, has shown effective to natural language processing applications such as semantic parsing [25], search query retrieval [18], sentence modeling [8] and other traditional natural language processing tasks [4]. Kalchbrenner et al. [9] proposed a dynamic convolutional neural network (DCNN) model to handle the input sentences with varying length and induced a feature graph over the sentence. Such model is capable of explicitly capturing short and long-range relations. Kim [11] presented two simple CNN models with little hyper-parameter tuning for sentence-level classification.

3 Our Approach

In this study, we present a CNN-SVM combined model for sentiment analysis. The system framework is shown in Fig. 1. Firstly, CBOW model is applied to learn word embedding from a large collection of raw text. Secondly, a CNN model is applied to construct distributed sentence representations for input labeled data. Finally, the distributed sentence feature representations are used as the features for SVM classifier training by learning the probability distribution over labels.

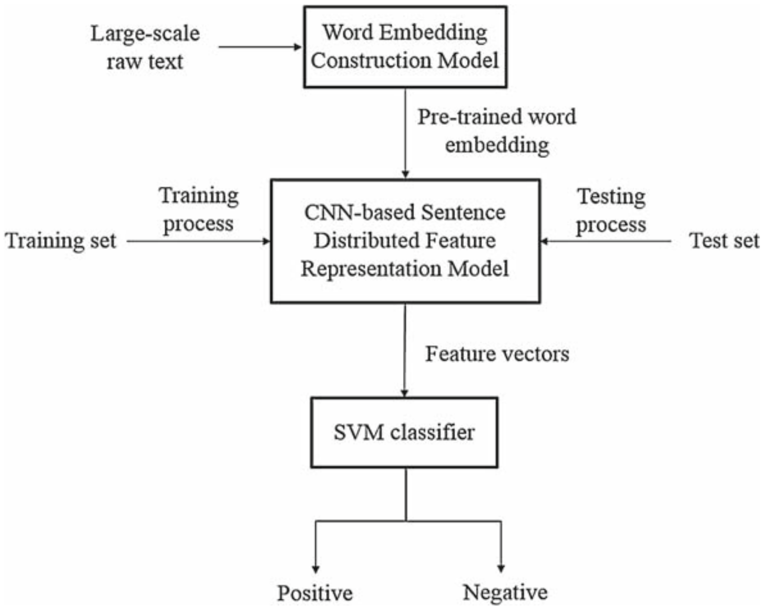

Fig. 1. Framework of CNN-SVM combined model.

3.1 Word Embedding Construction

Word embedding, wherein words are projected to a lower dimensional vector space via a hidden layer, are essentially feature learners that encodes semantic features of words in their dimensions. Mikolov et al. [13] introduced CBOW model to learn vector representations which captures syntactic and semantic word relationships from unlabeled text. The main idea is to predict a word by using its surrounding words in a context. Each word is mapped to a unique vector which is represented by a column in a matrix. The position of the word in the vocabulary is the index of the word in the matrix. The input layer of this model is the embeddings of surrounding words while the hidden layer is the concatenation of the word embedding which is used as features for predicting the target word in a sentence. Formally, given a sequence of words which are belong to a sentence in training data $w_1, w_2, \ldots, w_n$, the target of this model is to maximize the average log probability:

$$\frac{1}{n}\sum_{t=k}^{n-k} \log p(w_t|w_{t-k}, \ldots, w_{t+k}) \tag{1}$$

When the training converges, the words with similar semantic meanings are likewise close in the lower dimensional vector space.

3.2 CNN-Based Sentence Distributed Feature Representation

CNN is a kind of artificial neural network. Its weight shares network structures. It makes CNN more similar to biological neural network, and reduces the complexity of network model as well as the number of weight.

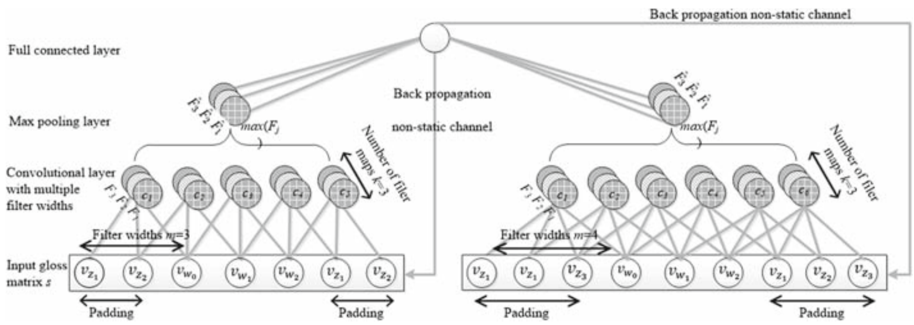

Fig. 2. A one-dimensional CNN with 2 filter widths.

As shown in Fig. 2, The one-dimensional CNN proposed by [11][1] is used for constructing distributed sentence feature representations. It includes four layers, namely input layer, convolutional layer, pooling layer, and fully connected layer. Let $w_i \in \mathbb{R}^d$ be the d-dimensional word embedding which is the i-th word in the sentence. A sentence s of length n (padded when the sentence length is under n) is represented as

$$s = [v_{z_1}, \cdots, v_{z_{m-1}}, v_{w_1}, \cdots, v_{w_n}, v_{z_1}, \cdots, v_{z_{m-1}}] \tag{2}$$

The vector obtained by concatenation is regarded as the input of the CNN model. A convolution operation involves a filter $w \in \mathbb{R}^{hk}$, which is applied to a window of h words to produce a new feature. The new feature is generated as follows:

$$c_i = f(w \cdot s_{i:i+h-1} + b) \tag{3}$$

where w and b are parameters of CNN. f is a non-linear function. $s_{i:i+m-1}$ refers to the i-th to $(i+m-1)$-th column of s. All of the possible windows of words in the sentence are applied to produce a feature map in the convolutional layer.

After convolution operation, a max-over-time pooling operation is applied to the feature map. The pooling operation takes the maximum value $\hat{c}$ in the feature map c and takes it as the feature corresponding to the particular filter. This operation constitutes the pooling layer and gets an m-dimension feature vector where m is the number of filters. The CNN model uses multiple filters with varying window sizes. These features are then transmitted to the last layer, namely fully connected layer, whose output is the probability distribution over labels.

[1] https://github.com/yoonkim/CNN_sentence.

The pre-trained word embeddings are fine-tuned via back propagation in the training processing of CNN model. Fine-tuning allows them to learn more meaningful representations of words. If the words do not appear in the pre-trained word embedding, they are initialized randomly. The vectors in the fully connected layer of CNN are regarded as the distributed sentence feature representations, and then these sentence representations are regarded as feature vectors in a SVM classifier.

3.3 CNN-based SVMs Classifier

SVM is a kind of supervised machine learning model for classification and regression analysis. Given a set of training examples, each marked for belonging to one of two categories, the SVM training algorithm builds a model that assigns new examples into one category or the other. Thus, SVM is a non-probabilistic binary linear classifier.

The CNN model for sentence classification, which is proposed by Kim [11], used a fully connected softmax layer as the classification layer. However, this classification layer is too simple to sentiment classification task. Fortunately, the output values of CNN pooling layer can be regarded as feature vectors for the input sentence. They may used as the input of other classifiers.

In this paper, we propose a CNN-based SVM classifier which treats CNN as the automatic feature learner and SVM as the sentiment classifier. The outputs of CNN, the distributed feature representations for the input sentences, are regarded as features in SVM. The SVM classifier is trained by using sentiment labeled sentences. When this model is applied to sentiment classification, the input sentences are transferred to distributed feature representations and then fed to SVM classifier for classification. Such a combined model is expected to combine the advantages of CNN and SVM.

4 Evaluation and Discussion

4.1 Experiment Settings

The dataset of NLPCC2014 Sentiment Classification with Deep Learning Technology Task (NLPCC-SCDL) is adopted to evaluate the proposed combined sentiment classification model. The NLPCC-SCDL task is designed to evaluate the sentiment analysis models based on deep learning. This dataset includes Chinese and English product reviews (Chinese side and English side for short, respectively) from multiple domains including book, DVDs and electronics. The statistics of NLPCC-SCDL are given is Table 1.

As for the evaluation metric, the ones adopted in NLPCC-SCDL evaluation are used here for fair evaluation and comparison. The metric is based on precision (P), recall (R), and $F1$ measure.

Let TP be the number of correctly classified positive samples, FP be the number of the falsely classified negative samples, TN be the number of the correctly classified negative samples, and FN be the number of the falsely classified positive samples. The metrics are defined as follows:

Table 1. The statistics of NLPCC-SCDL dataset

	Training set		Test set	
	Positive	Negative	Positive	Negative
English dataset	5000	5000	1250	1250
Chinese dataset	5000	5000	1250	1250

$$
\begin{aligned}
P_{pos} &= \frac{TP}{TP+FP} \qquad & P_{neg} &= \frac{TN}{TN+FN} \\
R_{pos} &= \frac{TP}{TP+FN} \qquad & R_{neg} &= \frac{TN}{TN+FP} \\
F1_{pos} &= \frac{2 \times P_{pos} \times R_{pos}}{P_{pos}+R_{pos}} \qquad & F1_{neg} &= \frac{2 \times P_{neg} \times R_{neg}}{P_{neg}+R_{neg}}
\end{aligned} \tag{4}
$$

where the *pos* subscript refer to positive class and *neg* subscript refer to negative class.

Furthermore, we use classification accuracy (Acc) in the training process because single metric in the training process is helpful to parameter optimization in our proposed model.

$$
Acc = \frac{TP+TN}{TP+FP+TN+FN} \tag{5}
$$

For pre-trained word embeddings, in the Chinese experiment, we trained the CBOW model by 50 million unlabeled Chinese sentences and finally obtained 399,059 word embeddings. In the English experiment, we use the publicly available vectors trained on 100 billion words from Google News by the CBOW model. All of the word embeddings have dimensionality of 300.

The CNN model proposed by Kim [11] is used. Meanwhile, we use SVM classifier with radial basis function in this experiment.

Four systems/algorithms are used as comparison systems. The first model is based on sentiment knowledge features, labeled as $Senti$. The second model, TF, refer to the model using word frequency features. The third one is a CNN-based model proposed by Kim which uses a fully connected layer as a classifier. The last one, labeled as $NLPCC - SCDLbest$, refers to the system which achieves the best performance in the NLPCC-SCDL evaluation [22]. For fair comparison, the adopted labeled dataset and evaluation metric are the same as in NLPCC-SCDL evaluation.

4.2 Experimental Results and Analysis

Training Stage. In the training process, the parameter optimization for CNN and SVM are conducted through closed testing on training data with 5-fold cross validation. Accuracy, Acc, is adopted in the training stage as the metric because the single metric is helpful to speed the parameter optimization. In CNN model,

Table 2. The parameter optimization results of CNN on training dataset with 5-fold cross validation (Chinese Side)

Filter \ Hidden unit	50	100	150	200
[1, 2, 3]	0.758	0.772	0.769	0.768
[2, 3, 4]	0.757	0.764	0.760	0.772
[3, 4, 5]	0.776	0.772	0.777	0.776
[4, 5, 6]	0.761	**0.778**	0.767	0.764
[5, 6, 7]	0.748	0.768	0.770	0.768
[6, 7, 8]	0.765	0.766	0.772	0.774

Table 3. The parameter optimization results of CNN on training dataset with 5-fold cross validation (English Side)

Filter \ Hidden unit	50	100	150	200
[1, 2, 3]	0.855	0.860	0.863	0.857
[2, 3, 4]	0.860	0.861	0.867	0.865
[3, 4, 5]	0.861	0.865	0.859	0.862
[4, 5, 6]	0.857	0.858	0.860	0.859
[5, 6, 7]	0.864	0.861	**0.868**	0.859
[6, 7, 8]	0.856	0.858	0.859	0.865

we adjust the parameter $Filter$ and $Hidden\ unit$ which $Filter$ is the window sizes of filter and $Hidden\ unit$ is the number of each window size filter. The product of $Filter$ and $Hidden\ unit$ is the size of feature vector which is the output of CNN pooling layer. The parameter optimization results of CNN are shown in Tables 2 and 3 on Chinese side and English side, respectively.

Table 2 shows that the classification accuracy on the Chinese dataset achieves the highest value of 0.778 when $Filter$ is set to [4, 5, 6] and $Hidden\ unit$ is set to 100. The highest accuracy value of 0.868 on the English dataset is achieved when $Filter$ is set to [5, 6, 7] and $Hidden\ unit$ is set to 150. Since the size of $Filter$ and $Hidden\ unit$ determine the time complexity of CNN model, these two parameters are selected within the appropriate range. Hence, in testing process, we let the parameter $Filter$ be [4, 5, 6], $Hidden\ unit$ be 100 and $Filter$ be [5, 6, 7], $Hidden\ unit$ be 150 of CNN for Chinese and English side, respectively.

In the process of parameters adjustment for SVM, the grid searching range of each parameter is: $\sigma = \left[10^{-3}, 10^{-2}, \ldots, 10^{2}, 10^{3}\right]$ and $C = \left[10^{-3}, 10^{-2}, \ldots, 10^{2}, 10^{3}\right]$. We tried 49 combinations on Chinese and English training dataset. The highest accuracy is achieved when C is 0.1 and $Gamma$ is 0.01 for Chinese dataset, C is 0.01 and $Gamma$ is 0.1 for English dataset, respectively.

Table 4 gives the closed testing results by our proposed model and the four comparison models.

It is observed that our proposed model achieves the highest classification accuracy on both Chinese side and English side. The 0.009 and 0.025 accuracy

Table 4. Classification accuracy by different models on training datasets

Dataset	Senti	TF	NLPCC2014 Best	CNN-based Model	Our model
Chinese	0.663	0.700	0.714	0.769	**0.778**
English	0.756	0.814	0.792	0.843	**0.868**

Table 5. Performance on NLPCC-SCDL testing dataset

Dataset	Model	Positive			Negative		
		P	R	F1	P	R	F1
Chinese dataset	NLPCC2014 best	0.758	0.789	0.773	0.780	0.748	0.764
	CNN-based model	0.759	0.796	0.777	0.781	0.749	0.764
	Our model	**0.766**	**0.806**	**0.785**	**0.795**	**0.754**	**0.774**
English dataset	NLPCC2014 best	0.856	0.866	0.861	0.864	0.855	0.860
	CNN-based model	0.871	0.860	0.865	0.860	0.873	0.866
	Our model	**0.890**	**0.886**	**0.888**	**0.886**	**0.891**	**0.889**

improvement on Chinese and English from the CNN-based model using fully connected layer are obtained, respectively. This shows the contribution of CNN and SVM combined model. Meanwhile, the achieved accuracy is obviously higher than the ones achieved by the $NLPCC - SCDLbest$, $Senti$ and TF. It is also observed that the achieved accuracies on English side are higher than Chinese side by all of the five tested systems. These results indicate that the sentiment analysis of Chinese is more complex and difficult.

Testing Process. In the testing stage, we use the optimized parameters of CNN and SVM. The final adopted parameters in testing stage are: $Filter$ is [4, 5, 6], $Hidden\ unit$ is 100, C is 0.1 and $Gamma$ is 0.01 for Chinese side and $Filter$ is [5, 6, 7], $Hidden\ unit$ is 150, C is 0.01 and $Gamma$ is 0.1 for English side. For the comparison CNN-based model, the same parameters are utilized. The achieved performances on Chinese and English dataset by different models are listed in Table 5. Noted that, the public evaluation metrics adopted in NLPCC-SCDL are utilized here.

It is observed that our proposed CNN and SVM combined model outperforms the CNN-based model and the NLPCC-SCDL best system. Compared to the CNN-based model, the $F1$ values on positive and negative categories by our model are increased for 0.8 % and 1.0 %, 2.3 % and 2.3 % on Chinese and English datasets, respectively. This result clearly shows that contribution of our CNN-SVM combined model by appending a SVM classifier.

Compared to the NLPCC-SCDL best system, our combined model obtains obvious improvement. The increments of $F1$ values on positive and negative are 1.2 % and 1.0 % for Chinese dataset, 2.7 % and 2.9 % for English dataset, respectively. The achieved performances are the highest one on NLPCC-SCDL dataset, based on our knowledge. These results show the effectiveness of our proposed model for sentiment analysis.

5 Conclusion

In this paper, we propose a CNN-SVM combined model for sentiment classification. This model treats CNN as a feature learner which automatically learns the feature representations for the input sentences. It is shown helpful to improve the feature representation and feature learning. Moreover, the combined model takes the advantages of SVM classifier on classification efficiency in order to improve the classification capability of CNN. The experiment results on the NLPCC2014 Sentiment Classification with Deep Learning Technology Task dataset show that our proposed combined model outperforms CNN model and the best system in the NLPCC 2014 evaluation, which shows the effectiveness of our proposed CNN-SVM combined model for sentiment classification.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. 61370165, 61203378), National 863 Program of China 2015 AA015405, Natural Science Foundation of Guangdong Province S2013010014475, Shenzhen Development and Reform Commission Grant No.[2014]1507, and Shenzhen Peacock Plan Research Grant KQCX20140521144507925

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17–23 May 2010, Valletta, Malta (2010)
2. Bengio, Y., Ducharme, R., Vincent, P.: A neural probabilistic language model. J. Mach. Learn. Res. **3**, 1137–1155 (2003)
3. Collobert, R., Weston, J.: A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine Learning (ICML), pp. 160–167. ACM (2008)
4. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
5. Graves, A., Mohamed, A.-R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6645–6649. IEEE (2013)
6. Huang, M., Ye, B., Wang, Y., Chen, H., Cheng, J., Zhu, X.: New word detection for sentiment analysis. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), pp. 531–541. Association for Computational Linguistics, Baltimore,Maryland, June 2014

7. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398. Springer, Heidelberg (1998)
8. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, 22–27 June 2014, Baltimore, MD, USA, vol. 1, pp. 655–665 (2014). Long Papers
9. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), pp. 655–665. Association for Computational Linguistics, Baltimore, June 2014
10. Kim, S.M., Hovy, E.: Automatic detection of opinion bearing words and sentences. In: Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP), pp. 61–66 (2005)
11. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751, October 2014
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held 3–6 December 2012, Lake Tahoe, Nevada, USA, pp. 1106–1114 (2012)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held 5–8 December 2013, Lake Tahoe, Nevada, USA, pp. 3111–3119 (2013)
14. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL), 21–26 July 2004, Barcelona, Spain, pp. 271–278 (2004)
15. Pang, B., Lee, L.: Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL), pp. 115–124. Association for Computational Linguistics (2005)
16. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86 (2002)
17. Salvetti, F., Lewis, S., Reichenbach, C.: Automatic opinion polarity classification of movie. Colorado Res. Linguist. **17**, 2 (2004)
18. Shen, Y., He, X., Gao, J., Deng, L., Mesnil, G.: Learning semantic representations using convolutional neural networks for web search. In: Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion, pp. 373–374. International World Wide Web Conferences Steering Committee (2014)
19. Tang, D., Wei, F., Qin, B., Zhou, M., Liu, T.: Building large-scale twitter-specific sentiment lexicon: a representation learning approach. In: Proceedings of the 25th International Conference on Computational Linguistics (COLING), pp. 172–182 (2014)

20. Tekiroğlu, S.S., Özbal, G., Strapparava, C.: A computational approach to generate a sensorial lexicon. In: Proceedings of the 25th International Conference on Computational Linguistics (COLING), p. 114 (2014)
21. Tong, R.M.: An operational system for detecting and tracking opinions in on-line discussion. In: Working Notes of the ACM SIGIR 2001 Workshop on Operational Text Classification, vol. 1, p. 6 (2001)
22. Wang, Y., Li, Z., Liu, J., He, Z., Huang, Y., Li, D.: Word vector modeling for sentiment analysis of product reviews. In: Zong, C., Nie, J.-Y., Zhao, D., Feng, Y. (eds.) NLPCC 2014. CCIS, vol. 496, pp. 168–180. Springer, Heidelberg (2014)
23. Xu, L., Lin, H., Zhao, J.: Construction and analysis of emotional corpus. J. Chin. Inf. Process. **22**(1), 116–122 (2008)
24. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 42–49. ACM (1999)
25. Yih, W.-T., He, X., Meek, C.: Semantic parsing for single-relation question answering. In: Proceedings of the 53rd Annual Meeting on Association for Computational Linguistics (ACL) (2014)