Journal of Computers Vol. 30 No. 4, 2019, pp. 130-139
doi:10.3966/199115992019083004012

Semantic Representation Based on Clustering and Attention Mechanism to Identify Deceptive Comment Models

Jian-Xin Zhang[1], Xue-Dong Du[1*], Bin-Guo Wang[1]

[1] Provincial Computer Experimental Teaching, Demonstrating Center of Shandong Province,
Department of Computer Science and Engineering, Shandong University of Science and Technology,
Qingdao, 266590, China
zhangjianxin157@163.com, duxuedongstd@163.com, wangbg866@163.com

Received 13 January 2019; Revised 3 March 2019; Accepted 12 March 2019

Abstract. Deceptive reviews of products influence customer's decision and damage the reputation of the product. Most of the existing methods for detecting fraudulent comments use feature design, which is difficult to learn the potential semantics of comments. We propose a neural network based on clustering and attention mechanism to learn the semantic representation of reviews. Specifically, we use DBSCAN to discover the semantic groups in the word embedding space, and then construct the semantics of different semantic groups through the attention mechanism. The model computes the representations of the semantic units and combine them into the sentence representation. In feature selection, we perform feature combination to improve performance. Then we demonstrate the effectiveness of the proposed model through Amazon dataset experiments. In the experiment, the model surpassed the state-of-art method.

Keywords: attention mechanism, clustering, convolutional neural networks, deceptive review, semantic units

1 Introduction

It has become increasingly common for people to read online reviews before deciding to purchase a product [1]. For the business community, some corporate employers write positive reviews to damage the object's reputation. Deceptive reviews refer to the unrealistic promotion or incitement of products to achieve the purpose of influencing the user's opinions or consumer behavior [2]. Deceptive reviews are widely distributed in accommodation, travel and other review sites or e-commerce sites. There are 2%-6% deceptive reviews at Orbitz, Priceline, Expedia, Tripadvisor et al. [3-4]. It is difficult for customer to distinguish deceptive reviews. Therefore, identifying deceptive reviews has become a network security issue that needs to be solved urgently.

Over the years, many studies have proposed various methods for detecting online reviews and have achieved high detection accuracy [4-6]. The main task in this field is to distinguish between false and real reviews. Traditional methods use machine learning algorithms to build classifiers. In this direction, most studies have focused on designing effective features to improve classification performance [2-5]. However, due to the hidden nature and diversity of deceptive reviews. For example, there must be a variable number of error cases in the reviews data set manually annotated by humans, which will affect the classifier [5]. In view of the current natural language processing tasks, models based neural network have good performance. They can learn semantic representations [6, 10].

In this work, we try to overcome the shortcomings of manual annotation, and get more accurate semantic representation of text. We present a neural network model based on semantic clustering and attention mechanism (CANN) to identify deceptive reviews. We use a fast clustering algorithm based on density peak search to obtain the semantic groups in the word vector space. And the attention mechanism

* Corresponding Author

is used in the process of constructing the semantic units to obtain the accurate representation. We optimize the small batch samples iteratively through backpropagation algorithm. This model expands the word vector by using multi-scale semantic information to better understand the semantics of text.

The major contributions of the work presented in this paper are as following:

- We propose a neural network model for learning text semantic representation by clustering extended semantic units. To learn the semantics of sentence, we extend the semantic unit through preset threshold.
- We construct semantic units by adding the attention mechanism, which can accurately acquire the whole semantic of the sentence.
- We use multiple text features and make a feature combination to further improve the accuracy of our model.

2 Related Work

In recent years, many technologies and methods have been used to detect deceptive reviews. All these studies have achieved remarkable result. According to the different features of text, these methods can be roughly divided into three categories: syntax analysis, semantic analysis, stylistic and metadata analysis.

Deceptive reviews detection method based on syntax analysis refers to the analysis of the word bag features and POS features. Mukherjee et al. [4] use the word bag features and POS features to obtain 65.6% and 67.8% accuracy on the hotel and hotel domain datasets by using the support vector machine classifier. Ott [8], Shojaee [9] and Li [10] use the word bag feature, POS features and stylistic features to obtain the detection accuracy of 84%-89.6% on the data set constructed by the crowdsourcing platform. When the size of the annotated data is small, the support vector machine classifier has more outstanding detection performance than other classifiers.

The semantic analysis-based model use feature analysis or semantic representation methods to extract or abstract the information on the semantic level. Li et al. [10] use the word vector as input. They use convolutional neural networks to learn the semantic representation of the review text. Raymond et al. think that deceptive reviews exist in the case of mutual copying. They detect deceptive reviews by identifying semantically duplicated comments. The advantage of the semantic analysis model is that it does not rely on labeled data. The disadvantage is that the rules of the heuristic method are relatively simple. It is easy to misjudge the comment text that is semantically similar to the deceptive reviews.

Ott and Li et al. [8] use the LIWC method combined with the word bag feature to extract stylistic features in deceptive reviews detection. The detection performance is better than the word bag feature alone. Jindal and Liu et al. [2, 7] obtain 63%-78% AUC values on Amazon datasets by combining stylistic feature and metadata. Researchers such as Li, Hammad [11], and Mukherjee [12] use metadata features in their research to analyze data in review texts.

The representation learning model based on Neural Network have proven to be effective in task-specific feature engineering [13]. Compared with feature engineering, representation learning does not require much prior knowledge. As a continuous real-valued vector, semantic representations can be applied as features to various natural language processing tasks [14-16], such as POS markup, block, Named Entity Recognition [14, 17], semantic role tagging, parsing [18], language modeling [19-20], sentiment analysis task [21-22] and text classification [23].

3 Semantic Model Based on Clustering and Attention Mechanism

In view of the shortcomings of traditional methods, we cluster pre-trained word vectors and construct semantic units through attention models to identify deceptive comments. As shown in Fig. 1, we use the DBSCAN method to cluster word vectors in the word embedding space, and use the attention model to calculate the semantic group similar to the semantic unit. Among them, we use KL-divergence to calculate the weight of the semantic group. Finally, the semantics of the word vector is constructed by context, and the representation of the semantic unit in the text is calculated. We combine the extended matrix with the projection matrix, and feed it to the convolutional layer, and extract high-level local features. Finally, we use softmax loss function as the classifier [24].

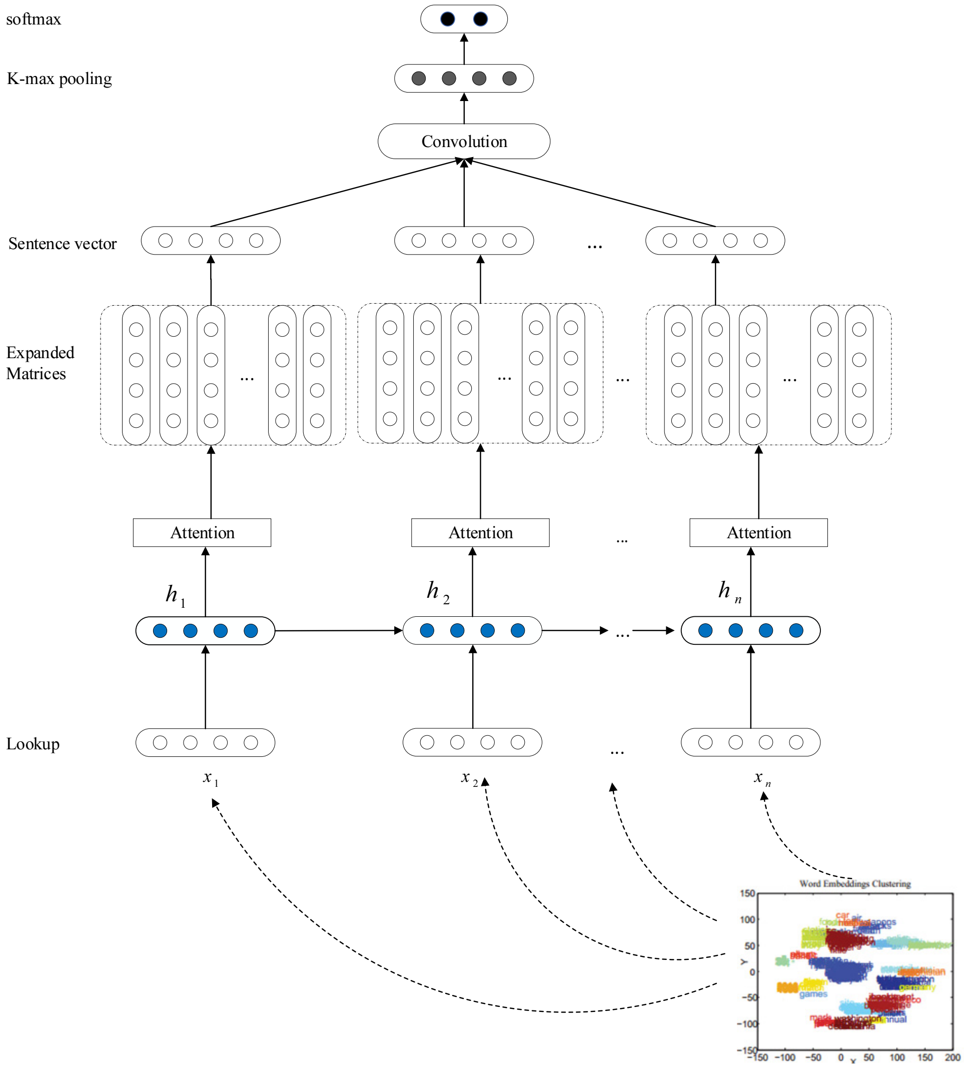

Fig. 1. Semantic model based on clustering and attention mechanism

3.1 Semantic Clustering

In the embedding space, the clustering method can discover the semantic group because the semantic relationship between words and words is related [25]. When implemented the clustering method, we found that the number of word vectors in a word is huge, and the number of semantic groups cannot be predicted. Therefore, we try to use a fast algorithm based on density peak search to solve these problems [26].

The main idea of the density-based clustering method is to find high-density regions separated by low-density regions. This is consistent with the distribution characteristics of the word vector. The clustering algorithm assumes that the cluster centers are surrounded by points with lower local density, and the distance between them and any point with higher local density is relatively large. In this algorithm, it is mainly necessary to calculate two data points: local density ρ_i, and the distance from the higher density point δ_i.

The local density is defined as follows:

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \quad \textbf{(1)}$$

among them

$$\chi(\cdot)=\begin{cases}1, if\ d_{ij}<d_c \\ 0, otherwise\end{cases} \tag{2}$$

d_{ij} is the distance between data points and d_c is the cutoff distance.

The meaning of the formula is to find the number of data points that are less than the cutoff distance d_c from the i-th data point.

The distance from the higher density point δ_i is calculated:

$$\delta_i=\begin{cases}\min_{j:\rho_j>\rho_i}(d_{ij}), if \rho_j<\rho_{\max} \\ \max_j(d_{ij}), otherwise\end{cases} \tag{3}$$

3.2 Semantic Unit Detection Based on Attention Mechanism

For text, the method of obtaining semantic representations have an important problem: the semantics of text is usually determined by some key phrases. If simply semantically represent the word, it may cause ambiguity and affect the semantic representation of the entire statement. Therefore, semantic unit detection based on attention mechanisms is useful. It captures significant local information.

The main idea of semantic unit detection is to define a convolution-like operation to semantically synthesize the word vector from the context. Among them, we use a vector matrix with variable width to capture significant local information [23].

The essence of the operation is one-dimensional convolution, which is defined as follows:

$$[seu_1, seu_2, ..., seu_{l-m+1}]=WM\otimes E \tag{4}$$

among them,

$$seu_i=\sum_{j=1}^{|PM^i|}WM_j^i \tag{5}$$

In order to more accurately identify the semantic units, attention mechanisms are referenced into the model. We use KL-divergence as the weight of the semantic group, and the KL-divergence value represents the characteristics of the segmentation semantic group. In fact, we try to use TF-IDF as a weighting method during the experiment. However, the experimental effect is not as good as KL-divergence. We use the preset distance threshold as a constraint to select the semantic unit that satisfies the condition. Among them, we use Euclidean distance. Therefore, we calculate Euclidean distance between semantic units and semantic groups. For a semantic unit, we select the semantic group close to it through the attention mechanism. If the distance between the semantic unit and the nearest word vector is less than the threshold, then the nearest word vector extension vector matrix is selected, otherwise discarded. Therefore, semantic groups are used as supervisory information to obtain more accurate features.

3.3 Architecture Description

We extract local features through a convolutional layer after extending the comment text. The calculation of the convolutional layer is defined as equation (6), and the inner product of the convolution kernel and the input matrix vector is calculated [23].

$$C=\begin{pmatrix}c_1\\c_2\\\cdots\\c_{d/2}\end{pmatrix}=\begin{pmatrix}k_1\\k_2\\\cdots\\k_{d/2}\end{pmatrix}\otimes\begin{pmatrix}X_1\\X_2\\\cdots\\X_{d/2}\end{pmatrix}^T \tag{6}$$

In order to obtain the most relevant global features with fixed length, we use K-max pooling to down sample the features, as follows:

$$\hat{C} = \max^{(k)}(C) \quad \textbf{(7)}$$

We use tangent function to compute the feature map C, and get the feature representation of input comment text.

$$\hat{f} = \tanh(\hat{C}) \quad \textbf{(8)}$$

After the above hierarchical sequence, a fixed size semantic representation is obtained. At the last level of the model network, it is linked to weights W_z , as follows:

$$\varphi(x_i, w_z) = w_z \hat{f} \quad \textbf{(9)}$$

In order to transform vectors into probability distributions, we use the softmax function. Each component of the output vector can be regarded as the score of the tag.

$$p(c_j \mid \chi_i, w_z) = \frac{\exp(\varphi_j(x_i, w_z))}{\sum_{j=1}^{|C|} \exp(\varphi_j(x_i, w_z))} \quad \textbf{(10)}$$

We use cross-entropy function in network training to minimize the cross-entropy of predictive distribution. Because the cross-entropy function has been proved to be able to accelerate the back-propagation algorithm, and provide good overall network performance [27], especially for classification tasks.

3.3 Multi-feature Fusion

We add two types of features to the proposed model. In addition to the text content, we use the metadata of the comment as a feature of the model. Metadata is a feature that comments in addition to text content, such as publication date, time, comment rating, comment ID, product ID, and feedback information. The reason why we use a variety of features is because the deceptive reviews in the real world are widely distributed and large in magnitude, and it is difficult to accurately label and have certain examples of misjudgment. This creates a problem of smaller corpus in the field of deceptive reviews. Although more and more tagged data has appeared under the efforts of predecessors, the types of data vary greatly, and the effect of the deep learning model cannot be fully utilized. Therefore, establishing such a model can alleviate this problem.

4 Experiments

4.1 Experiment Setup

The dataset we use is a review from the amazon.com e-commerce site. The review data set (https://www.cs.uic.edu/~liub/FBS/fake-reviews.html) was collected from the amazon.com in June 2006, including books, music, industrial products and other fields. The data includes 5.8 million reviews, 2.14 million users, 6.7 million products. So, it is reasonable as an experimental data set.

First, we divide the data into training sets, validation sets, and test sets. Then we split the sentences and use NLTK (http://www.nltk.org/) for word segmentation. We use two different training word vectors in experiment to compare. The data used are publicly available, as shown in Table 1.

Table 1. Method of training word vectors

Embeddings	Glove	Word2Vec
Training corpus	Wikipedia/Gigword	Google News
Dimension	50	300
Vocabulary size	400,000	3,000,000

GloVe: GloVe is an unsupervised learning algorithm for obtaining vector representations of words, proposed by Pennington et al. [28]. GloVe perform statistical training on a corpus of global word summaries. The results show the linear substructure of the word vector space. The corpus contains 6 billion words, built by Wikipedia and Gigaword.

Word2Vec: Word2ve was created by a research team led by Google's Tomas Mikolov. Word2Vec is based on the context of every word in the text, mapping each word to the same coordinate system. The relationships between these words are contextually related, and they are sequential.

4.2 Baseline Method

We compare the proposed model with the following baseline method.

CNN: Ye Zhang [29] proposed convolutional neural networks for sentence classification models.

Tanh-RNN: Ma et al. [30] proposed tanh-RNNs as a basic recurrent neural network model with an implicit layer.

MPIPUL: The method applies DPMM [31] to cluster spy samples. Then combine the population and individual strategies to predict and classify spy samples. Finally, they use labeled data to train learning support vector machine classifiers.

LSTM-1: This method [32] is a long-term and short-term memory network model with a hidden layer.

Bi-LSTM: The bidirectional LSTM model is a variant of the LSTM model and it has good performance in many natural language processing tasks.

SVMs: Support Vector Machines map feature vectors to higher-dimensional spaces to establish maximum-interval hyperplanes, which maximizes the separation of data points of different categories and minimizes classification errors.

4.3 Analysis of Results

The experimental results show that our cluster-based convolutional neural network model is much more accurate than traditional algorithms. The accuracy and F1 values of the proposed method exceed the proposed baseline method. The main reasons are attributed to the following three aspects: (1) using the Bagging model to fuse multiple text features and improving the accuracy; (2) using the density-based peak search algorithm to capture deep internal connections; (3) using the multiple scale semantic unit enables the model to better understand text information.

From Table 2, it can be found that more complex models like tanh-RNN and Bi-LSTM are not as simple as CNN. Overfitting is the main reason. Bi-LSTM achieved 91% F1 on the training data, but the results for the test data were low. From small data sets, a neural network model with many parameters is not necessarily a good choice.

Table 2. Comparison with deceptive comment recognition algorithms

Models	Accuracy	Recall rate	F1
CANN-Glove	0.832	0.937	0.927
CANN-WordVec	0.846	0.942	0.936
CNN	0.789	0.775	0.782
tanh-RNN	0.689	0.924	0.884
LSTM-1	0.652	0.928	0.913
Bi-LSTM`	0.613	0.942	0.913
MPIPUL	0.813	0.842	0.824
SVMs	0.656	0.685	0.657

To demonstrate the importance of the model in this paper compared to baseline, we designed a 5-fold cross-validation experiment. The comparison results are shown in Fig. 2. From the experimental results in Fig. 2, we performed a T test [33], and the values are shown in Table 3. As can be seen from Table 3, all p values are less than 0.01, which indicates that the method of this paper is clearly superior to other methods. Overall, comparing the results in Fig. 2 with the values in Table 3, the model in this paper is valid.

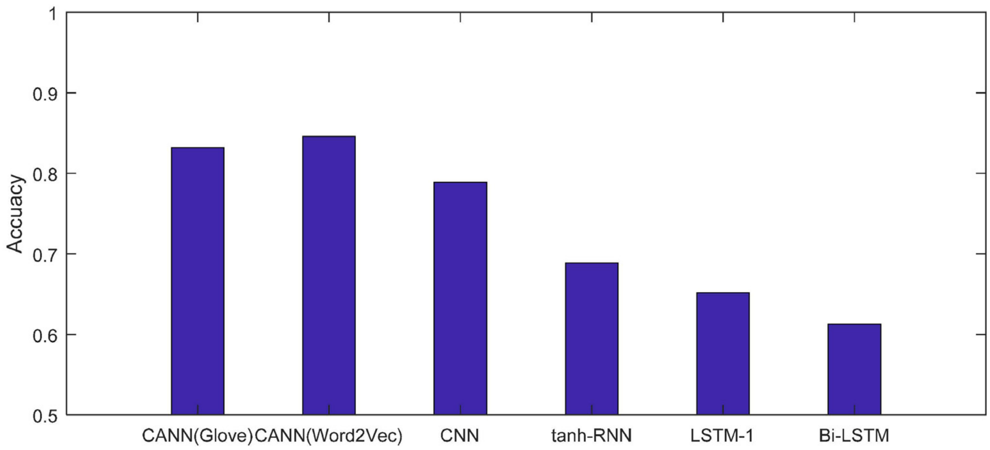

Fig. 2. Experimental results (mean accuracy ± standard deviation)

Table 3. The p-values of T-test

Models	CANN	LSTM	Tanh-RNN
Glove	0.001424	0.000352	0.004602
Word2Vec	0.000351	0.000375	0.002051

We consider the impact of different clustering algorithms on deceptive comment detection. We compare two representative clustering algorithms: K-means algorithm and DBSCAN algorithm.

From the experimental results Table 4, the model based on DBSCAN clustering algorithm is better than the K-means algorithm. At the same time, we can also see that using clustering algorithm is better than not using clustering algorithm. This shows that the clustering algorithm helps to understand the semantics of the text and improve the recognition ability of the model.

Table 4. Influence of different clustering algorithms

Models	Accuracy	Recall rate	F1
No-cluster	0.785	0.940	0.885
K-means	0.832	0.936	0.933
DBSCAN	0.846	0.942	0.936

In addition, we also compared two attention models, including based on KL-divergence and based on TFIDF notation. The experimental results are shown in Table 5. It can be seen that the attention model based on KL-divergence is superior to other models. At the same time, we also found that using the attention mechanism can improve the model recognition rate.

Table 5. Comparison of multiple attention models

Models	Accuracy	Recall rate	F1
No-attention	0.813	0.871	0.875
TF-IDF	0.822	0.940	0.944
KL-divergence	0.835	0.944	0.912

4.4 Hyperparametric Influence

We compare the effects of two parameters by experiment, which are the window width and the preset threshold. To obtain a semantic unit representation, we use a multiscale window matrix. For example, if you use m window matrices, they range in width from 2 to $m+1$. Fig. 3 shows the experimental results of the window width. For CANN (Word2Vec), when the window width is two, the classification accuracy is the highest. We can also conclude that small windows may lose some key information leading to ambiguous phrase formation, while large-sized windows may generate noise and reduce accuracy.

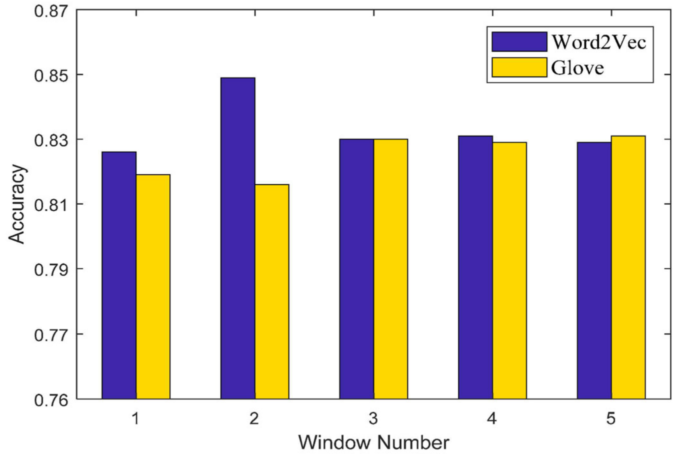

Fig. 3. The number of multi-scale semantic expansion window matrices

As described in Section 3.1.2, the representation of the semantic unit is a semantic synthesis of the word vector. For each semantic unit, the most recent word vector is selected according to the preset distance threshold. Considering the impact of the threshold on the performance of the algorithm, we conduct an experiment. The result is shown in Fig. 4. When d is too small, only a few word vectors fit the distance. However, when d is too large, an unrelated word vector will appear. In addition, the different thresholds of the pre-training vector method have different effects on the model.

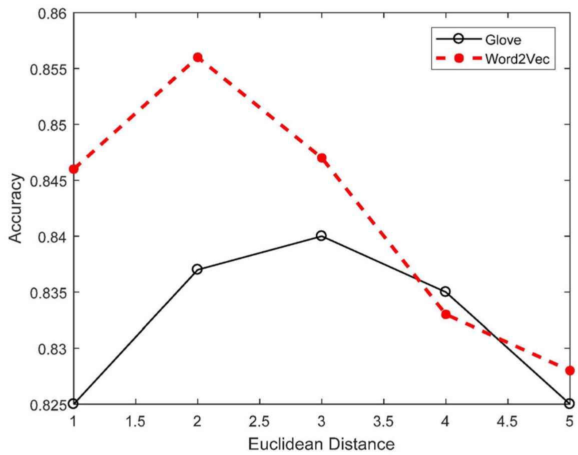

Fig. 4. Influence of Euclidean distance preset threshold

From Fig. 3 and Figure 4 we can see that the performance of the model varies with the pre-training vector. This is because the corpus used in the two data sets, the dimension of the word vector, the size of the vocabulary, and the vocabulary coverage of the word vector affect the accuracy of the classification.

5 Conclusion

In this paper, the semantic representation algorithm based on clustering and attention mechanism proposed can be used to identify fraudulent spam comments. The model discovers semantic groups by clustering and extends the multi-scale semantic unit matrix using the attention model to better learn the semantics of the text. And we combine a variety of features to help the learning algorithm achieve better performance, improve the accuracy and F1 value. We built experiments on the latest public data sets. The results show that neural networks based on clustering and attention mechanisms are more effective than other neural network models in deceptive comment detection.

In the future, we can consider a generic model that can detect multiple domains.

Acknowledgements

This work was supported by National High-tech R&D Program of China (863 Program) (Grant No. 2012AA062202).

References

[1] R. Ireland, A. Liu, Application of data analytics for product design: sentiment analysis of online product reviews, CIRP Journal of Manufacturing Science and Technology 23(2018) 128-144.

[2] N. Jindal, B. Liu, Opinion spam and analysis, in: Proc. 2008 International Conference on Web Search and Data Mining, 2008.

[3] M. Ott, C. Cardie, J. Hancock, Estimating the prevalence of deception in online review communities, in: Proc. 2012 International Conference on World Wide Web, 2012.

[4] A. Mukherjee, Detecting deceptive opinion spam using linguistics, behavioral and statistical modeling, Tutorials 26-31(2015) 21-22.

[5] M. Ott, Y. Choi, C. Cardie, J.T. Hancock, Finding deceptive opinion spam by any stretch of the imagination, in: Proc. 2011 Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011.

[6] L. Li, W. Ren, B. Qin, T. Liu, Learning document representation for deceptive opinion spam detection, in: Proc. 2015 Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, 2015.

[7] N. Jindal, B. Liu, E.-P. Lim, Finding unusual review patterns using unexpected rules, in: Proc. 2010 International Conference on Information and Knowledge Management, 2010.

[8] J. Li, M. Ott, C. Cardie, E. Hovy, Towards a general rule for identifying deceptive opinion spam, in: Proc. 2014 Annual Meeting of the Association for Computational Linguistics, 2014.

[9] S. Shojaee, M.-A.-A. Murad, A.-B. Azman, N.-M. Sharef, S. Nadali, Detecting deceptive reviews using lexical and syntactic features, in: Proc. 2013 International Conference on Intellient Systems Design and Applications, 2013.

[10] F.-H. Li, M. Huang, Y. Yang, X. Zhu, Learning to identify review spam, in: Proc. 2011 International Joint Conference on Artificial Intelligence, 2011.

[11] A.M.E. Halees, A.A. Hammad, An Approach for Detecting Spam in Arabic Opinion Reviews, International Arab Journal of Information Technology 12(2015) 9-16.

[12] A. Mukherjee, V. Venkataraman, B. Liu, N. Glance, What yelp fake review filter might be doing, in: Proc. 2013 International AAAI Conference on Weblogs and Social Media, 2013.

[13] A. Prieto, B. Prieto, E.-M. Ortigosa, E. Ros, F. Pelayo, J. Ortega, I. Rojas, Neural networks: an overview of early research current frameworks and new challenges, Neurocomputing 214(2016) 242-268.

[14] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa, Natural language processing (almost) from scratch, Journal of Machine Learning Research 12(2011) 2493-2537.

[15] E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences, in: Proc. 2014 Annual Meeting of the Association for Computational Linguistics, 2014.

[16] N.-X. Bach, N.-D. Linh, T.-M. Phuong, An empirical study on POS tagging for Vietnamese social media text, Computer Speech & Language 50(2018) 1-15.

[17] J. Turian, L. Ratinov, Y. Bengio, Word representations: a simple and general method for semi-supervised learning, in: Proc. 2010 Annual Meeting of the Association for Computational Linguistics, 2010.

[18] R. Socher, J. Bauer, C.D. Manning, A.Y. Ng, Parsing with compositional vector grammars, in: Proc. 2013 Annual Meeting of the Association for Computational Linguistics, 2013.

[19] A. Mnih, G.E. Hinton, A scalable hierarchical distributed language model, in: Proc. 2009 Advances in Neural Information Processing Systems, 2009.

[20] Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin, A neural probabilistic language model, Journal of Machine Learning Research 3(2003) 1137-1155.

[21] R. Socher, A. Perelygin, J.Y. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proc. 2013 Conference on Empirical Methods in Natural Language Processing, 2013.

[22] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, B. Qin, Learning sentiment-specific word embedding for twitter sentiment classification, in: Proc. 2014 Annual Meeting of the Association for Computational Linguistics, 2014.

[23] P. Wang, B. Xu, J. Xu, G. Tian, C.-L. Liu, H. Hao, Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification, Neurocomputing 174(2016) 806-814.

[24] P. Wang, J. Xu, B. Xu, C.-L. Liu, H. Zhang, F. Wang, H. Hao, Semantic clustering and convolutional neural network for short text categorization, in: Proc. 2015 Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing, 2015.

[25] T. Mikolov, I. Sutskever, K. Chen, G.-S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proc. 2013 Advances in Neural Information Processing Systems, 2013.

[26] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, Science 344(2014) 1492-1496.

[27] G.E. Hinton, N. Srivastava, A. Krizhevsky, Improving neural networks by preventing coadaptation of feature detectors, Computer Science 3-4(2012) 212-223.

[28] J. Pennington, R. Socher, C. Manning, Glove: global vectors for word representation, in: Proc. 2014 Conference on Empirical Methods in Natural Language Processing, 2014.

[29] Y. Zhang, B. Wallace, A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification, in: Proc. 2017 International Joint Conference on Natural Language Processing, 2017.

[30] J. Ma, W. Gao, P. Mitra, S. Kwon, B.J. Jansen, K.-F. Wong, M. Cha, Detecting rumors from microblogs with recurrent neural networks, in: Proc. 2016 International Joint Conference on Artificial Intelligence, 2016.

[31] Y.-W. Teh, M.I. Jordan, M.J. Beal, D.M. Blei, Sharing clusters among related groups: hierarchical dirichlet processes, in: Proc. 2005 Advances in Neural Information Processing Systems, 2005.

[32] G. Rao, W. Huang, Z. Feng, Q. Cong, LSTM with sentence representations for document-level sentiment classification, Neurocomputing 308(2018) 49-57.

[33] D.H. Johnson, The insignificance of statistical significance testing, The Journal of Wildlife Management 63(1999) 763-772.