The Classification of Chinese Sensitive Information Based on BERT-CNN

Yujie Wang[2], Xianjun Shen[1(✉)], and Yujuan Yang[1]

[1] School of Computer, Central China Normal University, Wuhan, Hubei, China
xjshen@mail.ccnu.edu.cn
[2] Collaborative & Innovation Center, Central China Normal University, Wuhan, Hubei, China

Abstract. The traditional classification method of Chinese sensitive information mainly relies on the frequency of the co-occurrence of sensitive words and keywords. However, it is difficult to detect the meaning and context relationship of some complex statements. In this paper, a new model is proposed to classify Chinese network sensitive information. The model, which is based on CNN (Convolutional Neural Network) and latest pre-trained BERT (Bidirectional Encoder Representation from Transformers), is called the BERT-CNN deep learning model. Firstly, nearly 20,000 comments of corresponding videos are collected from YouTube, after data cleaning and labeling, the texts are divided into four categories to build the SLD (sensitive language detection) dataset. Finally, the word vectors from the pre-trained BERT are used as the input of the CNN model, constructing the Chinese sensitive information classifying BERT-CNN model. The result reveals that, compared with the traditional neural network model, the BERT-CNN model proposed in this paper improves the generalization ability of word embedding, and can effectively achieve the recognition and classification of the network sensitive information in short text dataset. The experimental results perform better than the classic CNN and RNN (Recurrent Neural Network) models.

Keywords: Sensitive information · Short text classification · BERT · CNN

1 Introduction

In recent years, the international situation has been complex and volatile, such as trading war between China and the United States, demonstrations and disturbances around the world. There is a large amount of data in various social platforms, which can be used for public opinion analysis. In the current international environment, the classification of Chinese sensitive information is of great significance.

Text classification is a key link in natural language processing (NLP). It paved the way for the following tasks, such as search engine, network information filtering and public opinion analysis [1]. With the development of network society, more and more sensitive information is flooded in various social platforms and video websites. The main representatives of social platforms are twitter and microblog, while the main representatives of video websites are YouTube and YOUKU.

K. Li et al. (Eds.): ISICA 2019, CCIS 1205, pp. 269–280, 2020.
https://doi.org/10.1007/978-981-15-5577-0_20

Because YouTube does not need any threshold to upload video and has no high technical requirements, it is convenient for people of all ages to upload video and pictures. Therefore, YouTube is the fastest growing and most used video website in the world. The total number of video uploads reached 114 billion minutes, with a billion users and billions of comments. Compared with social media, YouTube tends to spread non-textual content, which will contain more information. The comment under video has more information for us to analyze.

At present, research on sensitive information mainly focuses on the research and analysis of hate and aggression [2], cyberbullying [3]. Since 2019, the conflicts on the internet have become more and more intense. A variety of statements containing sensitive information are increasing in the network platforms. The research on the network sensitive information has become more and more detailed, such as the research on network sensitive information based on OLID data set, it contains three subtasks, one of which, task A, is to classify offensive and non-offensive texts [4]. Since the definition of the task is a little vague and repetitive in related subtasks, it is also necessary to distinguish hate speech from general offensive speech [5]. Deep learning algorithm is based on large-scale manually labeled data, however, it lacks uniform standards, and everyone has different definitions of sensitive information, leading to the problem of semantic ambiguity. Some studies only consider the sub-task of hatred, not abusive bullying or other directions [6].

The method based on constructing a sensitive word dictionary can deal with unstructured data. Therefore, it is widely used for detecting sensitive information. However, with the rapid development of the Internet, the expansion speed of sensitive words is faster than the filling speed of dictionaries. Therefore, the method based on word frequency statistics can hardly be perfectly applied. At the same time, if we want to determine whether a sentence is sensitive or not, it may have nothing to do with the presence of sensitive words and keywords. We need to understand the meaning of the sentence from the context.

With the rapid development of the Internet, the expansion speed of sensitive words is faster than the filling speed of dictionaries. Therefore, the method based on word frequency statistics can hardly be perfectly. At the same time, if we want to determine whether a sentence is sensitive or not, it may have nothing to do with the presence of sensitive words and keywords. We need to understand the meaning and context relationship of the sentence.

In this paper, the model we proposed fuses the convolutional neural network and BERT, which can learn semantic and grammatical information of the context. BERT-CNN can understand meaning and context relationship of some complex statements and effectively improve the accuracy of identifying and classifying sensitive information on the Internet. The comments under the video were crawled through Google YouTube API. After data cleaning and labeling, we build SLD dataset. The experimental results show that, compared with the traditional neural network model, the BERT-CNN deep learning model can effectively classify dataset, which consist of sensitive information on the Internet, and is significantly superior to the classical CNN and RNN models.

2 Related Work

In this part, we will give a description of the earlier work of the short texts classification and public opinions analysis based on social network platform in the field. Normally, a network sensitive information detection includes specifying the problem, labeling the harmful information as aggression, cyberbullying or hate speech, and then classifying them using different algorithm. As for the traditional classification of short texts, the main task contains three themes: characteristics engineering, characteristics selection and the machine learning algorithm selection based on the former two themes. In 2012, junmingxu [7] researched about the bully action existing in the network social media, they confirmed several key problems and arranged the problems into NLP tasks: context classification, character labeling and emotion analysis using topic model to identify related topics [8]. Homophobia and socialism may be easily labeled as hate speech [9], yet some kinds of gender aggression can be difficultly classified, because it is hard to define a large dictionary to accurately classify them. By the method of traditional machine learning, they classified nearly 24,000 tweet data into three categories. The Germ Eval released 8,500 labeled German twitters which were used for coarse-grained binary classification - offensive and unoffensive. Meanwhile, they carried about another task to subdividing the subset of the offensive set. The task included multiple hierarchical tasks where the binary classification was firstly finished and then the fine-grained classification was launched in the certain set of the offensive set from the former classification [10]. The idea made a pleasant effort in the realistic application. All the papers mentioned above used only some traditional machine learning algorithms, but just as was mentioned in the last parts of the papers, the problem is still far from settled although the algorithms, which have problems such as sparse matrix, are effective in some ways.

In recent years, the occurrence of DNN (deep neural network) settled the sparse matrix problem. Additionally, DNN model performs better than the traditional words-frequency statistical method, whose limitation leads to a difficulty learning the relationship of the context, in learning the grammar and meanings of the words, understanding the context profoundly and classifying contexts. Institute of Automation, Chinese Academy of Sciences [10] proposed a Recurrent Convolution Neural Network applying in context classification in 2015. The new neural network model, using cycle structure to study as more context information as possible and CNN to construct context representatives, can have less noises than the CNN. Some work compared the word2vec with the traditional n-gram model, and construct a twitter hate speech context classifying system based on deep learning. Four CNN models were trained based on word2vec at the same dataset to construct words embedding of meaning information, generate words embedding including ones combined with n-g characters randomly. Maxpooling was used to narrow the characteristics set in the net, and Tweet was classified by softmax function. After ten cross validations, the model based on word2vec embedding proved to be the best. The latest works were mainly about improving classification performance with perceptual network, Mikolov [11] employed a new continuous Skip-gram for effective embedding performance. Kalchbrenner and Blunsom [12] proposed a new dialect behavior classification cycle network. Collobert [13] introduced a CNN which is used for meaning character markings.

3 Method

In this paper, the model that we proposed is BERT-CNN. The model structure is shown in Fig. 1. The BERT-CNN model is applied to the dataset of Chinese sensitive information. In this model, the word and sentence embedding are learned by utilizing the pre-trained BERT model. After that, the embedding vectors are extracted from BERT as the input of the CNN. In other words, we treat the embedding extracted from BERT as the conventional embedding layer.

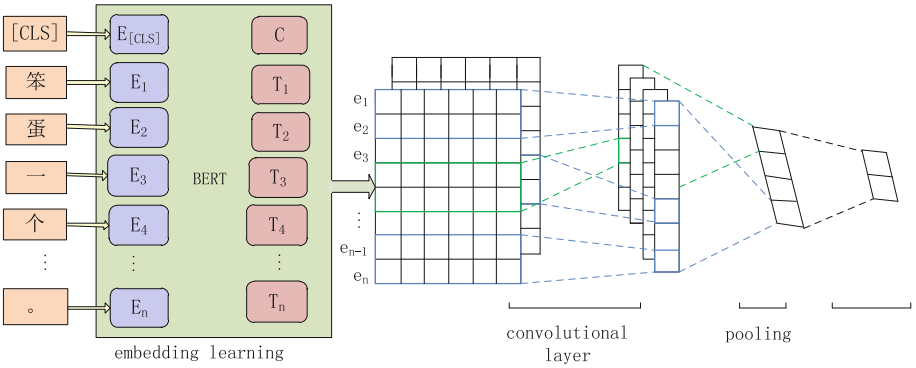

Fig. 1. The architecture of BERT-CNN

3.1 BERT

Compared with the ELMO [14] model proposed by Google in 2018, BERT [15] has changed language model from BiLSTM to Transformer [16], and realized the concept of bidirectional encoding in a real sense. model structure of BERT is shown in Fig. 2.

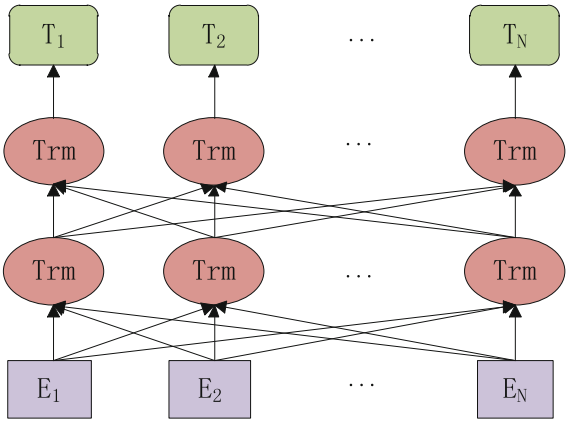

Fig. 2. The architecture of BERT

In the past, the language model was the probability distribution of a long statement, specifically, it is the probability value of the text of length of m. In order to facilitate calculation, the N-Gram model was proposed. The formula of N-Gram model is as follows:

$$p(S) = p(w_1 w_2 \cdots w_n) = p(w_1)p(w_2|w_1) \cdots p(w_n|w_{n-1}w_{n-2} \cdot \cdot w_2 w_1) \quad (1)$$

N-Gram is used to calculate the probability that the current word appears for the first n words. At that time, the N-Gram model would have the problem of sparse of matrix as n increased. In order to solve the problem of sparse of matrix, the solution given by the neural network is to optimize the model parameters based on the BP algorithm, through the random initialization of the network model parameters. The latest pre-training idea is that the value of parameter is no longer random initialization, the task was trained to obtain a fixed parameter value, which was used to initialize the model, and then the model was trained.

BERT model further enhanced the generalization ability of word embedding, and more fully learned the feature of character level, word level and sentence level. In the process of language training, they abandoned to predict the model from left to right, rather than using random masking. The method of masking randomly masks off 15% token in the corpus, and then hidden layer which contain masked word treat as a input of the softmax to predict.

Some models can only learn the features of token level but also need the features of sentence level. BERT borrowed the idea of negative sampling from word2vec [17] and used it for sentence level sampling to make dichotomy of sentence level to judge whether a sentence is noise. BERT can directly obtain the unique embedding of an entire sentence. It adds a special mark (CLS) in every input, and then let the Transformer to deep encode. Because the Transformer can ignore the space and distance of the global information, and the highest hidden layer as the presentation of sentences or words, it can directly connect with softmax, through the BP (back propagation), BERT can learn the whole feature of the upper input.

Compared with GPT [18], BERT realizes the concept of bidirectional encoding in a real sense. Because of self-attention, the transformer used by BERT realizes full connection of the model. The transformer can be regarded as a multilevel Encoder-Decoder model. Each layer of Encoder has two operations, which are Self-Attention and Feed Forward. In the other hand, each layer of Decoder has three operations, which are Self-Attention, Encoder-Decoder Attention and Feed Forward operation. Self-Attention and Encoder-Decoder Attention both use Multiheaded Attention mechanism. The dot product of Attention is scale. The input includes Q and K with dimension d_k, and V with dimension d_v. Take the dot product of Q with all K and divide by $\sqrt{d_k}$. Then use a softmax function to get the weight of V.

$$Attention(Q, K, V) = softmax\left(\frac{QK^t}{\sqrt{d_k}}\right)V \quad (2)$$

3.2 TextCNN

TextCNN [19] is a convolutional neural network proposed in 2014 for text classification. Convolutional neural network is a common neural network model. The general structure of convolutional neural network includes convolutional layer, pooling layer and full connection layer. Compared with traditional neural network, it contains convolution layer and pooling layer. Neurons between the convolutional layers are no longer connected in full connection, and neurons in the next layer only receive part of the input results from neuron nodes in the previous layer. The incoming input of the next layer of neurons is determined by the output of the previous layer of neurons, while the output of the previous layer of neurons is determined by the convolution kernel. Compared with full connection, the convolution layer realizes parameter sharing greatly reduces the number of parameters. Pooling layer is the pooling of the results of the convolution layer, also known as sub-sampling. Once again, the pooling layer carries out feature screening for the convolution layer, and removes useless features, and further optimizes the parameters of the convolutional neural network.

In the convolutional process of TextCNN, we embed the result of last layer of BERT as embedding layer. If the dimension of the word vector is d and the number of words is s, the matrix $A \in R^{s \times d}$ can be obtained. Then we use the convolutional neural network to extract the features. Since the correlation degree of the words in the short text is very high, the 1-dimensional convolution kernel can be used. Assuming that there is convolutional kernel $W \in R^{h \times d}$, we have $h * d$ parameters that need to be updated, and the result obtained by the last layer of BERT goes through embedded layer to obtain matrix A. The formula of convolution layer can be expressed as:

$$o_i = w \cdot A[i : i + h - 1], i = 1, 2, \ldots, s - h + 1 \tag{3}$$

Add offset b, the characteristic is obtained under the activation of activation function f, and the specific formula is as follows:

$$c_i = f(o_i + \mathrm{b}) \tag{4}$$

In the process of pooling, because there are multiple convolution kernels of different sizes, the number of features obtained is generally different. Therefore, pooling layer is added to select the maximum feature value of each convolution kernel, and then cascade to obtain the final feature result. Then, the feature result inputs into softmax layer to complete classification of short text.

4 Experiment

4.1 Dataset

With comments greater than 1000 and view counts greater than 10000 as thresholds, we chose 20 hot Chinese videos from YouTube. In addition, the comments below video were crawled through Google YouTube V3 API, and the total number of comments was calculated as 20000. We stored the 20000 comments, then we cleaned

the data, and converted the traditional Chinese character in the comments into simplified Chinese character, and filtered out the useless data with messy codes and HTML tags. Finally, we got 18,707 comments. All the 18707 comments were manually labeled by two people. Before manually labeling the data, we formulated the labeling standard according to the actual situation. The labels used 4 highly different words (un-sensitive, abuse, political sensitive, heresy). Finally, there are 8603 pieces of non-sensitive data, 8228 pieces of abuse data, 1362 pieces of political sensitive data and 514 pieces of heresy data. In our paper, we collated all these comments into a new SLD (Sensitive Language Detection) dataset. Finally, SLD dataset was divided into train set, test set and validation set according to 6:2:2. The dataset partition is shown in Table 1.

Table 1. Dataset distribution

Data	Train set	Test set	Val set	Total
Un-sensitive	5357	1700	1546	8603
Abuse	5075	1660	1493	8228
Sensitive	841	303	218	1362
Heresy	327	96	91	514
Total	11600	3759	3348	18707

4.2 Experiment Setting

In process of training CNN and RNN and BERT-CNN, we use the cross entropy as loss function. CNN uses Adam as the optimizer, and its parameters are set as the learning rate: 0.001. At the same time, we added dropout technology and early-stop technology in the training process. The principle of dropout technology is to randomly abandon a certain proportion of nodes in the training process to prevent the occurrence of over fitting. In the end, dropout is set as 0.5. Early-stop technology evaluates the performance of the model on the validation set after each iteration. When the evaluation result of the validation set is no longer improved in N consecutive rounds, the iterative process is truncated and the training of the model is stopped. When N is set too small, the iteration may not converge. When N is set too large, the time consumed will increase. The hyper parameters setting in this paper is shown in Table 2.

Table 2. Hyper parameter setting

Parameter	Value
Embedding dim	64
Learning rate	1e−3
Train epoch	100
Dropout	0.9
Batch size	64
Epoch	10

4.3 Evaluation Metrics

The formula of evaluation standard is as follows. P value is the precision rate, indicating the proportion of texts which are correctly predicted in the predicted text, and the R value is the Recall rate, indicating the proportion of correctly predicted text in the all positive text. The value F_1 is the harmonic mean of P and R. TP is the number of successfully predicted positive cases, FP is the number of falsely predicted positive cases, and FN is the number of falsely predicted negative cases.

$$P = \frac{TP}{TP + TF} \tag{5}$$

$$R = \frac{TP}{TP + FN} \tag{6}$$

$$F_1 = \frac{2 \cdot P \cdot R}{P + R} \tag{7}$$

5 Results and Discussion

5.1 Performance Evaluation

In order to verify the performance of classification of the BERT-CNN we proposed. We conducted comparative experiments on three models of CNN, RNN and BERT-CNN under the same hyper parameters and same dataset.

The experiment results are shown in Table 3. Table 3 shows the results of P value, Recall value and F value of the three models on the same validation set.

Table 3. The performance of comparison

Approach	Tag	Precision	Recall	F_1
CNN	UN-SEN	95.0	93.0	94.0
	ABU	95.0	97.0	96.0
	SEN	43.0	33.0	37.0
	HER	79.0	92.0	85.0
RNN	UN-SEN	91.0	94.0	93.0
	ABU	96.0	95.0	95.0
	SEN	32.0	10.0	15.0
	HER	80.0	86.0	83.0
BERT- CNN	UN-SEN	94.0	94.0	94.0
	ABU	96.0	97.0	97.0
	SEN	**79.0**	**72.0**	**75.0**
	HER	81.0	90.0	85.0

The results of the classification of different labels are shown in Table 3. In particular, un-sensitive labels and abuse labels have a significantly better results than the other two labels, and F_1 value of these two types of labels is higher than 90%. In the non-sensitive label, the F_1 value of BERT-CNN is 1% higher than the RNN. On the Abuse tag, F_1 value of the BERT-CNN is 1% higher than CNN and 2% higher than RNN. Compared with the traditional neural network model, BERT-CNN has a considerable improvement in sensitive labeling. In the label of heresy, all three models performed similarly. BERT-CNN had the same F_1 value as CNN on the heresy label, which is 2% higher than RNN. The results of abuse label shown in Fig. 3.

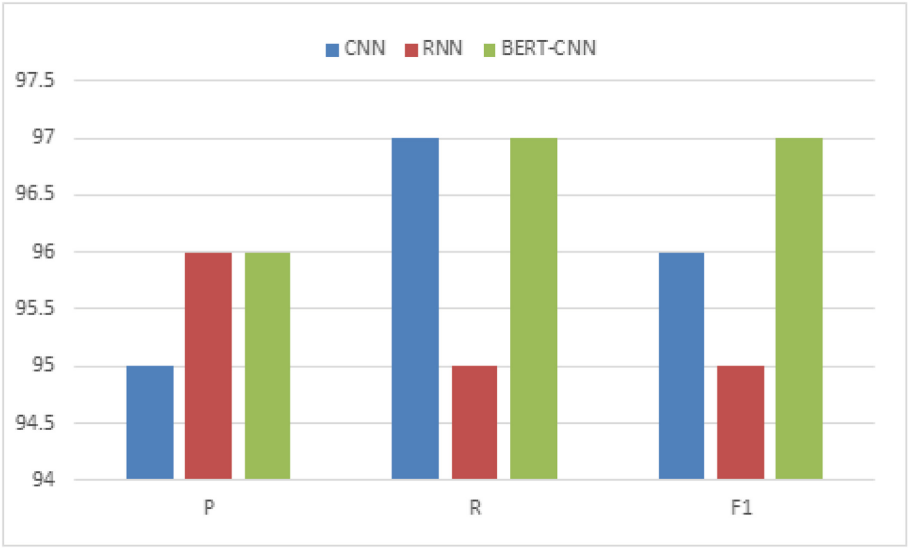

Fig. 1. The performance of BERT-CNN on abuse label

In the classification of sensitive information, it can be clearly seen from Fig. 4 that BERT-CNN is far better than the other two comparison of models in P value, Recall value and F_1 value.

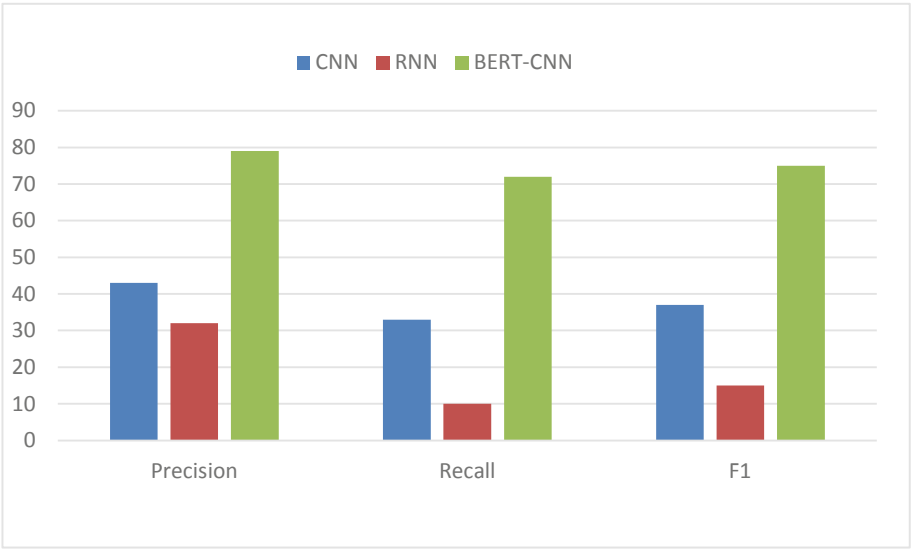

Fig. 4. The performance of BERT-CNN on sensitive label

6 Conclusion

Detecting the specific meaning and context relationship of some complex sentences in the dataset of unstructured short texts based on emotion dictionary shows high difficulty. In This paper, we build a new dataset SLD (Sensitive Language Detection) to achieve Chinese sensitive information classification. We propose a BERT-CNN model based on pre-trained model. BERT improved the generalization ability of word embedding, and added features of characters, words and sentences levels. The embedding learned from BERT was used as the input layer of TextCNN for Chinese sensitive information classification. Dropout technology was applied during the training to prevent over-fitting.

The result reveals that, compared with the traditional neural network model, the BERT-CNN model proposed in this paper improves the generalization ability of word embedding, and can effectively achieve the recognition and classification of the network sensitive information in short text dataset. and is significantly superior to the classical CNN and RNN models.

Although BERT-CNN has a significant improvement compared with CNN and RNN in the tetrad classification, it is still can be improved in the sensitive labeling. Since the SLD data set has a certain imbalance of negative and positive examples, the prediction effect of short text is influenced. In addition, comments under YouTube videos are not independent and have a strong interactive relationship, that is to say, comments do not exist independently and will have a certain logical relationship with other comments. How to classify the text of sensitive information in a more fine-grained way, taking other relevant comments of the same video into consideration as context, will be the future direction of this article.

Acknowledgement. This research is supported by the National Language Commission Key Research Project (ZDI135-61), the National Natural Science Foundation of China (No. 61532008 and 61872157), and the National Science Foundation of China (61572223).

References

1. Aggarwal, C.C., Zhai, C.X.: A Survey of Text Classification Algorithms. In: Aggarwal, C., Zhai, C. (eds.) Mining Text Data, pp. 163–222. Springer, Boston (2012). https://doi.org/10.1007/978-1-4614-3223-4_6
2. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: SemEval-2019 task 6: identifying and categorizing offensive language in social media (OffensEval) (2019)
3. Dinakar, K., Reichart, R., Lieberman, H.: Modeling the detection of textual cyberbullying 2011 (2011)
4. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: Predicting the type and target of offensive posts in social media (2019)
5. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language (2017)
6. Gambäck, B., Sikdar, U.K.: Using convolutional neural networks to classify hate-speech. In: Proceedings of the First Workshop on Abusive Language Online 2017, pp. 85–90 (2017)
7. Xu, J.M., Jun, K.S., Zhu, X., Bellmore, A.: Learning from bullying traces in social media. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies 2012 (2012)
8. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
9. O'Connor, B., Balasubramanyan, R., Routledge, B.R., Smith, N.A.: From tweets to polls: linking text sentiment to public opinion time series 2010 (2010)
10. Wiegand, M., Siegel, M., Ruppenhofer, J.: Overview of the germeval 2018 shared task on the identification of offensive language (2018)
11. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Twenty-Ninth AAAI Conference on Artificial Intelligence 2015 (2015)
12. 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 2013, pp. 3111–3119 (2013)
13. Kalchbrenner, N., Blunsom, P.: Recurrent continuous translation models. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing 2013, pp. 1700–1709 (2013)
14. 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)
15. Devlin, J., Chang, M., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
16. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. arXiv preprint arXiv:1802.05365 (2018)
17. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems 2017, pp. 5998–6008 (2017)
18. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)

19. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageunderstandingpaper.pdf (2018)
20. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)