Proceedings of the 2023 International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/8/20230221

Spam email filtering leveraging improved text convolutional neural network

Dongjie Chen

Smart City College, Beijing Union University, Beijing, 100101, China

2019240383002@buu.edu.cn

Abstract. This paper outlines the development of anti-spam technology in the last two decades and introduces a novel approach using a convolution neural network (CNN) to tackle the problem of spam filtering. The study uses the TREC06c dataset, containing Chinese spam email data, the dataset is split between a training set and a test set. The paper also introduces the concept of word embeddings, which converts each word in the text into a real-valued vector, better reflecting the semantic relationships between words. The TEXT-CNN algorithm is then discussed, which applies convolutional neural networks to text data and is generated by modifying the TEXT-CNN model to improve its performance in the spam filter. The conclusion of this article is that TEXT-CNN model demonstrates great results in the task of identifying spam emails, and the classification efficiency can be further improved by introducing an attention mechanism and batch processing mechanism by improving the model. The article also provides some ideas for further improvement.

Keywords: deep learning, convolutional neural network, spam email.

1. Introduction

From the birth of the first spam email in 1978 to the official recognition of the concept of spam emails due to the Green Card incident in 1994, to the emergence of the first program capable of mass automatic sending of spam emails in 1995, spam emails have flooded people's lives with their low cost [1,2]. While email provides people with convenience, economic, and fast services, it also gives some businesses, companies even criminals the opportunity to use email for advertising and illegal activities, which causes a user not only to spend a lot of time dealing with spam but also may suffer serious consequences such as computer viruses and fraud by clicking on links in emails, resulting in serious losses [3,4].

From the first mention of anti-spam technology in 1996, to today already over 20 years later, with the continuous development and progress of technology, anti-spam technology has also been constantly updated and developed. The history of anti-spam technology can be divided to 4 periods [5,6]. The first generation of technology mainly included blacklisting and whitelisting techniques and keyword search, but they only got a limited effect. In the second generation of technology, real-time blacklisting technology improved upon the first generation of blacklisting technology, while electronic signature technology limited senders. After that, spam filtering technology did not make further breakthroughs to against the large number of spam emails until the advent of technologies such as intelligence created by machines and the act of computers learning., which gave rise to the third generation of technology, which was finally able to effectively filter large-scale spam emails. Today, the fourth generation of technology combines multiple technologies, including layered filtering technology and other filtering technologies, to leverage each other's advantages, providing a comprehensive solution that includes the most effective anti-spam technology.

There are many classification methods are used to use on spam filter. It could be found that today most Spam Filtering research still using the old machine learning technology. Like bayes, SVM, KNN and so on [7]. All of them have their advantages and disadvantages, but all of them are too old. So, convolution neural network (CNN) has been chosen as model in this research. It is a deep learning method with a good performance [8].

2. Method

2.1. Dataset

This study uses the TREC06c dataset, which contains cleaned Chinese spam email data [9], the data is split into a training set and a test set in a ratio of 3 to 1. The TREC 2006 Spam Corpus is a dataset used for training and testing spam filters. It consists of approximately 65,000 emails, including both spam and non-spam emails. The emails in the dataset were collected by the organizers of the TREC 2006 Spam Track from various sources, such as email servers, newsgroups, and mailing lists, and were manually labeled and classified.

The emails in the TREC 2006 Spam Corpus are stored in their original text format, with each email labeled as either spam or non-spam and accompanied by additional metadata information such as subject, sender, and recipient. This dataset has become a widely recognized standard dataset and is commonly used for research and evaluation of spam filters.

2.2. Word embeddings

The introduction of word embeddings is primarily aimed at addressing the issue in traditional text classification and natural language processing where it is difficult to handle the semantic relationships between words [10]. Traditional text processing methods often treat each word as a discrete symbol without considering the associations and semantic information between words. Word embeddings, on the other hand, can convert each word in the text into a real-valued vector, which can better reflect the semantic relationships between words such as word similarity and semantic distance. Word embeddings not only improve the effectiveness of natural language processing tasks but are also an important component in deep learning models.

2.3. Text-CNN

The CNN mimics the biological visual perception mechanism to construct, this type of model can be utilized for both supervised and unsupervised learning approaches. Most CNN algorithms are applied to image processing, but TEXT-CNN (Convolutional Neural Network for Text Classification) is an algorithm focus on NLP field, it is generated in 2014 by Kim Yoon, he modified CNN model to make it can get a better result on NLP field.

The core idea is to apply convolutional neural networks to text data, use convolutional layers to capture local features in the text, then compress these features into fixed-dimensional vectors through pooling layers, and finally input these vectors into fully connected layers for classification. Compared with machine learning text classification algorithms based on traditional vectorization model like TF-IDF or word-bag, TEXT-CNN can automatically learn local features in the text, thus achieving more accurate text classification.

Text-CNN is a common text classification algorithm, and its advantages and disadvantages are as follows:

For the advantages, firstly, it can capture word order information. The Text-CNN method employs convolutional neural networks to capture text features which can capture word order information and thus is more accurate than traditional text classification algorithms based on bag-of-words. Secondly, it has a certain degree of translational invariance. Text-CNN uses convolutional operations to capture local features in text, which has a certain degree of translational invariance and can reduce the impact of noise in the dataset on the model to some extent. Thirdly, it is simple and easy to use. Text-CNN has a relatively simple model structure, which is easy to implement and adjust, so it is widely used in text classification tasks.

As for the disadvantages, firstly, it is unable to handle variable-length text sequences: Text-CNN takes a fixed-length text sequence as input, so it is unable to handle variable-length text sequences and requires truncation or padding operations. Secondly, the size of the convolutional kernel needs to be manually adjusted. Text-CNN leverages convolutional kernels of varying sizes to extract features from text data; however, the adjustment of kernel sizes needs to be done manually. If the size of the convolutional kernel is not properly selected, it may affect the performance of the model. Thirdly, it is insensitivity to word order information. Text-CNN cannot capture word order information, so in some text classification tasks, recurrent neural network models are needed to process the temporal information of text.

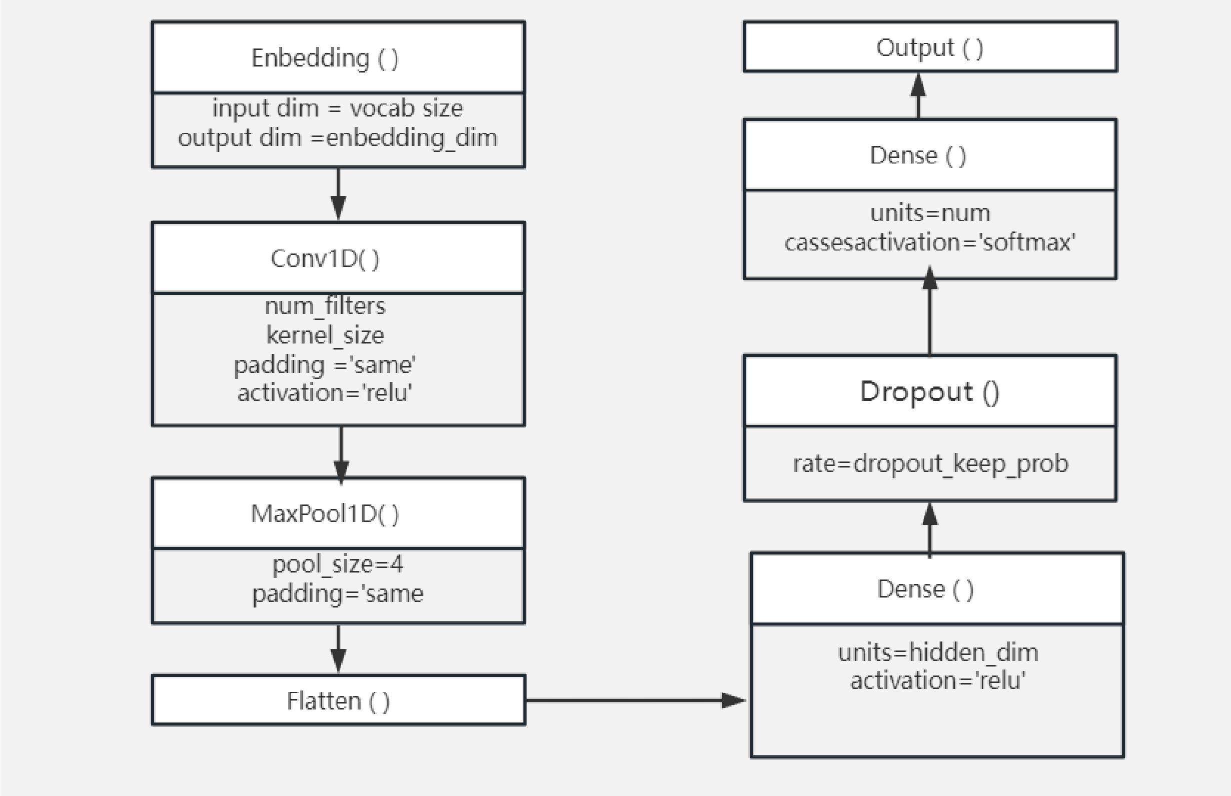

Figure 1. Structure of TEXT-CNN.

2.4. Improvements

Improvements were made to the existing structure by adding batch layers, which help the model better adapt to different data distributions, improve training stability and convergence speed, and thus enhance model performance. Batch normalization is one of the most used optimization techniques in deep learning and can speed up the training process. Given that the dataset in this project is large and attention mechanisms have been added in subsequent improvements, training time may increase significantly. Therefore, to expedite the training process and enhance the model's generalization performance, batch normalization layers were employed. Four batch normalization layers were added in total, each added after the non-linear layers of the conv, attention, flatten, and dense layers.

In addition, an attention layer was added before the max pool layer to introduce an attention mechanism to the model. Its role is to automatically learn the importance of different parts based on the input information, and then calculate the output of these parts weight. This approach allows the model to focus on key information when understanding the text, rather than simply summarizing the information of the entire sentence with equal weight.

L2 regularization is used in the first dense layer, L2 regularization is a technique used to alleviate overfitting by penalizing large weights during training of a neural network, The purpose of incorporating this technique is to avoid the model from overfitting to the training data. As the task of spam email classification is relatively simple and the model has strong learning ability, it is necessary to minimize the occurrence of overfitting.

In the end, Adaptive Moment Estimation (ADAM) is chosen as optimizer into model, ADAM is an optimization algorithm with adaptive learning rate, it combines the advantages of gradient descent and momentum method. It has a fast convergence speed and good performance.

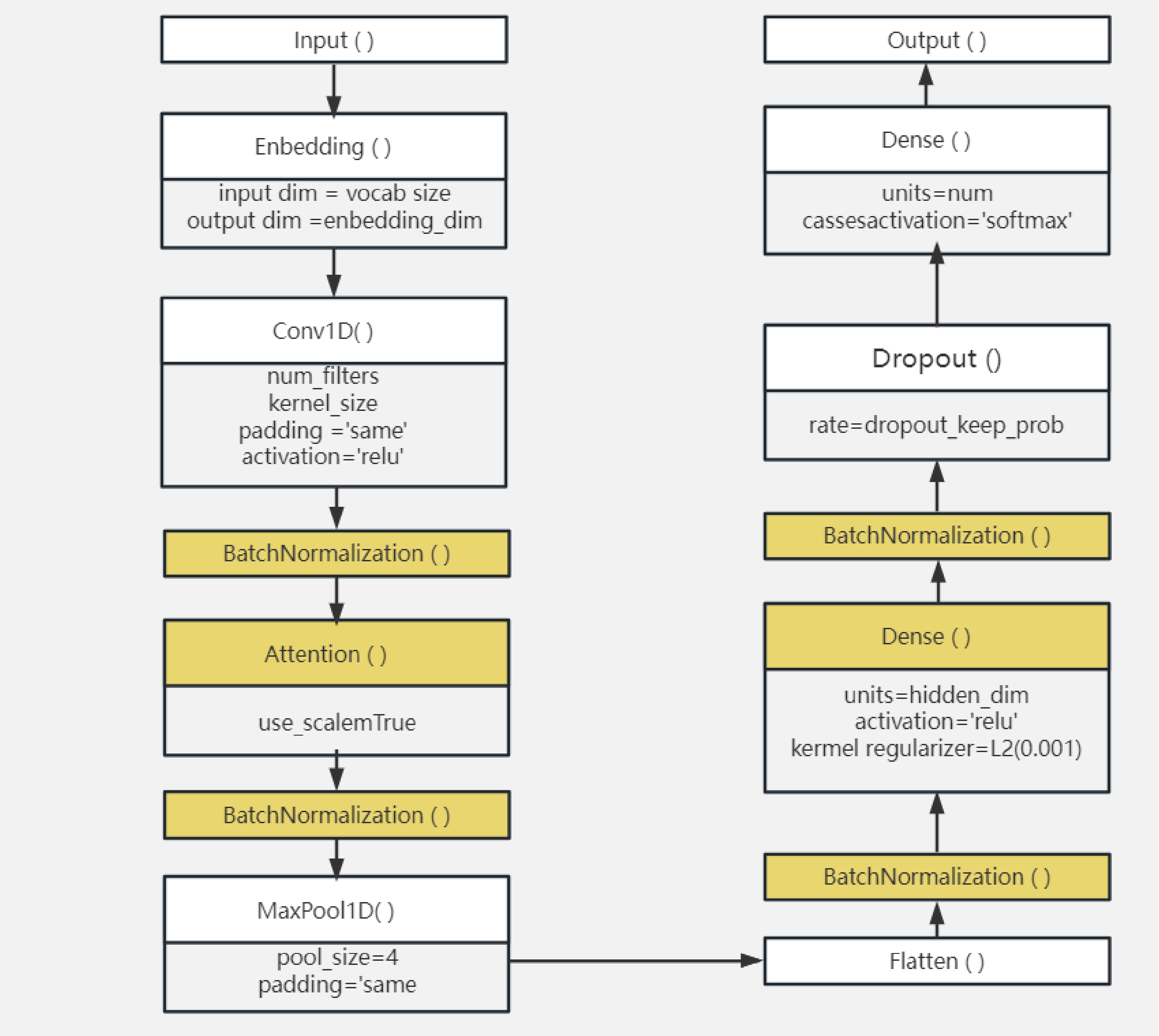

Figure 2. Structure of improved model.

2.5. Evaluation metrics

The current evaluation metrics for this study are as follows:

Accuracy: Accuracy refers to the ratio of correctly classified samples to the overall number of samples.

Precision: In binary classification problems, Precision is the ratio of true positive samples to the total number of samples predicted as positive by the model.

Recall: In binary classification problems, recall refers to the ratio of true positive samples to the total number of actual positive samples.

F1-score: The F1-score is a metric that calculates the harmonic mean of precision and recall, and it can provide an assessment of the model's overall performance.

3. Result

Table 1 and Table 2 present the confusion matrices of two different models, both of which were evaluated using the same test dataset consisting of 16155 emails.

Table 1. Confusion matrix of Text-CNN model.

	Predict label: spam	Predict label: normal
Actual label: spam	10430	306
Actual label: normal	187	5232

Table 2. Confusion matrix of improved Text-CNN model.

	Predict label: spam	Predict label: normal
Actual label: spam	10671	67
Actual label: normal	282	5135

There is evaluation report of two models shown in Table 3. It prints each evaluation metrics of two model.

Table 3. Performances of both improved and original Text-CNN model.

Model	accuracy	precision	recall	F1-score	support
TEXT-CNN	0.969	0.970	0.971	0.970	16155
Improved	0.978	0.979	0.994	0.978	16155

It is clear that all the evaluation metrics become higher about improved model. Especially in the recall of spam it even gets 0.994 from 0.971.

4. Discussion

According to the result of the experiment, it can be found that although original TEXT-CNN already can get a fantastic result, improved model still gets a better performance. The attention mechanism can help the model more accurately identify the features related to spam emails. So, the model's recall rate of 0.994 means that it can almost identify every spam email. And under the effects of L2 regularization and batch normalization, the overfitting problem has been alleviated, and the model's generalization ability and robustness have been improved, which has also resulted in better performance during testing.

Despite the satisfactory experimental results, there are also some shortcomings can be improved.

The first aspect that needs improvement is the training time. Although the addition of the attention mechanism in the model improves its ability to capture the features of spam emails and leads to better results, it also significantly increases the model's training time. Taking the example of a training set of approximately 45,000 emails and iterating for 3,000 times, with GPU computation on a RTX4090 graphics card, the original model only takes about 35 seconds, However, the improved model takes about 70 seconds to complete the training, which is twice the training time of the original model.

The second aspect that needs improvement is the dataset. Although the TREC06C dataset used in this experiment contains a large amount of data, it is from many years ago and may differ significantly from the latest spam emails. Moreover, this experiment only used one dataset, which could cause the model to be strongly influenced by the characteristics of the dataset.

Finally, another improvement can be made by using pre-trained word embedding models such as word2vec, glove, and BERT, which can further enhance the classification efficiency. Pre-trained word embeddings can be obtained by learning from large-scale language corpora, which contain rich linguistic knowledge and can better express the meaning and semantics of words. Therefore, using pre-trained word embeddings when training natural language processing models can lead to better model performance and accuracy.

5. Conclusion

The spam email classification algorithm presented in this paper achieved satisfactory experimental results by utilizing a TEXT-CNN-based model with attention mechanism, L2 regularization, and batch normalization techniques. The TREC06 dataset was selected as the experimental dataset, which consists of approximately 50,000 emails for testing and 16,000 emails for training. The experiment achieved satisfactory results. By improving the model, all evaluation metrics were improved, especially, the recall metric has been significantly improved. The addition of attention mechanism improved the model's ability to capture spam email features, but it also increased the training time. There is considerable potential for further enhancements in this research, such as using more datasets to reduce the impact of dataset characteristics on the model or using larger and more diverse datasets as well as pre-trained word embeddings to further improve model performance.

In summary, the research has confirmed that employing deep learning techniques for spam email classification is a viable approach. The research showcased the promise of deep learning in classifying spam emails and offered valuable directions for future investigations in this field. In the future, researchers can further explore new algorithms to improve the classification performance and combine them with practical application scenarios to achieve a more intelligent and efficient spam email classification system.

References

[1] Karim, A., Azam, S., Shanmugam, B., Kannoorpatti, K., & Alazab, M. (2019). A comprehensive survey for intelligent spam email detection. IEEE Access, 7, 168261-168295.

[2] Cormack, G. V. (2008). Email spam filtering: A systematic review. Foundations and Trends in Information Retrieval, 1(4), 335-455.

[3] Dada, E. G., Bassi, J. S., Chiroma, H., Adetunmbi, A. O., & Ajibuwa, O. E. (2019). Machine learning for email spam filtering: review, approaches and open research problems. Heliyon, 5(6), e01802.

[4] Sharma, A. K., & Sahni, S. (2011). A comparative study of classification algorithms for spam email data analysis. International Journal on Computer Science and Engineering, 3(5), 1890-1895.

[5] Ferrara, E. (2019). The history of digital spam. Communications of the ACM, 62(8), 82-91.

[6] Hedley, S. (2006). A brief history of spam. Information & Communications Technology Law, 15(3), 223-238.

[7] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260.

[8] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., et al. (2018). Recent advances in convolutional neural networks. Pattern recognition, 77, 354-377.

[9] Tang, X., Wan, Y., Liu, Y., & Cai, J. (2017, October). Chinese spam classification based on weighted distributed characteristic. In 2017 Chinese Automation Congress (CAC), 6618-6622.

[10] Lai, S., Liu, K., He, S., & Zhao, J. (2016). How to generate a good word embedding. IEEE Intelligent Systems, 31(6), 5-14.