Spam Detection Based on BERT Model: A Descriptive Study

Hailin Xu

School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan Hubei 430074, P.R.China

ABSTRACT

This thesis proposes a spam detection method based on BERT (Bidirectional Encoder Representations from Transformers) model. Spam problem is becoming more and more serious in today's Internet environment, and effective identification and filtering of spam is crucial to protect users' information security and improve the quality of mail services.This study describes in detail the implementation steps of the BERT model-based spam detection method. First, we use a pre-trained BERT model as a feature extractor to convert email text data into a BERT representation. Then, fine-tuning (fine-tuning) is performed using the labeled spam dataset to learn decision bounds for mapping the BERT representation to spam classes by training a classifier. Finally, we evaluate the performance of the method using a test dataset and compare it with other commonly used spam detection methods.Experimental results show that the spam detection method based on the BERT model achieves significant improvements in metrics such as accuracy and recall. Compared with traditional rule-based or feature engineering methods, this method can better capture the complex semantic and contextual information in spam, thus improving the accuracy of spam detection.The spam detection method based on BERT model proposed in this study has good performance and application prospects. Future research can further explore how to combine other technical means and optimization strategies to further improve the effectiveness of spam detection and promote the wide application and diffusion of the method in practical applications.

KEYWORDS

BERT model; Spam detection; Feature extraction; Fine-tuning; Performance evaluation

DOI: 10.47297/taposatWSP2633-456912.20230402

1 Introduction

With the rapid development of the Internet and the popularity of electronic communication, spam has become a serious problem. Spam not only occupies users' valuable time and network bandwidth, but also may contain fraudulent messages and malicious links, posing threats to users' personal privacy and network security. Therefore, automated spam detection becomes an important task to protect users from spam. Effective identification and filtering of spam is crucial to protect users' information security and improve the quality of mail services, and spam detection is now commonly treated as a text classification problem. The basic idea of text classification is to transcribe text data into a machine-readable form, and then use algorithms to classify the text. Traditional text

classification algorithms commonly convert text into a high-dimensional vector representation first, and then use statistical learning methods, such as Naive Bayesian and support vector machines, to classify it. Deep learning has made remarkable progress in the field of natural language processing in recent years[1]. BERT (Bidirectional Encoder Representations from Transformers), a pre-trained language model based on Transformer architecture, effectively captures semantic and contextual information in text by learning contextually relevant word vector representations. Captures semantic and contextual information in the text. This makes BERT a powerful tool for processing natural language tasks[2].In this thesis, we work on improving the performance of spam detection using the BERT model. Our goal is to exploit the representation capability and context understanding of the BERT model to extract key features in spam and perform accurate spam classification by a classification model. We will use a large-scale spam dataset for empirical evaluation and comparison with traditional spam detection methods to validate the effectiveness and advantages of the BERT model in spam detection. This paper is structured as follows: In Section II, we present the related research work and background of spam detection. Section III will detail the methods used in this project, including the principles and features of the BERT model. In Section IV, we compare the spam detection method based on the BERT model with other spam detection methods. Finally, Section V summarizes the main ideas of this thesis and presents the directions for future research.

2 Literature Review

In order to solve the problem of spam detection, researchers have proposed various methods for spam detection. Traditional machine learning based methods are widely studied and used in spam detection. Some of the commonly used algorithms include Naive Bayesian and support vector machines. The Naive Bayesian algorithm is based on Bayes' theorem and the assumption of conditional independence of features, and performs classification by calculating the probability of features in emails[3]. The support vector machine algorithm achieves classification by mapping the data to a high-dimensional feature space and finding the optimal hyperplane in that space[4]. These methods have achieved some results in spam detection and are widely used in practical systems.Deep learning has been applied and progressed in the field of natural language processing, bringing new opportunities for text classification tasks represented by spam detection. Among them, deep learning methods based on neural networks perform well in spam classification tasks. Deep learning models are able to extract features automatically and have strong generalization ability by learning high-dimensional representation and contextual information of text data[5].The Transformer architecture has made great breakthroughs in natural language processing tasks in recent years. The Transformer model achieves global modeling of text sequences through a self-attentive mechanism, overcoming the limitations of traditional sequence models[6].The BERT model is a pre-trained language model based on the Transformer architecture, learned through large-scale unsupervised training of Context-dependent word vector representation, which can capture rich semantic and contextual information in text. The BERT model-based spam detection approach leverages the powerful representation and context understanding capabilities of the BERT model to extract key features in spam and perform accurate spam classification by a classification model[7]. This approach no longer relies on manually designed rules and features and can better handle new fraudulent techniques and language variants in spam.

3 Method

(1) Data set description

The dataset used in the project is from Kaggle and is a set of English SMS messages containing 5574 messages, each of which is marked as either legitimate or spam[8]. In this dataset, each row contains one message, and a message consists of two parts: v1 indicates the category of the message as legitimate or spam, and v2 indicates the content of the message.

(2) Data processing

In this project, the original dataset was renamed, preprocessed, partitioned and coded. The data renaming section renames and maps the dataset with labels, renaming the "v1" column to "Class" and the "v2" column to "Text". After that, the labels are converted to digital form, i.e. "ham" is mapped to 0 and "spam" is mapped to 1.After that, the text needs to be replaced with spaces by regular expressions for characters other than letters, and then deactivated words are removed and extracted using word stems, and finally the processed word list is returned as a string by concatenating spaces to facilitate subsequent processing.The dataset needs to be divided into a training set and a test set, with the test set accounting for 20% of the total sample.Finally we use the BERT tokenizer to encode each text, generating input IDs and attention masks that convert the text into a form of input acceptable to the BERT model.

(3) Model design

The BERT model is applicable to many natural language processing tasks, including text classification, named entity recognition and sentence similarity recognition, etc. Spam detection belongs to the category of text classification tasks, and we need to predict whether a pre-processed unclassified text message is 0 (not spam) or 1 (spam), which is obviously a binary classification problem. In this regard, we can use the pooled representation of the sequence output of the BERT model as the feature representation of the whole sequence integrated information and use it as the input of the subsequent layers to finally complete the task of spam detection.Before building our spam detection model using the BERT model, it is necessary to review the BERT model, which has revolutionized the field of natural language processing since its introduction by Jacob Devlin and his team. BERT is based on the Transformer architecture, a deep learning model designed to capture the contextual relationships between words in a sentence. Unlike Transformer, the main structure of the BERT model comes from the stack of Transformer bidirectional encoders. Unlike traditional language models that process text unidirectionally, BERT exploits the bidirectional nature of the Transformer to learn the contextual representation of words by considering both the left and right contexts of words. A key feature of BERT is its pre-training strategy. Before fine-tuning for a specific downstream task, BERT is pre-trained on a large-scale corpus (e.g., Wikipedia and BookCorpus), using masked language to model the target. During pre-training, a certain percentage of tokens in the input are randomly masked, and the model needs to predict these masked tokens based on the surrounding context[2]. This process allows BERT to learn deep contextual representations that capture rich semantic information.After pre-training is complete, BERT can be fine-tuned on a variety of NLP tasks. Fine-tuning includes adding a task-specific layer on top of the pre-trained BERT model and training it with task-specific labeled data. Also fine-tuning includes hyper-parameter tuning, where different learning rates, optimizers, number of epochs, and other hyper-parameters can have an impact on the final detection results.

(4) Model structure

The spam detection model in this project is designed to have an input layer, a BERT model, a fully connected layer with 32 neurons and a ReLU activation function to process the output of the BERT model, a Dropout layer, and a fully connected layer using a Sigmoid activation function as the output layer.This model passes the input to the BERT model for feature extraction, and then processes the features through the fully connected layer and the output layer to finally output a probability value for binary prediction of spam.. The weights of the model can be updated during the training process.

4 Test

In this section, we will first discuss the impact of different hyperparameter settings (including the number of epochs, length of text sequences and batch_size) on the final performance of the model, and then we will choose the model with the best hyperparameter settings to compare with other text classification algorithms, and we will choose the accuracy and f1 values as the metrics to measure the performance of the model in this project.

(1) Epoch number

Epochs are viewed as the process of training a model by propagating the entire training data set through a neural network or machine learning algorithm in one forward and backward pass. Too few epochs may cause underfitting and the model may not learn the features and patterns of the data adequately. And too many epochs may cause overfitting problems, which can cause the model to overfit the training set and lead to poor performance on unseen datasets, while too many epochs can lead to long training times and huge computational resources. To be able to observe the effect of training at different epochs, we first set the number of training rounds to a larger value, i.e., 20. We can change the code to save its performance metrics within each training round, and finally, for visual representation, we give these data as a line graph, as shown in Figure 1.

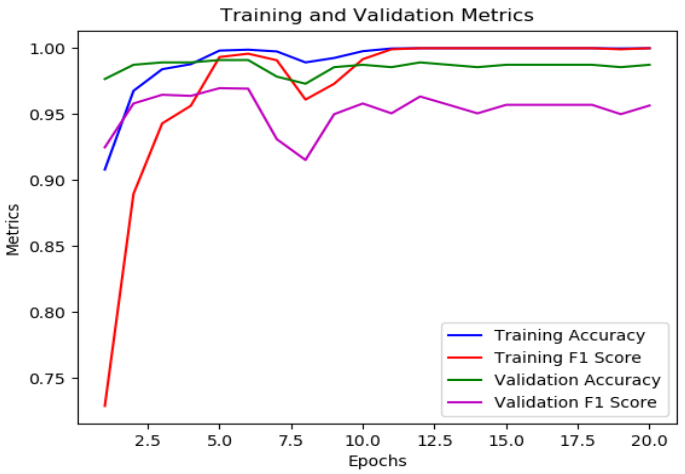

Figure 1. Accuracy and F1 values of the model under different epochs

Observing Figure 1, we can find that after the epochs of the model are greater than 10, the accuracy and f1 values of the model no longer change significantly, and the model has started to overfit on the training set at this time. In order to obtain an acceptable performance, we finally choose

to set the value of epochs to 10.

(2) Text sequence length

The text sequence length determines the fixed length of the model input and also affects the input dimension of the model. Longer text sequence length will retain more contextual information and will perform better in terms of semantic comprehension. However, this comes with a cost, such as increased computational and storage resource consumption. Shorter text sequence length will save the above resources consumption, but it may lose some contextual information and make the model's performance on some tasks degrade. We set the text sequence length to 32 and 64, respectively, and observe the performance of the model after training for 10 epochs, as shown in Figure 2.

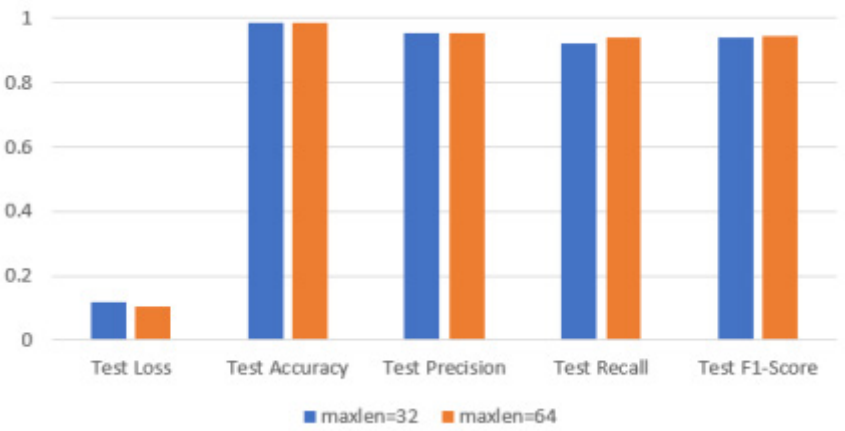

Figure 2. The performance of the model with the text sequence length set to 32 and 64

It can be observed in Figure 2 that there is no significant difference in the performance of the model in the test set when the text sequence is set to 32 and 64, but a smaller text sequence length can bring faster training speed in this dataset while ensuring the training effect. Therefore, we finally choose 32 as the length of the text sequence.It is worth noting that the setting of the text sequence length is related to the selection of the dataset. A shorter text sequence seems to be a better setting in this project, because the length of the text in the dataset we use is generally short, and a text sequence length setting of 32 is sufficient for most of the data. For datasets with longer text, we need to choose a longer and appropriate text sequence length.

(3) Batch size

Batch size is the number of samples that are passed into the model at each pass during model training, and it serves to control the number of samples that the model references at each parameter update. A larger batch_size can speed up the training because less computation is required for each parameter update and more memory is needed to store samples and compute intermediate results. A larger batch_size may also lead to faster convergence of the model. However, blindly using a larger batch size may bring memory consumption and may cause the model to fall into local minima or plain regions, while a larger batch size is more likely to overfit the training set. Therefore choosing an appropriate batch size is necessary. We set the batch size to 16, 32 and 64 respectively, and the length of text sequence is 32, and train 10 epochs. the training results are shown in Figure 3.

Figure 3. Model performance under different batch size

By looking at Figure 3, we can see that the model performs significantly better when the batch_size is 16 than the model performance under other batch size settings, while other settings of batch_size values do not significantly improve the performance of the model.

(4) Comparison

In this subsection, we will compare the tuned model with other machine learning-based text classification algorithms such as Naive Bayesian classifiers, support vector machines, and decision trees. Again, we use Accuracy and f1 values as a measure of model performance. The performance of the final bert-based model and the other models on the same dataset is shown in Figure 4.

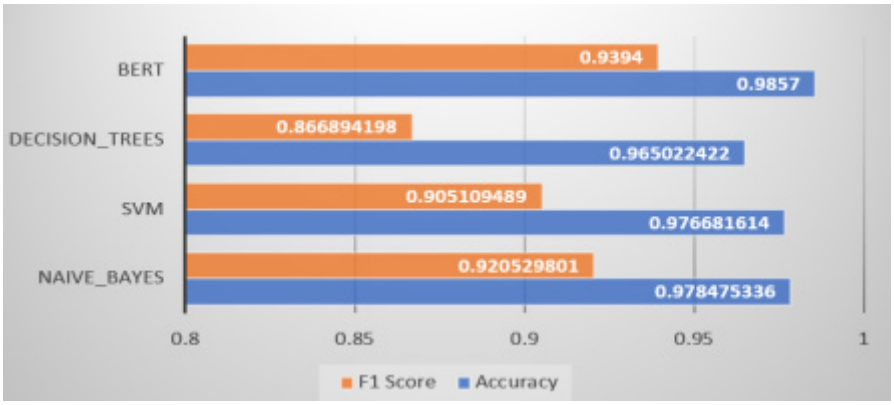

Figure 4. Performance of different models

By observing Figure 4, we find that the spam detection model based on the BERT model has the highest accuracy and f1 value. This indicates that the BERT model can accomplish the detection of spam with high accuracy while ensuring the ability to identify the real spam among all emails in the dataset.

5 Conclusion and Prospect

The BERT model has achieved excellent results in the field of natural language processing since it was proposed. We exploited the powerful context understanding capability of the BERT model in this study and used it in the task of spam detection and achieved acceptable results. Through experimental comparisons, we see the excellent performance of BERT in the spam classification task. In our

experiments, the BERT model achieves 98.57% accuracy and 0.9394 F1 value, showing its powerful ability to handle the spam classification problem. Compared with traditional machine learning algorithms, such as Naive Bayes and support vector machine (SVM), the BERT model performs better in the spam classification task. Its high level of representational power and context understanding enables it to better capture semantic and contextual information in text, thus improving classification performance. With the rapidly changing content of spam on today's Internet, the BERT model has demonstrated superior capability in identifying and detecting new spam messages compared to traditional machine learning algorithms. The study has achieved remarkable results; however, there are still opportunities for further exploration and improvement. The spam dataset we used in this study is relatively small, and future work can consider collecting larger and more diverse spam datasets, and by feeding the model with these larger scale data, we believe the model will show more powerful spam discrimination capabilities. Although our tuned BERT model performs well in this spam classification task, it does not mean that we have found the best tuning strategy for the model under the spam classification task, and the performance of the model may also be improved by adjusting other hyperparameters, while the relationship between hyperparameters may also affect the final performance of the model. Finally, although BERT performs well in the classification task, its black-box nature limits the interpretation and understanding of the model decisions. Future work can explore interpretive approaches based on the BERT model to better understand the decision process and key features of the model.

About the Author

Hailin Xu is undergraduate of Huazhong University of Science and Technology, and his research field is computer science and technology.

References

[1] Dada, E.G., Bassi, J.S., Chiroma, H., Abdulhamid, M., & Ajibuwa. (2019). Machine learning for email spam filtering: review, approaches and open research problems. Heliyon. pp. 1-23.

[2] Devlin, J., Chang, M.W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North. Minnesota. pp. 1-16.

[3] Hovold, J. (2005). Naive bayes spam filtering using word-position-based attributes. In: Conference on Email and Anti-Spam. California. pp. 41-48.

[4] Drucker, H., Wu, D., Vapnik, V. N. (2002). Support vector machines for spam categorization. In: IEEE Transactions on Neural Networks. New York. pp. 1048-54.

[5] Shahariar, G. M., Biswas, S., Omar, F., Shah, F. M., & Hassan, S. B. (2019). Spam review detection using deep learning. In: 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). Vancouver. pp. 0027-0033.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., ... & Polosukhin, I. (2017) Attention is all you need. In: Advances in Neural Information Processing Systems. Long Beach, California. pp. 30.

[7] Tida, V.S., & Hsu, S.H. (2022). Universal spam detection using transfer learning of BERT model. In: Proceedings of the Annual Hawaii International Conference on System Sciences, Hawaii International Conference on System Sciences. Hawaii. pp. 1-9.

[8] SMS Spam Collection Dataset. (2016). https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset