Spam Detection and Classification Based on DistilBERT Deep Learning Algorithm

Tianrui Liu[1], Shaojie Li[2], Yushan Dong[3], Yuhong Mo[4] and Shuyao He[5]
[1]*Electrical and Computer Engineering, University of California San Diego, La Jolla, United States*
[2]*Huacong Qingjiao Information Technology (Beijing) Co., Ltd., Beijing, China*
[3]*University of Maryland, MD, United States*
[4]*Electrical and Computer Engineering, Carnegie Mellon University, United States*
[5]*Northeastern University, Boston, United States*

[1]*Corresponding Author: til028@ucsd.edu*

Received: 09-04-2024 Revised: 23-04-2024 Accepted: 10-05-2024

ABSTRACT
This paper discusses the importance of spam classification in the field of information security. With the popularity of the Internet and email, spam has become one of the major issues affecting user experience and information security. The study begins with preprocessing text data in various ways, including converting to lowercase, removing irrelevant content, links, punctuation, etc., and filtering deactivated words and words of length 1. By applying the DistilBERT model to the text classification task, the results show that it achieves 93% accuracy in spam classification, effectively distinguishing between spam and non-spam emails. The confusion matrix showed that 18,500 emails were correctly classified and a small number of spam emails were misclassified as non-spam emails. Overall, the DistilBERT model showed high accuracy in spam classification, but more algorithms are still expected to emerge to improve the prediction accuracy. This study provides a useful reference for improving spam filtering systems in the future, which is expected to further enhance user experience and information security.

Keywords: *spam detection, distilbert, accuracy, classification*

I. INTRODUCTION

Spam classification is one of the important issues in the field of information security, and with the popularity of the Internet and the widespread use of email, spam has become one of the main factors affecting user experience and information security [1]. Researchers have begun to focus on how to identify and filter spam through automated techniques to reduce the disruption and threat to users [2]. Traditional rule-based or keyword-matching approaches are gradually showing limitations in the face of ever-changing forms of spam, so natural language processing and machine learning techniques have been introduced to improve classification accuracy and generalisation.

Natural Language Processing (NLP) is a discipline that studies the interaction between human language and computers, and in spam classification, NLP techniques are widely used in text data preprocessing, feature extraction and model training [3]. Firstly, in the text data preprocessing stage, NLP can help to convert the raw text data into a form that can be understood and processed by computers, such as word splitting, deletion of stop words, stemming extraction and other operations. Second, in the feature extraction stage, NLP can help extract key information and features from text data, thus providing useful input for subsequent model training. Finally, in the model training phase, the machine learning model combined with NLP technology can effectively learn the hidden patterns and laws in the text data and classify them accurately [4].

In spam classification, commonly used machine learning models include Simple Bayes, Support Vector Machines, Decision Trees, Random Forests, and so on. These models can be used to build classifiers by exploiting features in text data and automatically classify newly received emails [5]. Typically, researchers divide the dataset into a training set and a test set, train the model and tune the parameters on the training set, evaluate the classifier performance on the test set, and further improve the model based on the evaluation results.

Overall, natural language processing machine learning models play a crucial role in spam classification. By continuously optimising the algorithms, enriching the feature representations, and improving the model structure, the accuracy, efficiency, and robustness of the spam classification system can be improved to better protect the users from the nuisance and risk caused by spam.

II. DATA SET SOURCES AND DATA ANALYSIS

The dataset used in this paper is selected from the open source spam dataset published by MIT, which contains more than 190,000+ emails labelled as spam and non-spam. Each email is represented by its text content and its corresponding tag. Part of the dataset is shown in Table 1.

Table 1. Partial text data

Label	Text
Spam	you registered to receive this and similar off...
Ham	would it be unreasonable to require one cent i...
Ham	market notice january escapenumber escapenumbe...
Spam	hi get yescapenumberur hard rescapenumberck es...
Spam	how are you today i am dr thomas fernandez a c...

III. DATA PREPROCESSING

The data preprocessing process begins by defining three functions for text data preprocessing, namely text preprocessing, drop stopwords, and delete one characters.The text preprocessing function performs a number of processes on the text data, including converting the text to lowercase, removing the contents of square brackets, non-word characters, links, HTML tags, punctuation, line breaks and words containing numbers. Next, the drop stopwords function removes stop words from the text based on a collection of English stop words. Finally, the delete one characters function removes words of length 1 from the text to avoid interference with subsequent classifiers.

Subsequently, in applying the preprocessing functions and label encoding functions to the dataset, the original dataset was first copied to a new dataset full data. Then, the complete preprocessing process for text data was achieved by applying the preprocessing function to each line of text data named text column in turn and storing the processing results in the new column preprocessed text. Finally, the label encoder is used to encode the classification labels in the column named label and the encoding results are stored in the new column encoded_label for use in the machine learning model. The preprocessed dataset is shown in Table 2.

Table 2: Data preprocessing

Label	Text	Preprocessed text	Encoded label
Spam	you registered to receive this and similar off...	registered receive similar offers youve heard ...	1
Ham	would it be unreasonable to require one cent i...	would unreasonable require one cent increments...	0
Ham	market notice january escapenumber escapenumbe...	market notice january escapenumber escapenumbe...	0
Spam	hi get yescapenumberur hard rescapenumberck es...	hi get yescapenumberur hard rescapenumberck es...	1
Spam	how are you today i am dr thomas fernandez a c...	today dr thomas fernandez consultant cardiolog...	1

IV. METHOD

DistilBERT is a lightweight BERT-based model designed to reduce the size and computational cost of BERT models while maintaining high performance [6]. DistilBERT achieves this by using distillation techniques to extract knowledge from large, pre-trained language models, and then transfer this knowledge to smaller, faster models, thus achieving reduced performance while maintaining the computational resource requirements.

The model structure of DistilBERT is shown in Fig. 1, which is based on the Transformer architecture and includes multiple layers of Transformer encoders. Compared with the original BERT, DistilBERT employs several streamlining and optimisation strategies [7,8]. Firstly, DistilBERT reduces the number of layers and hidden units of the Transformer encoder to reduce the number of parameters. Second, DistilBERT introduces a technique called "knowledge distillation" to improve performance by allowing smaller models to learn how to reproduce the predictions of larger models. In addition, during pre-training and fine-tuning, DistilBERT employs a number of regularisation methods to further improve generalisation [9].

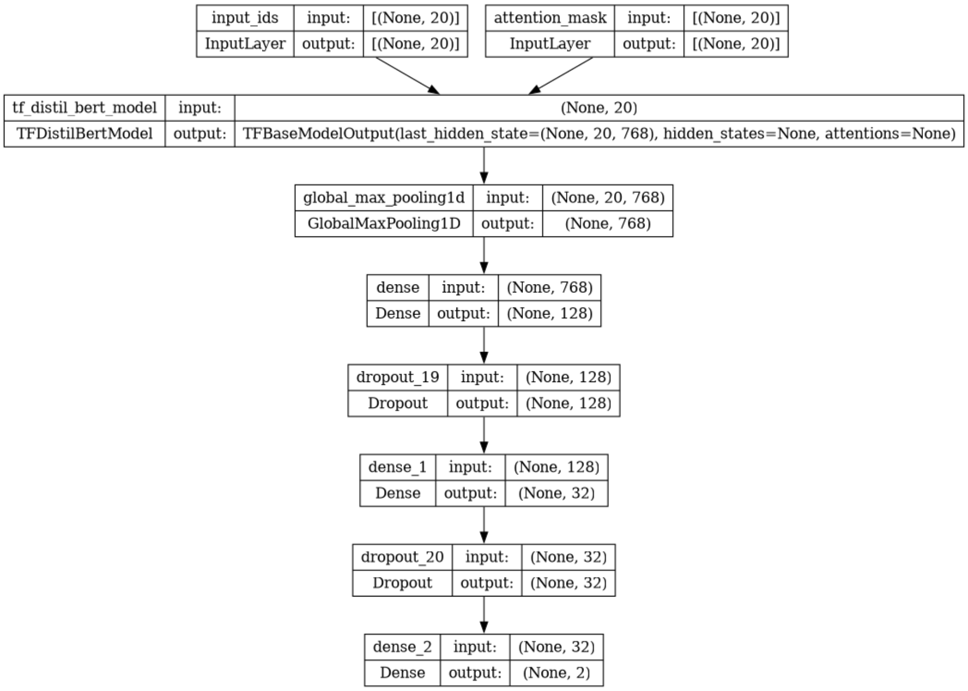
input_ids | input: | [(None, 20)]
InputLayer | output: | [(None, 20)]
attention_mask | input: | [(None, 20)]
InputLayer | output: | [(None, 20)]
tf_distil_bert_model | input: | (None, 20)
TFDistilBertModel | output: | TFBaseModelOutput(last_hidden_state=(None, 20, 768), hidden_states=None, attentions=None)
global_max_pooling1d | input: | (None, 20, 768)
GlobalMaxPooling1D | output: | (None, 768)
dense | input: | (None, 768)
Dense | output: | (None, 128)
dropout_19 | input: | (None, 128)
Dropout | output: | (None, 128)
dense_1 | input: | (None, 128)
Dense | output: | (None, 32)
dropout_20 | input: | (None, 32)
Dropout | output: | (None, 32)
dense_2 | input: | (None, 32)
Dense | output: | (None, 2)

Figure 1: DistilBERT Structure
（Photo credit : Original）

In principle, DistilBERT utilises a teacher-student network framework for knowledge distillation. A teacher-student network is a framework in which a large teacher model (BERT) and a small student model (DistilBERT) participate in training. During training, the teacher model generates soft labels, i.e., predictions in the form of probability distributions; while the student model attempts to learn the knowledge embedded in the teacher model by minimising the difference between the soft labels and its own predictions. In this way, although the student model is small, it can still acquire information from the rich representation space of the teacher model and gradually approach or even surpass the teacher model performance [10].

Overall, DistilBERT, as a lightweight and efficient language representation learning model, shows good performance in natural language processing tasks. Through distillation techniques and structural optimisation, it reduces resource consumption while maintaining high accuracy, allowing more application scenarios to benefit from the powerful pre-trained language representation learning technique.

V. RESULT

In this paper, DistilBERT model is used for text classification task. Firstly, the input layers input ids and input mask are defined, then the inputs are encoded to get the embeddings representation by DistilBERT model, and then the pooling operation is performed on the encoded results by the global max pooling layer global maxPool 1D. This is followed by a series of sense layers for feature extraction and classification, including a sense layer with an activation function of GELU, a dropout layer to prevent overfitting, and a sense layer with a softmax activation function of output dimension 2 for multiclassification prediction. Finally, two callback functions, early stopping and model checkpoint, are defined for implementing early stopping and saving the optimal model weights during the training process. The experiments are based on python 3.10, epoch is set to 30, and the dataset is allocated by dividing the training, validation and test sets in the ratio of 4:3:3.

The accuracy of the output model for spam prediction is shown in Table 3. Output the confusion matrix of the model for spam and non-spam prediction and the results are shown in Fig. 2.

Table 3: Model evaluation parameter

	Precision	Recall	F1 score	Support
Ham	0.88	0.99	0.93	10554
Spam	0.98	0.86	0.92	9446
Accuracy			0.93	20000
Macro avg	0.93	0.92	0.92	20000
Weighted avg	0.93	0.93	0.92	20000

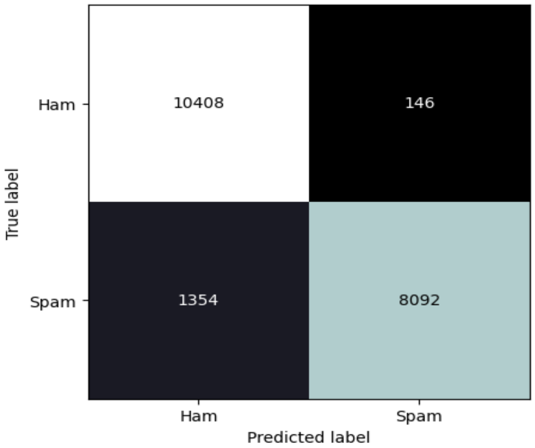

Figure 2: Confusion matrix
（Photo credit : Original）

The Accuracy, Precision, Recall and F1 scores of the model are shown in Figure 3.

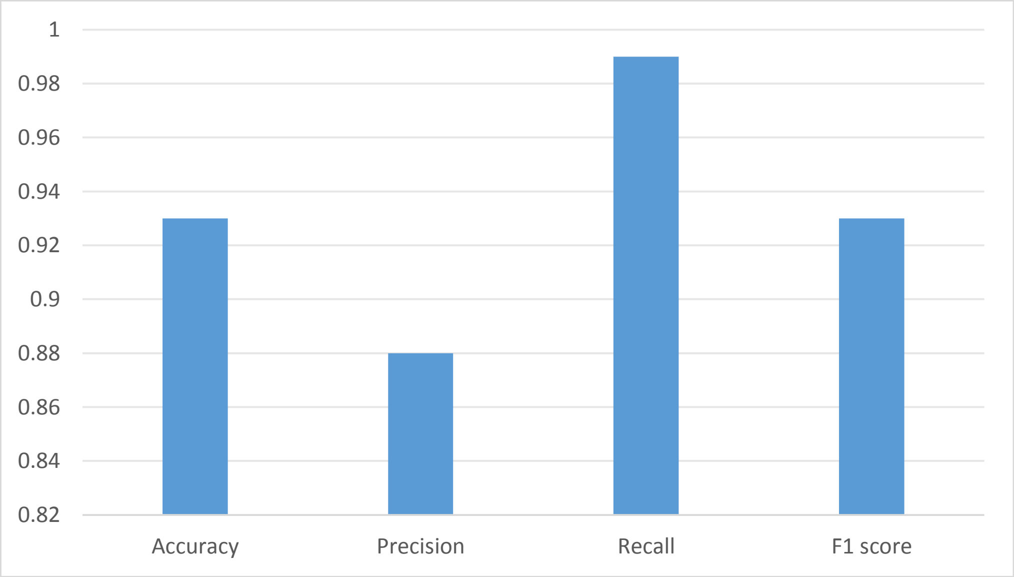

Figure 3. Modelling evaluation
（Photo credit : Original）

VI. CONCLUSION

Spam classification has been one of the issues of great concern in the field of information security. With the popularity of the Internet and the widespread use of email, spam has become an important factor affecting user experience and information security. In order to effectively identify and filter spam emails, this paper adopts a series of text data preprocessing steps and DistilBERT model for the classification task.

Firstly, the raw text data was cleaned and normalised by converting the text to lowercase letters, removing content in square brackets, non-word characters, links, HTML tags, punctuation, line breaks and words containing numbers. Then, the English deactivated words collection was used to remove the deactivated words in the text, and words of length 1 were deleted, thus reducing the interference of noise on the classification results.

During the application of DistilBERT model, a satisfactory classification accuracy of 93% was achieved. The model demonstrated good performance in distinguishing spam and non-spam emails, showing in the confusion matrix that most of the emails were correctly classified, but there were also a small number of spam emails that were misclassified as non-spam emails and non-spam emails that were misclassified as spam emails.

Taken together, these results show that the DistilBERT model performs well on the spam classification task, but there is still room for improvement. In the future, we expect to further optimise the algorithm to improve the prediction accuracy and explore more efficient and accurate spam classification methods. Through continuous improvement and innovation, we can more effectively protect users from spam, improve information security, and provide users with a better network experience.

REFERENCES

1. Mu, Pengyu, Wenhao Zhang., & Yuhong Mo. (2021). Research on spatio-temporal patterns of traffic operation index hotspots based on big data mining technology. *Basic & Clinical Pharmacology & Toxicology. 128*(111), River ST, Hoboken 07030-5774, NJ USA: Wiley.
2. Mo, Y., Qin, H., Dong, Y., Zhu, Z., & Li, Z. (2024). Large language model (llm) ai text generation detection based on transformer deep learning algorithm. *International Journal of Engineering and Management Research*, *14*(2), 154-159.
3. Sumathi, V. P., V. Vanitha., & R. Kalaiselvi. (2023). Performance comparison of machine learning algorithms in short message service spam classification. *2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA).* IEEE.
4. Ma, Danqing, Bo, Dang, Shaojie, Li, Hengyi, Zang., & Xinqi, Dong. (2023). Implementation of computer vision technology based on artificial intelligence for medical image analysis. *International Journal of Computer Science and Information Technology*, *1*(1), 69–76.
5. Zhu, Armando, Jiefeng Li., & Cewu Lu. (2021). Pseudo view representation learning for monocular RGB-D human pose and shape estimation. *IEEE Signal Processing Letters 29*, 712-716.
6. Li, Yanjie, et al. (2021). Transfer-learning-based network traffic automatic generation framework. *6th International Conference on Intelligent Computing and Signal Processing (ICSP)*. IEEE.
7. Liu, Tianrui, et al. (2024). *Rumor Detection with a novel graph neural network approach.* arXiv preprint arXiv:2403.16206.
8. Mo, Y., Qin, H., Dong, Y., Zhu, Z., & Li, Z. (2024). Large language model (llm) ai text generation detection based on transformer deep learning algorithm. *International Journal of Engineering and Management Research, 14*(2), 154-159.
9. Zhang, Jingyu, et al. (2024). *Research on detection of floating objects in river and lake based on AI intelligent image recognition.* arXiv preprint arXiv:2404.0688.
10. Xiang, Ao, et al. (2024). *Research on splicingimage detection algorithms based on natural image statistical characteristics.* arXiv preprint arXiv:2404.16296.
11. Li, Zhenglin, et al. (2023). Stock market analysis and prediction using LSTM: A case study on technology stocks. *Innovations in Applied Engineering and Technology*, 1-6.
12. Li, Shaojie, Yuhong Mo., & Zhenglin Li. (2022). Automated pneumonia detection in chest x-ray images using deep learning model. *Innovations in Applied Engineering and Technology,* 1-6.
13. Dai, Shuying, et al. (2024). The cloud-based design of unmanned constant temperature food delivery trolley in the context of artificial intelligence. *Journal of Computer Technology and Applied Mathematics, 11,* 6-12.