The Study of Performance for Cross-Platform Spam Filtering Based on the Random Forest Algorithm

Zhengchi Ma[1, *, †], Ruoyu Ouyang[2, †] and Hanzhang Wang[3, †]

[1]School of Statistics and Data Science, Nankai University, Tianjin, China

[2]Department of Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong, China

[3]School of Applied Mathematics and Statistics, Shandong University, Weihai, China

*Corresponding author: zhengchima@mail.nankai.edu.cn

†These authors contributed equally.

Abstract. The objective of this study was to investigate the performance of the Random Forest algorithm in spam detection when generalized from email spam to social media comment spam. The dataset used involved the use of two sources: an email dataset and a YouTube spam comment dataset. Text processing techniques and feature extraction methods were applied to preprocess the datasets using scikit-learn package. Labels were mapped from "spam" and "ham" to "1" and "0" respectively for training and testing the model. The email spam dataset was split into training and testing datasets, and the first 3000 lines were used for training the model. The generalization ability of the model was tested on the YouTube spam comment dataset. Multiple decision trees were created using the Random Forest algorithm and were trained on different subsets of the training data. The results indicated that the accuracy rate of the prediction on the YouTube spam comment dataset was only around 62%, which is comparatively low. This suggests that the Random Forest algorithm, when used for spam detection, may not have good enough generalization ability to be applied in practice. Additionally, as the number of trees increased, the maximum accuracy decreased, indicating the possibility of overfitting. Although the accuracy of the models was modest, possible improvements could be made to the pre-processing of the data so that the features extracted from the text can have greater conformity with social media spams. In conclusion, further work is needed before the model can be used in generalized situations.

Keywords: Random Forest Algorithm; Machine Learning; Spam Prediction.

1. Introduction

The term "spam" connotes the dissemination of unsolicited and/or unwanted messages in large quantities across various electronic communication channels such as email, text messages, and social media. The messages are typically sent with the intention of promoting a product or service, or with the goal of tricking recipients into providing personal information or clicking on malicious links. Stavroulakis et al. note that spam can be used as a tool for fraud, phishing, and other forms of malicious activity [1].

One of the main harms of spam is that it is a major nuisance for recipients. The sheer volume of unwanted messages that individuals receive can be overwhelming, leading to frustration, stress, and a general feeling of being overwhelmed. This can be especially problematic for businesses, as spam messages can distract employees from their work and reduce overall productivity. Another harm of spam is that it can be a drain on resources. Sorting through and deleting large volumes of unwanted messages takes time and effort, which can be especially problematic for businesses that receive a high volume of spam. Additionally, some spam messages may be filtered into a separate folder, which can lead to important messages being overlooked or missed.

Overall, spam is harmful in a variety of ways and can have a significant impact on individuals and businesses alike. It is important for individuals to be vigilant in protecting themselves against spam, and for businesses to take steps to prevent spam from reaching their employees and customers.

The pervasiveness of spam in the digital environment is a significant challenge that individuals and organizations face on a daily basis. As a result, researchers have focused on developing effective techniques for spam detection to mitigate its adverse impact.

One approach to spam detection is content-based filtering, which involves analyzing the message content to identify patterns or characteristics that are typical of spam messages. Gordon et al. conducted a systematic review of email spam filtering techniques [1]. Researchers have used various machine learning algorithms to classify emails as spam or not based on their content features due to their satisfactory performance in various tasks [2-4], such as keywords, subject lines, message body, and attachment type. For example, Naive Bayes, Decision Trees, and Support Vector Machines have been used for spam classification with high performance [5-7]. Naive Bayes is a probabilistic algorithm that calculates the probability of a message being spam or not based on the frequency of certain words or phrases in the message. Decision Trees use a hierarchical tree structure to classify messages based on a series of binary decisions, such as the presence or absence of specific keywords. Support Vector Machines are a type of supervised learning algorithm that classifies messages by identifying the best hyperplane that separates spam from non-spam messages in a high-dimensional feature space. Another approach to spam detection is sender-based filtering, which involves analyzing the sender's reputation and behavior to determine whether the message is likely to be spam. This approach can be effective in identifying messages from known spammers or sources with a history of sending spam. For example, Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM), and Domain-based Message Authentication, Reporting, and Conformance (DMARC) are authentication protocols that can verify the authenticity of email senders and detect spoofed messages.

In addition to content-based and sender-based filtering, researchers have also explored other techniques for spam detection, such as behavior-based filtering [8], which involves analyzing user behavior to detect unusual patterns that may indicate spam activity, and hybrid approaches that combine multiple techniques to achieve higher accuracy rate.

Overall, previous research has demonstrated that a combination of content-based and sender-based filtering techniques can be effective in detecting spam messages with high accuracy rates. However, spam detection remains a challenging problem due to the constant evolution of spam tactics and the need for ongoing monitoring and adaptation of detection techniques. Further research is needed to develop more robust and adaptive techniques for spam detection that can keep pace with the changing landscape of spam activity.

In this study, we employed the use of the random forest algorithm for classification purposes. We are continuously adjusting the model's parameters to achieve the best possible performance in identifying and classifying spam or toxic comments. To train our model, we utilized the "YouTube" database from Goneee, United States, which is available on the Kaggle platform. Random forest algorithm offers several advantages, including reducing overfitting, handling missing data, and avoiding the need for feature scaling. It is considered one of the most reliable algorithms in the field of machine learning and has a wide range of applications across various industries. The experimental results demonstrated the superiority of the method.

2. Method

2.1. Dataset description and preprocessing

Two datasets provided by Kaggle were utilized in this study [9, 10]. The first dataset consists of 5572 lines, containing information about the content of emails and their corresponding labels, which were further divided into training and testing datasets. The other dataset includes 204 comments from YouTube spam, along with their labels.

There were three main procedures of the data preprocessing. Firstly, general text processing was performed according to the "bag-of-word" model, transforming the data into a standardized format by removing unnecessary columns, converting words into lowercase, and discarding punctuation. Secondly, after all of the processing above, the feature of the text could be extracted directly using

the “feature_extraction” method in scikit-learn package. While doing this, it was necessary to delete some stop word in English. The labels of the text were mapped from {“spam”, “ham”} to {“1”, “0”} so that we could train and test the model. Thirdly, the datasets were split into training and testing datasets. The first 3000 lines in the email spam dataset were used to train the model while the last of this dataset were used to test the validity of the model we trained. As for the Youtube spam comment dataset, it was used for testing the generalization ability of the model.

2.2. Random Forest

The random decision forest algorithm is a highly regarded and commonly used machine learning algorithm that extends the decision tree algorithm to build a decision-making model based on input data. It is widely used in classification and regression tasks across diverse domains such as finance, healthcare, and social media analysis due to its accuracy and robustness. The random decision forest algorithm works by creating multiple decision trees that are trained on different subsets of the training data. The final prediction for the given input is obtained by aggregating each decision tree's output. The randomness in the algorithm arises from training each decision tree on a randomly selected subset of features and training samples.

To train the decision trees, a unique training set, denoted by Si, is chosen for each decision tree. If the termination condition is reached at a node, it is called a leaf node. For a classification problem, the output of this node is the class Cj that contains more samples than all others, where p is the proportion of Cj in the current sample set. For a regression problem, the output is the average value of each sample value in the current node sample set. If the current node does not meet the termination condition, the best feature from all feature dimensions is chosen, and the best threshold number Th is found. The samples are divided into different nodes by comparing their feature values with Th. The above process is repeated for all nodes, and the tree can make a prediction for any sample.

The algorithm's strength lies in its ability to handle complex datasets with many features and instances while reducing the risk of overfitting by introducing randomness. It is also scalable and can handle large datasets with ease. In summary, random decision forests are a robust and accurate machine learning algorithm used for various machine learning problems. Their popularity and effectiveness are demonstrated by their use in real-world applications and research studies.

2.3. Implementation Details

In this study, we employed the Random Forest algorithm to assess the generalization ability of the model we trained on the email spam datasets. Fig. 1 provides the general procedure of the proposed method. The general idea is to train a feasible random forest model on the email spam dataset and then test the accuracy of it on the datasets of a different system like the social media spam comment datasets or the SMS short message spam datasets. Specifically, we tested the generalization ability of the model on the Youtube spam comment dataset. In order to find the model with the best performance, we gradually varied the number of trees in the forest while training the models and then test their performance individually on both the email spam dataset and the Youtube spam comment dataset. This was realized by varying the parameter “n_estimators” in the class RandomForestClassifier from scikit-learn package. The models with “n_estimators” from 1 to 300 were trained, after which the results were visualized by using matplotlib. Then the results could be compared and analyzed so that we could have a general image about the generalization ability of the spam filtering model based on the random forest algorithm.

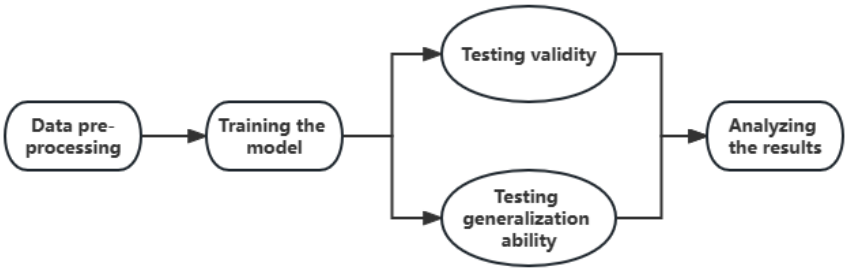

Fig. 1 The workflow of the proposed method.

3. Results and Discussion

The Random Forest models were trained with varying number of trees in the forest on the data set of email spams. After finishing training each of the model, the validity of them were first tested on email spams testing data set. Most of the testing accuracy were over 96%, which indicated that the models were valid on email spam filtering. After determining the validity of the models, they were then applied on the Youtube spam comment detecting. The models with different n_estimators (indicating the number of trees) varying from 1 to 300 were tested and the result is shown in Fig. 2.

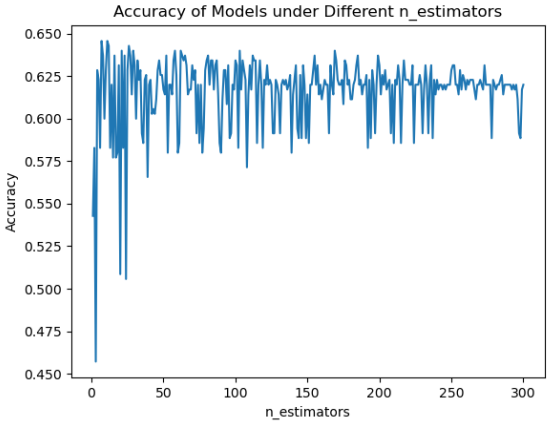

Fig. 2 The accuracy of random forests under different n_estimators.

As is shown in Fig. 2, the accuracy of the prediction is highly volatile while changing the number of trees. The accuracy was relatively low when the number of trees is too small. While the number of trees increasing, the model performed the best when there are 7 trees, with the accuracy over 64.5%. After the trees were increased to more than 50, the accuracy became more stable than the situation with less trees and fluctuated between 58% and 63% approximately. Generally, the accuracy of the prediction on Youtube spam data set was around 62%.

According to the results, it is evident that the accuracy rate of the prediction on Youtube spam comment data set was approximately 62%, which is considerably low. This indicates that the Random Forest algorithm, when carrying out spam detecting generalized from email spam to social media comment spam, does not have the generalization ability that is good enough to be applied in practice. Besides, as the number of trees increases, the maximum of accuracy decreased in general, which means over-fitting might appear when there are too many unnecessary trees. The accuracy of the models was relatively low but there are still some possible ways to get it improved. For example, some changes could be made while pre-processing the data so that the features extracted from the text can have greater conformity with the social media spam. In general, there is still plenty of works that should be conducted before the model can be used in generalized situations.

4. Conclusion

In conclusion, this study aimed to assess the generalization ability of a spam filtering model trained on email datasets using the Random Forest algorithm. We gradually varied the number of trees in the forest during training and conducted extensive experiments to evaluate the model's effectiveness and generalization ability. Our experimental results demonstrated that the Random Forest models have some generalization ability and performed the best when there are 7 trees. The generalization accuracy is not high enough to meet the requirement for practical use but these findings do suggest that the Random Forest algorithm has the promise be an effective tool for developing spam filtering models that can be applied to various datasets and scenarios after some other procedures. Moving forward, we plan to promote and extend the application of our model to more scenarios. This could involve exploring additional datasets, fine-tuning the model for specific applications, or incorporating new features or algorithms to further enhance its performance. Overall, the results of this study contribute to the development of effective and versatile spam filtering models that can be utilized in various domains.

References

[1] Gordon V. Cormack. Email Spam Filtering: A Systematic Review, Foundations and Trends® in Information Retrieval: Vol. 1: No. 4, pp 335-455. http://dx.doi.org/10.1561/1500000006, 2008.

[2] Khaidem L, Saha S, Dey S R. Predicting the direction of stock market prices using random forest. arXiv preprint arXiv:1605.00003, 2016.

[3] Yu Q, Wang J, Jin Z, et al. Pose-guided matching based on deep learning for assessing quality of action on rehabilitation training. Biomedical Signal Processing and Control, 2022, 72: 103323.

[4] Petre E G. A decision tree for weather prediction. Universitatea Petrol-Gaze din Ploiesti, 2009, 61(1): 77-82.

[5] Hovold J. Naive Bayes Spam Filtering Using Word-Position-Based Attributes. CEAS. 2005: 41-48.

[6] Wijaya A, Bisri A. Hybrid decision tree and logistic regression classifier for email spam detection, 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE). IEEE, 2016: 1-4.

[7] Amayri O, Bouguila N. A study of spam filtering using support vector machines. Artificial Intelligence Review, 2010, 34: 73-108.

[8] Wu C H. Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks. Expert systems with Applications, 2009, 36(3): 4321-4330.

[9] Kaggle, YouTube Spam Collection Data Set, https://www.kaggle.com/datasets/lakshmi25npathi/images, 2019.

[10] Kaggle, Spam Email, https://www.kaggle.com/datasets/mfaisalqureshi/spam-email, 2021.