DEFEND YOUR INBOX: SPAM EMAIL DETECTION WITH MACHINE LEARNING

Leping Li *
Department of Statistics and Data Science
Southern University of Science and Technology
12112627@mail.sustech.edu.cn

Qiang Hu *
Department of Statistics and Data Science
Southern University of Science and Technology
12111214@mail.sustech.edu.cn

Jingjuan Huang *
Department of Statistics and Data Science
Southern University of Science and Technology
12112847@mail.sustech.edu.cn

Junjie Qiu *
Department of Statistics and Data Science
Southern University of Science and Technology
12111831@mail.sustech.edu.cn

ABSTRACT

Spam email detection is the task of identifying and filtering unwanted or malicious messages that are sent to a large number of recipients via email. Spam emails can cause various problems for users, such as wasting their time and resources, compromising their security and privacy, and exposing them to scams, malware, or phishing. Therefore, developing effective and robust spam filters is an important and challenging problem for email service providers and researchers. In this paper, we implement several machine learning methods, including SVC, Logistic Regression, Decision Tree and Random Forest for monolingual spam email detection. We test model performance on several datasets including English and Chinese emails. We conduct experiments delving into factors like language, data size, distribution, model type and more. We apply reverse fourier transformation to SVC and try to improve its efficiency. This is a novel attempt to use a low parametric approximation to speed up the training process. The code is available at GitHub.

1 INTRODUCTION

Spam email detection is the task of identifying and filtering unwanted or malicious messages that are sent to a large number of recipients via email. Spam emails can cause various problems for users, such as wasting their time and resources, compromising their security and privacy, and exposing them to scams, malware, or phishing. Therefore, developing effective and robust spam filters is an important and challenging problem for email service providers and researchers.

Previous work on spam email detection can be broadly categorized into two approaches: rule-based and machine learning-based (Revar et al., 2017). Rule-based methods rely on manually crafted rules or heuristics that capture the characteristics or patterns of spam emails, such as the sender's address, the subject line, the keywords, or the attachments. These methods are easy to implement and interpret, but they require constant updating and maintenance as spammers change their strategies and techniques to evade detection. Moreover, rule-based methods may not generalize well to new or unseen types of spam emails (Akinyelu, 2021), and they may produce false positives or negatives.

Machine learning-based methods, on the other hand, use algorithms that learn from labeled or unlabeled data to automatically classify emails as spam or non-spam. These methods can adapt to the dynamic and adversarial nature of spam emails, and they can achieve high accuracy and performance (Jáñez-Martino et al., 2023).

Though deep learning has been widely used in many natural language processing tasks and has achieved great success, we want to focus on the traditional machine learning methods in this paper.

*Equal Contribution

To verify the effectiveness of these methods, we will use several datasets to train and test various machine learning models, and compare their performance in terms of accuracy, time.

We summerize our contributions as follows:

- We implement several machine learning methods, including SVC, Logistic Regression, Decision Tree and Random Forest for monolingual spam email detection.
- We test model performance on several datasets including English and Chinese emails. We conduct experiments delving into factors like language, data size, distribution, model type and more.
- We apply reverse fourier transformation to SVC and try to improve its efficiency. This is a novel attempt to use a low parametric approximation to speed up the training process.

2 PRELIMINARIES

Let $\mathcal{X}$ be the input space and $\mathcal{Y}$ be the output space. Every input vector $\mathbf{x} \in \mathcal{X}$ is created by specifying whether a token embedding is present or absent in the email ($x_i = 1$ *or* 0). The output y can be either 0 (non-spam) or 1 (spam). The goal of the spam email detection task is to learn a function $f : \mathcal{X} \to \mathcal{Y}$ that maps the input to the output. In this project, we will use the following machine learning models to learn the function f:

SUPPORT VECTOR CLASSIFIER

Support Vector Classifier (SVC) is a machine learning model that aims to find the optimal hyperplane that separates the data points of different classes with the maximum margin. The hyperplane is defined by a linear combination of the features, such as

$$w^\top x + b = 0 \tag{1}$$

where w is the weight vector, b is the bias. The data points that lie on the margin are called support vectors, and they determine the optimal hyperplane. SVC can also handle non-linearly separable data by using kernel functions, such as polynomial, radial basis function (RBF), or sigmoid, to map the data to a higher-dimensional space where a linear hyperplane can be found. SVC is widely used for classification problems such as face detection, text categorization, and image recognition (Jain et al., 2000).

LOGISTIC REGRESSION

Logistic Regression is a machine learning model that predicts the probability of a binary outcome (such as yes/no, true/false, or 0/1) based on one or more predictor variables (also known as independent variables, features, or predictors). Logistic Regression uses a logistic function to model the relationship between the predictor variables and the binary outcome. The logistic function is defined as

$$\frac{1}{1 + e^{-z}} \tag{2}$$

where z is a linear combination of the predictor variables, such as

$$z = w^\top x + b \tag{3}$$

. The logistic function produces a probability score between 0 and 1, which can then be converted to a binary prediction by using a threshold value. Logistic Regression is often used for binary classification problems such as spam detection, credit scoring, and medical diagnosis (Musa, 2013).

Decision Tree

Decision Tree is a machine learning model that splits the data into smaller and smaller subsets based on a series of questions or rules, until the subsets are homogeneous or pure enough to make a prediction. Each question or rule corresponds to a node in the tree, and each subset corresponds to a branch or a leaf. The root node is the first question or rule that applies to the entire data, and the leaf nodes are the final predictions for each subset. Decision Tree can handle both categorical and numerical data, and can perform both classification and regression tasks (Hammann & Drewe, 2012).

Random Forest

Random Forest is a machine learning model that combines multiple decision trees to create an ensemble that is more accurate and robust than a single decision tree. Random Forest works by randomly selecting a subset of features and a subset of data points (also known as bootstrapping) to build each decision tree, and then aggregating the predictions of all the trees by using majority voting (for classification) or averaging (for regression). Random Forest can handle high-dimensional data, missing values, outliers, and non-linear relationships. Random Forest can also provide feature importance analysis, which measures how much each feature contributes to the prediction. Random Forest is widely used for classification and regression problems such as fraud detection, customer segmentation, and stock price prediction (Liu et al., 2015).

Reverse Fourier Transformation

Fourier transformation is a widely used method in signal processing. It can transform a kernel function to a frequency domain. The kernel function is defined as

$$K(x, x') = \phi(x)^\top \phi(x') \tag{4}$$

where ϕ is a mapping function. The Fourier transformation of K is defined as

$$\hat{K}(\omega) = \int_{-\infty}^{\infty} K(x, x') e^{-i\omega(x-x')} dx \tag{5}$$

where ω is the frequency. The inverse Fourier transformation of $\hat{K}$ is defined as

$$K(x, x') = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{K}(\omega) e^{i\omega(x-x')} d\omega \tag{6}$$

3 Experiments

3.1 Experiments Setup

Datasets. We use two datasets for our experiments: trec06c and trec06p (Voorhees & Buckland, 2006). The trec06c corpus contains mostly Chinese emails, while the trec06p corpus contains mostly English emails. Each corpus has about 37,000 emails, with about 25% of them being spam. The corpora also provide labels for each email, indicating whether it is spam or not, as well as the source of the email (such as a mailing list or a personal account). We selected 4800 emails from each corpus for training, and 1200 emails from each corpus for testing. We also created a customized test set of 9 emails, which are not included in the training or testing set, to test the models on out-of-distribution data.

Preprocessing. We preprocess the emails by performing tokenization on the email body and the email subject. We use the NLTK word tokenizer to split the email body and then vectorize the tokens by using diction-based encoding.

Model Training. We train the models on the training set and evaluate them on the testing set. We use the accuracy score to measure the performance of the models. We also record the training time of each model.

Tokenizer. We use NLTK (Loper & Bird, 2002) to tokenize the emails. NLTK provides a variety of tokenizers, including word tokenizers, sentence tokenizers, and regular expression tokenizers. We use the word tokenizer to split the emails into words, and then use the regular expression tokenizer to split the words into tokens. We also use the sentence tokenizer to split the emails into sentences, and then use the regular expression tokenizer to split the sentences into tokens. We compare the performance of the models with different tokenizers.

3.2 RESULTS

Delve into features and distribution.

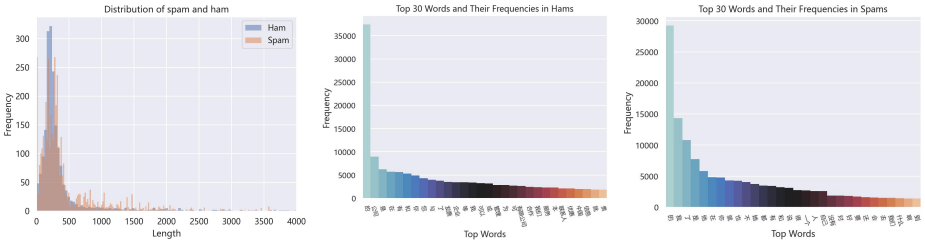

Figure 1: The first figure shows the distribution of length of spam and ham Chinese emails in trec06c. The second and third figure shows the top 30 most frequent words in spam and ham emails.

Delving into distribution of spam and ham Chinese emails can help us recognize the patterns that differentiate them. Figure 1 shows that though the distribution of length of spam and ham emails are similar, the top 30 most frequent words in spam and ham emails can be differentiated. We would like to leverage these features to train our models. The results of English emails in Figure 2 shows the same pattern.

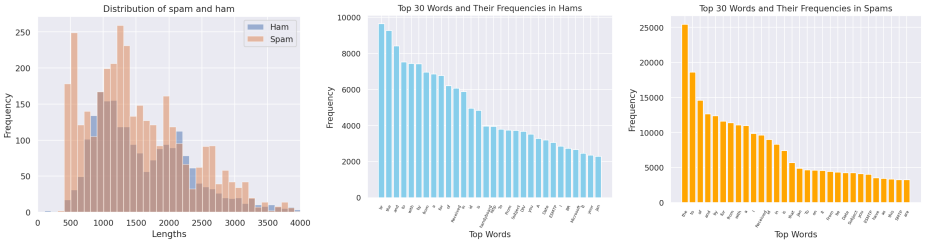

Figure 2: The first figure shows the distribution of length of spam and ham Enligsh emails in trec06p. The second and third figure shows the top 30 most frequent words in spam and ham emails.

Experiments on models

We conduct experiments on 4 models: SVC, Logistic Regression, Decision Tree and Random Forest. The results using full 6000 emails for training and test are shown in Table 1 and Table 2. SVC and Logistic Regression have a similar performance: They both have a high accuracy and a short training time. Decision Tree has a slow training time on trec06c and a low accuracy on trec06p. Random Forest has a middle accuracy and a long training time.

For further research on how our models perform like this, we plot the confusion matrix of the 4 models on the trec06c and trec06p dataset. The results are shown in Figure 3 and Figure 4. We can see that on Chinese dataset trec06c, Decision Tree tends to categorize spam emails as regular emails. On English dataset trec06p, Random Forest tends to categorize regular emails as spam emails. This

Model	Train Accuracy (%)	Test Accruracy (%)	Time Cost (s)
SVC	99.8	97.4	10.8
Random Forest	99.8	97.7	33.0
Logistic Regression	99.7	97.8	13.5
Decision Tree	99.8	95.3	33.4

Table 1: Performance of different models on the Chinese dataset trec06c. The models are trained and tested on separate set from trec06c. The resuls shows that the 4 methods has a good performance on the in-distribution dataset.

Model	Train Accuracu (%)	Test Accuracy (%)	Total Time Cost (s)
SVC	100.0	94.8	65.0
Random Forest	100.0	91.0	128.5
Logistic Regression	99.0	93.5	94.7
Decision Tree	100.0	89.5	57.1

Table 2: Performance of different models on the English dataset trec06p. The models are trained and tested on separate set from trec06p. The resuls shows that the 4 methods has a good performance on the in-distribution dataset.

shows that the 4 models we adopt do have their own advantages and disadvantages when meeting out-of-the-distribution dataset, which shows a pattern of overfitting.

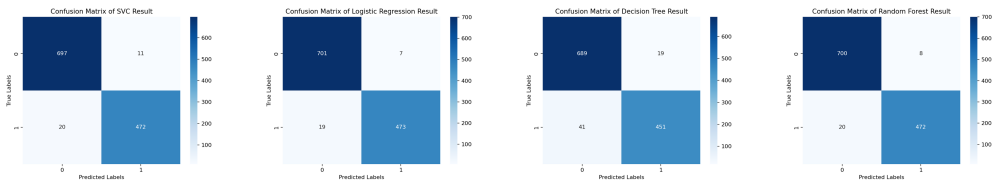

Figure 3: The confusion matrix of SVC, Logistic Regression, Decision Tree and Random Forest on Chinese dataset trec06c.

We also test the 4 models on 9 customized emails. The results are shown in Table 5. The ✓ refers to normal email, and the ✗ refers to email that is recognized as spam. We can see that the 4 models have different performance on the customized test set, which is consistent with the results on the in-distribution dataset.

Experiments on data size. We conduct experiments on SVC with different training set size. The results are shown in Table 3.

Applying reverse fourier transformation.

Also we apply reverse fourier transformation to SVC and try to improve its efficiency. This is a novel attempt to use a low parametric approximation to speed up the inference process. The results are shown in Table 4.

4 Conclusion

In this paper, we implement several machine learning methods, including SVC, Logistic Regression, Decision Tree and Random Forest for monolingual spam email detection. We test model performance on several datasets including English and Chinese emails. We conduct experiments delving into factors like language, data size, distribution, model type and more. We apply reverse fourier transformation to SVC and try to improve its efficiency. This is a novel attempt to use a low parametric approximation to speed up the training process.

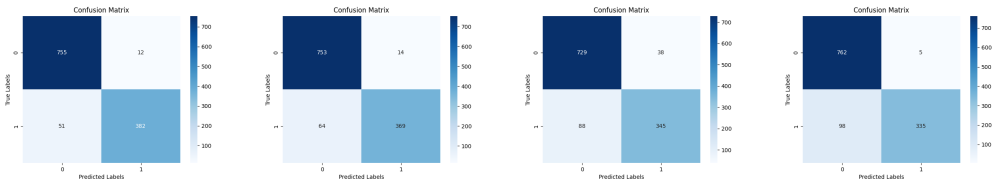

Figure 4: The confusion matrix of SVC, Logistic Regression, Decision Tree and Random Forest on English dataset trec06p.

Data size	6000	9000	12000	15000	18000
Train acc	1	1	0.9998	0.9999	0.9998
Test acc	0.9475	0.9189	0.9525	0.9197	0.9217
Training time	65.00	127.70	198.11	281.57	375.56

Table 3: SVC with different training set size. The trend shows that with a larger training set, the Testing accuracy will increase so that the model's generalization ability will be better. However, the training time will also increase.

5 Future Work

In the future, we would like to conduct experiments on multilingual spam email detection. We would also like to explore other machine learning models, such as Naive Bayes, K-Nearest Neighbors, and Gradient Boosting. We would also like to explore other tokenizers, such as the Stanford tokenizer and the spaCy tokenizer. We would also like to explore other feature extraction methods, such as TF-IDF and word embeddings. We would also like to explore other datasets, such as the Enron-Spam dataset and the SpamAssassin dataset. We would also like to explore other methods for improving the efficiency of SVC, such as using a smaller training set or using a smaller dimension for the reverse fourier transformation.

References

Andronicus A Akinyelu. Advances in spam detection for email spam, web spam, social network spam, and review spam: Ml-based and nature-inspired-based techniques. *Journal of Computer Security*, 29(5):473–529, 2021.

Felix Hammann and Juergen Drewe. Decision tree models for data mining in hit discovery. *Expert opinion on drug discovery*, 7(4):341–352, 2012.

Anil K Jain, Robert P. W. Duin, and Jianchang Mao. Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, 22(1):4–37, 2000.

Francisco Jáñez-Martino, Rocío Alaiz-Rodríguez, Víctor González-Castro, Eduardo Fidalgo, and Enrique Alegre. A review of spam email detection: analysis of spammer strategies and the dataset shift problem. *Artificial Intelligence Review*, 56(2):1145–1173, 2023.

Chengwei Liu, Yixiang Chan, Syed Hasnain Alam Kazmi, and Hao Fu. Financial fraud detection model: Based on random forest. *International journal of economics and finance*, 7(7), 2015.

Edward Loper and Steven Bird. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*, 2002.

Abdallah Bashir Musa. Comparative study on classification performance between support vector machine and logistic regression. *International Journal of Machine Learning and Cybernetics*, 4: 13–24, 2013.

Pooja Revar, Arpita Shah, Jitali Patel, and Pimal Khanpara. A review on different types of spam filtering techniques. *International Journal of Advanced Research in Computer Science*, 8(5), 2017.

Model	Training Time (s)	Inference Time (s)	Test Accuracy (%)
SVC rff k-dim=512	293.5	0.07	87.1
SVC rff k-dim=1024	87.1	0.12	87.6
SVC rff k-dim=2048	75.5	0.12	91.0
SVC rff k-dim=4096	98.6	0.18	92.5
SVC rff k-dim=8192	149.4	0.23	93.9
SVC rff k-dim=16384	217.8	0.43	93.5

Table 4: SVC with different reverse fourier transformation dimension. The trend shows that with a larger dimension, the inference time will increase so that the model's efficiency will be worse. However, the Accruracy will increase so that the reverse fourier transformation approximate the kernel function better.

Ellen M Voorhees and LP Buckland. Trec 2006. In *The Fifteenth Text REtrieval Conference, Gaithersburg, MD, USA, NIST*, 2006.

A Appendix

Model	SVC	Logistic Regression	Decision Tree	Random Forest
BB submission notification	✗	✗	✗	✓
Cloud class selection	✗	✗	✗	✗
CET 4 & CET 6 notification	✓	✗	✗	✓
SUSTech global	✓	✗	✓	✓
Volunteer recruitment	✓	✓	✓	✓
Credit certification of STAT&DS	✗	✗	✗	✗
Consultation of course selection	✗	✗	✗	✓
Chair advertisement	✓	✓	✗	✗
Youth learning from Xi Jinping	✓	✗	✓	✓

Table 5: Result of 4 models on 9 costomized test emails. The ✓ refers to normal email, and the ✗ refers to email that is recognized as spam. This shows that the 4 models we adopt do have their own advantages and disadvantages when meeting out-of-the-distribution dataset, which shows a pattern of overfitting.