

# A Hybrid Approach to Email Spam Detection: Random Forest and Sentiment Analysis

Shuyan Liu

School of Professional Studies

New York University

New York, New York

Email: sl10158@nyu.edu

**Abstract**—Email spam detection is a persistent problem in online communication, and traditional detection methods struggle to keep up with evolving spam tactics. This paper proposes a hybrid approach to the detection of email spam, combining the strengths of random forest classification and sentiment analysis. Our approach leverages the robust feature selection and classification capabilities of random forest to identify spam patterns, while incorporating sentiment analysis to capture the nuances of language used in spam emails. Our hybrid approach achieved an accuracy of 94.54%, outperforming the benchmark model Naïve Bayes. Our results show that integrating sentiment analysis with random forest classification can effectively combat email spam, making it a powerful tool for spam detection in the modern email communication environment.

## I. INTRODUCTION

In the realm of digital communication, email remains a crucial medium, facilitating personal, academic, and professional exchanges throughout the world. However, this ubiquity also makes email a prime target for spam, which not only disrupts communication, but also poses security risks. Traditional spam detection methods, such as rule-based filtering and Naïve Bayes classifiers, have been widely employed, but often fall short in dealing with the sophisticated and ever-evolving strategies employed by spammers. Consequently, there is a pressing need for more advanced and adaptive spam detection techniques.

While the Naïve Bayes algorithm has shown efficacy in spam detection, achieving accuracy rates such as 91.13% and 82.54% in studies by Rusland et al. [1] and Zhang et al. [2], its performance can be limited by its simplicity. Naïve Bayes assumes feature independence and often struggles with spam emails that cleverly blend legitimate content with deceptive cues. This simplicity, while beneficial for straightforward applications, can hinder its ability to adapt to sophisticated spamming techniques that exploit correlations between features.

This paper introduces a hybrid approach to enhance the accuracy and efficiency of email spam detection. By integrating the robust classification capabilities of Random Forest with the nuanced language understanding afforded by sentiment analysis, this method aims to identify and filter spam emails more effectively. Unlike Naïve Bayes, Random Forest does not assume feature independence and can handle the interplay between variables more effectively. It constructs multiple deci-

sion trees on various subsets of the dataset and aggregates their predictions, leading to more robust and less biased results. Sentiment analysis, on the other hand, provides insights into the emotional and psychological constructs within the text, which are often manipulated by spam content to mislead or entice the recipient.

Our hybrid model not only addresses the limitations of traditional algorithms, but also sets a new standard in spam detection by achieving an accuracy of 94.54%, thus significantly outperforming the benchmark model Naïve Bayes. This paper details the development, implementation, and evaluation of our approach, highlighting its potential as a powerful tool against the persistent challenge of email spam in modern communication environments.

## II. PROPOSED MACHINE LEARNING MODELS

### A. Naïve Bayes Classifier

The Naïve Bayes classifier is a probabilistic machine learning model that is widely used for email spam detection due to its simplicity and effectiveness. It is based on Bayes' Theorem, which uses the probabilities of events to make predictions. This model assumes that the presence (or absence) of a particular feature in a class is not related to the presence (or absence) of any other feature, known as conditional independence. Despite this simplification, Naïve Bayes can perform remarkably well and is especially fast for the training and prediction phases. It is known to outperform even highly sophisticated classification methods. T. M. Ma, K. YAMAMORI and A. Thida [3] The mathematical expression can be formulated as following:

$$P(c | x) = \frac{P(x | c)P(c)}{P(x)} \quad (1)$$

where,

- $P(c | x)$  is the posterior probability of class  $c$  given predictor(s)  $x$ .
- $P(x | c)$  is the likelihood, which is the probability of observing the predictor(s)  $x$  given that the class is  $c$ .
- $P(c)$  is the prior probability of class  $c$ , indicating how frequent the class  $c$  is in the data set before observing  $x$ .
- $P(x)$  is the marginal likelihood or evidence, which is the probability of observing predictor(s)  $x$  across all classes.

We employ the Naïve Bayes model against which we compare the performance of our proposed hybrid spam detection approach.

### B. Random Forest Classifier

Random Forest (RF) is an ensemble learning method, renowned for its high accuracy and robustness. Developed by L. Breiman [4], this algorithm improves decision making by combining the predictions of multiple decision trees into a final output. Each tree in the ensemble is constructed from a distinct bootstrap sample of the data. During the construction of these trees, the nodes are split using the optimal split selected from a randomly chosen subset of the features. This method not only leverages the strength of multiple learning models, but also introduces randomness into the model selection process, significantly reducing the risk of overfitting. Its speed and efficiency when applied on large datasets, it doesn't overfit, no presumptions on the distribution of the data are needed. L. Guo et al. [5]

## III. METHODOLOGY

1) *Dataset*: The empirical evaluation of our spam detection model uses three distinct data sets to ensure a comprehensive assessment across various types of emails. The TREC 2006 Spam Data comprise 53,668 emails, split between 29,923 ham and 23,745 spam messages, providing a substantial volume for both training and testing our algorithms. Following this, the TREC 2007 Spam Data includes 17,309 emails, with a division of 12,508 ham and 4,801 spam messages. Lastly, we include a Spam Email Dataset from Kaggle, which consists of 5,127 emails, closely balanced with 2,259 ham and 2,868 spam messages. This balanced dataset is critical for testing the model's effectiveness in scenarios where spam and ham are nearly equally represented. The varied nature and complexity of these datasets are instrumental in evaluating the generalizability and reliability of the proposed spam detection system.

2) *Data Preprocessing*: Effective data preprocessing is crucial for minimizing noise and enhancing the performance of the model. Our preprocessing pipeline incorporates several steps to standardize and refine the input data, ensuring it is optimally prepared for the subsequent modeling phases. These steps include:

*Lowercasing*: All text data is converted to lowercase to maintain consistency across the dataset, eliminating variations caused by case differences.

*Removal of Redundant Prefixes*: We remove specific prefixes, such as 'subject:', which could introduce bias if left within the text, as these elements do not contribute to distinguishing between spam and legitimate emails.

*Numeric Replacement*: Numeric values within the texts are replaced with a placeholder token. This approach prevents numbers from skewing the model's learning process, as their presence varies widely across emails.

*Removal of Stopwords*: Stopwords, which are commonly occurring words that offer little analytical value, are removed to sharpen the focus on more meaningful terms. This step is

critical in spam detection as it reduces the data dimensionality and emphasizes keywords that are more likely to be indicative of spam or non-spam emails.

*Tokenization*: Text data is broken down into individual words or tokens.

*Lemmatization*: Words are reduced to their base or dictionary forms using lemmatization. This process helps in consolidating different forms of the same word, ensuring that variations in tense or plurality do not affect the analysis.

Each of these preprocessing steps is designed to refine the dataset, reducing redundancy and emphasizing features that are most informative for spam detection. This careful preparation is essential for building a robust model capable of effectively identifying spam emails.

3) *Feature Extraction*: *TF-IDF Transformation*: We employed Term Frequency-Inverse Document Frequency (TF-IDF) to transform the text data into a numerical format that is more amenable to machine learning algorithms. As noted by J. Ramos [6] in his 2003 study, TF-IDF weighs the words' frequencies across documents against their distribution across the entire corpus, highlighting words that are particularly pertinent to individual documents more than those common throughout. This method enhances the model's ability to discern and learn from the patterns in the text data.

4) *Sentiment Analysis*: Sentiment analysis plays a pivotal role in our spam detection system, significantly enhancing its capability to differentiate between legitimate emails and spam. This technique leverages natural language processing (NLP) to assess the emotional content of email messages. Spam emails often employ distinct emotional cues, such as urgency or exaggerated positivity, to manipulate the recipient. These cues serve as significant indicators for distinguishing spam from legitimate emails. To quantify these emotional tones, we utilize the TextBlob library, which assigns polarity scores to the text. These scores range from  $-1.0$ , indicating a very negative sentiment, to  $1.0$ , reflecting a very positive sentiment. For integration with our Naïve Bayes model, we adjust these scores from the original range of  $[-1, 1]$  to  $[0, 2]$ . This adjustment better aligns with the model's requirements for handling input values, ensuring that the sentiment scores are effectively utilized to enhance spam detection accuracy.

5) *Experimental Setup and Model Comparison*: To rigorously evaluate the efficacy of our spam detection system, we will conduct a series of experiments comparing four different models the model building process is shown in Fig. 1.

- Naïve Bayes (NB): This baseline model uses standard spam detection techniques without sentiment analysis, relying primarily on TF-IDF scores.
- Random Forest (RF): As another baseline, this model applies a Random Forest algorithm, using the same TF-IDF features.
- Naïve Bayes with Sentiment Analysis (NB-SA): This model enhances the Naïve Bayes approach by incorporating sentiment analysis, adding sentiment scores as additional features to the model to see if recognizing emotional cues improves spam detection.

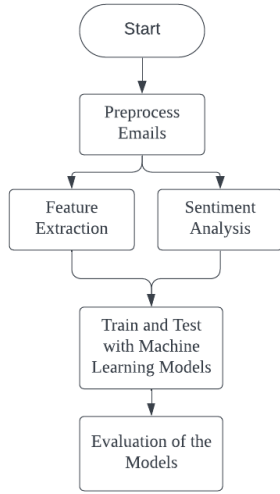


Fig. 1. Model Build Flow Chart

- Random Forest with Sentiment Analysis (RF-SA): This model combines Random Forest with sentiment analysis. We hypothesize that this model will perform the best due to its ability to utilize both the structured decision-making of Random Forest and the nuanced understanding of text sentiment.

Each model will be assessed using standard performance metrics in machine learning. A confusion matrix will serve as the basis for evaluating each proposed approach in terms of false positive, false negative, accuracy, precision, and recall.

#### IV. RESULTS AND DISCUSSION

TABLE 1 presents detailed results of our proposed models, demonstrating that the baseline Random Forest model (RF) significantly outperforms the Naive Bayes model (NB). The NB model displays a notably high False Positive Rate (FPR) of 38.91%, in contrast to the RF's 10.67%. In applications such as email spam detection, such a high FPR can lead to considerable user inconvenience by misclassifying legitimate emails as spam. Moreover, the RF model exhibits superior accuracy, achieving 87.32% compared to 66.67% for the NB model.

Integration of sentiment analysis markedly improves both models, with the Random Forest with Sentiment Analysis model (RF-SA) showing exceptional performance improvements, reducing the False Negative Rate to 3.69% and boosting accuracy to 94.54%. These enhancements suggest that while sentiment analysis can elevate the performance of basic models, the selection of the underlying algorithm is pivotal in maximizing the utility of advanced feature processing techniques.

The improvement from NB to NB-SA and RF to RF-SA highlights the significant role of sentiment analysis in capturing nuanced data details, thereby enhancing predictive accuracy.

TABLE I  
CLASSIFICATION RESULT FOR PROPOSED MODELS

Model	FP-Rate	FN-Rate	Accuracy	Precision	Recall
NB	26.25%	38.91%	66.67%	68.10%	66.67%
RF	15.23%	10.67%	87.32%	87.31%	87.32%
NB-SA	23.55%	21.24%	77.80%	78.10%	77.80%
RF-SA	7.59%	3.69%	94.54%	94.56%	94.54%

The Receiver Operating Characteristic (ROC) curves, as illustrated in Fig. 2, provide additional insights into the models' ability to discriminate between classes, affirming the superior discriminative power of the RF-SA model.

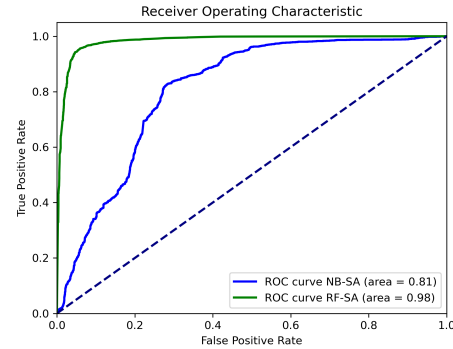


Fig. 2. ROC Curves for NB-SA and RF-SA Models

#### V. CONCLUSION

This study validates the effectiveness of a hybrid email spam detection approach that integrates Random Forest classification with sentiment analysis. This combination leverages the powerful feature selection capabilities of Random Forest and the nuanced detection offered by sentiment analysis to effectively combat the sophisticated tactics of modern spam emails. Our findings indicate that this method excels at identifying subtle linguistic indicators typical of advanced spam campaigns, thus minimizing false positives and enhancing the reliability of the detection system.

Future research should focus on refining the integration of sentiment analysis within the classification process, exploring the potential of more granular linguistic features such as syntax and contextual semantics to further enhance the model's accuracy. Additionally, experimenting with deep learning techniques, particularly those specializing in natural language processing, could offer new insights and improvements in spam detection capabilities.

Subsequent studies should also assess the scalability and efficiency of this hybrid model across larger and more diverse datasets, as well as under varying operational conditions, to confirm its effectiveness in a real-world environment. Such evaluations would provide deeper insights into the model's practical applications and limitations, facilitating its broader adoption and implementation in spam detection systems.

## ACKNOWLEDGMENT

I would like to express my sincere gratitude to Professor Eleftheria K. Pissadaki and Professor Thomas M. Schmidt for their invaluable guidance and mentorship throughout this research. Their insights and expertise have been crucial in shaping both the direction and the execution of this study. I am deeply thankful for their support and encouragement, which have significantly enriched my academic journey.

## REFERENCES

- [1] N. F. Rusland, N. Wahid, S. Kasim, and H. Hafit, "Analysis of Naïve Bayes algorithm for email spam filtering across multiple datasets," in *IOP Conference Series: Materials Science and Engineering*, vol. 226, no. 1, Aug. 2017, Art. no. 012091.
- [2] H. Zhang, N. Cheng, Y. Zhang, and Z. Li, "Label flipping attacks against Naïve Bayes on spam filtering systems," *Applied Intelligence*, vol. 51, no. 12, pp. 4503–4514, 2021.
- [3] T. M. Ma, K. Yamamori, and A. Thida, "A Comparative Approach to Naïve Bayes Classifier and Support Vector Machine for Email Spam Classification," in *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, Kobe, Japan, 2020, pp. 324–326, doi: 10.1109/GCCE50665.2020.9291921.
- [4] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.
- [5] L. Guo, N. Chehata, C. Mallet, and S. Boukir, "Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 1, pp. 56–66, 2011, Elsevier.
- [6] J. Ramos, "Using tf-idf to Determine Word Relevance in Document Queries," in *Proceedings of the First Instructional Conference on Machine Learning*, vol. 242, no. 1, pp. 29–48, Dec. 2003.