International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Optimizing Email Spam Detection and Natural Language Processing with Keyword Driven Multi-Class Labeling

Nilambari A. Jagtap[1], Om S. Patil[2], Sajjan R. Ahiwale[3], Ajay A. Mane[4]

[1,2,3,4]Department of Statistics, Tuljaram Chaturchand College of Arts, Commerce and Science, Baramati, Maharashtra, India,

Abstract

Email has become a key part of our daily lives, both at work and at home. It helps us communicate and share information quickly and easily. But with the rise of SPAM emails, there are growing problems like security risks, overloaded inboxes, and missed important messages. SPAM emails often contain scams, viruses, or unwanted ads, which can be dangerous for individuals and businesses. this project explores ways to improve SPAM detection using machine learning and natural language processing (NLP) to better understand email content and reduce errors. We also created a simple keyword-based system that sorts emails into five categories Spam, Promotion, Social, Updates, and Primary. While the keyword system helps manage emails quickly, it can misclassify some messages. the study shows how combining machine learning with NLP can make email filtering smarter and more reliable.

Keywords: Email SPAM detection, Natural Language Processing (NLP), Machine Learning Algorithms, Data Mining, Deep learning.

1. Introduction:

Email is a vital tool for communication in our personal and professional lives. the growing volume of spam unwanted or harmful emails makes managing our inboxes challenging and risky. While spam filters help by automatically sorting or blocking such emails, they are not perfect. A major problem is false positives, where important messages are wrongly marked as spam and lost. This can lead to missed opportunities or security issues.

To address these challenges, this study explores how Natural Language Processing (NLP) combined with machine learning models can improve spam detection. By analysing the actual content and context of emails, NLP helps distinguish spam from legitimate messages more accurately. We evaluate various models including Random Forest, SVM, KNN, Naïve Bayes, Logistic Regression, Decision Tree, and AdaBoost, using metrics like accuracy, precision, recall, and F1 score. To create a more intelligent, accurate, and reliable spam filter system that minimizes errors and enhances email security.

2. Methodology :

2.1 Data collection

The email data for this research was collected from ten individual Gmail accounts. Each user's mailbox was downloaded in MBOX format using Google Takeout. The MBOX files were parsed and converted

into Excel files using Python scripting. Each Excel file contained structured data including the date, sender, receiver, subject, and body of the emails. After processing the files individually, all Excel datasets were combined into a single master file.

The final consolidated dataset contains a total of 55,436 email records. All preprocessing steps were executed in Python to ensure consistency and automation. The data captures a wide range of real-world email content across different categories. This dataset serves as the basis for applying NLP and machine learning models for spam classification.

2.2 Data preprocessing

To prepare the email dataset for analysis, duplicates, null values, and irrelevant records were removed to ensure data quality. Text was normalized by converting it to lowercase and removing special characters. Tokenization was applied to split the text into individual words. Common stopwords like "is" and "the" were removed to highlight meaningful terms. Stemming was used to reduce words to their root forms by trimming suffixes. Lemmatization further refined this by returning proper base words using grammatical rules. These steps helped transform raw text into a clean, structured format suitable for NLP tasks.

2.3 Natural Language Processing (NLP) technique :

Natural Language Processing (NLP) techniques were employed to analyze the email content by first applying tokenization to break down text into individual words, followed by the removal of stopwords to retain meaningful terms, and then performing word normalization through stemming and lemmatization to reduce words to their root or dictionary forms; in addition, feature engineering was conducted by creating structural features such as subject length, word count, link and special character frequency, extracting date-based attributes like month and weekday, identifying keyword-based categories (spam, promotion, social, update, or primary), and converting the data into a machine-readable format using label encoding for output classes and vectorization techniques like CountVectorizer and TF-IDF to represent textual data numerically for model training.

3. Exploratory data analysis

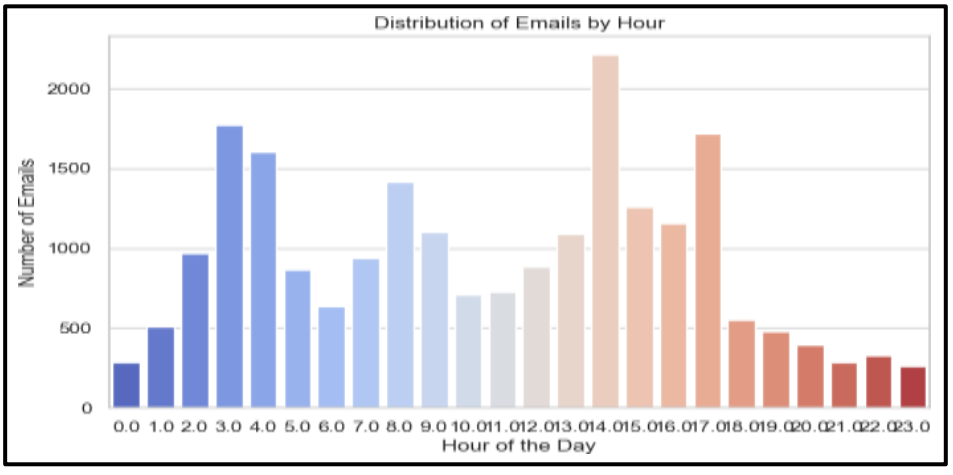

Figure 1

Figure 1 shows email activity peaks at 2:00 PM during working hours, with a secondary spike around 3:00 to 4:00 AM likely due to automation, reflecting both human and scheduled email behavior throughout the day.

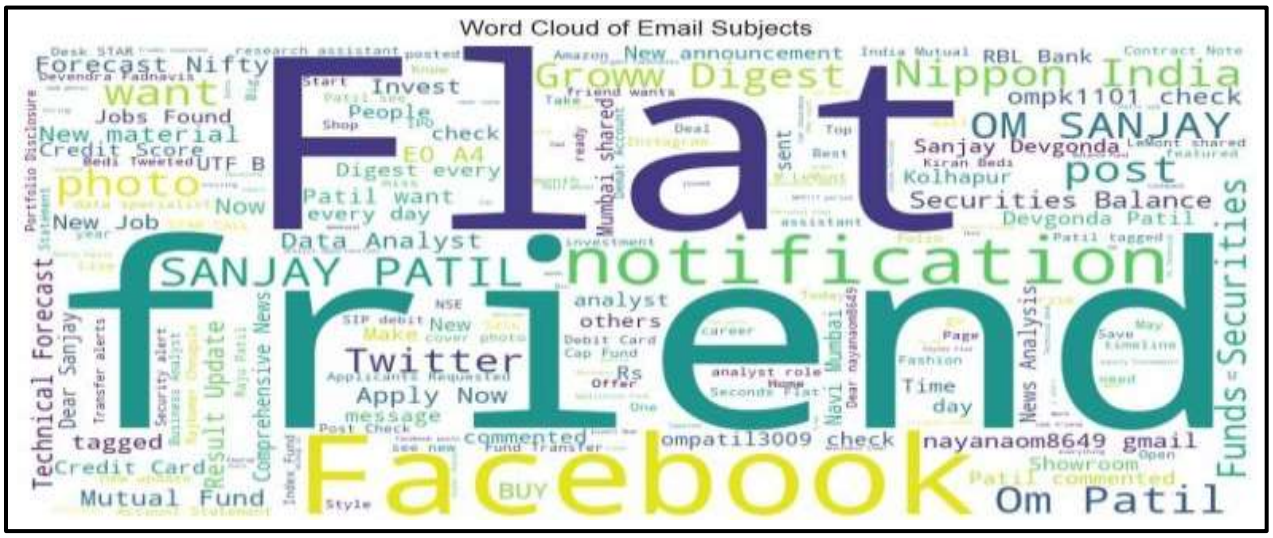

Figure 2

Figure 2 shows word cloud reveals frequent terms in subject lines, highlighting social notifications and professional or financial topics indicating diverse email categories.

4. Predictive statistical model:

4.1 Splitting Trian & Test Set:

The dataset was split into 70% training and 30% testing sets. Machine Learning models were then applied to learn patterns from the training data and make predictions on unseen emails.

4.2 Model building

Random Forest achieved the highest accuracy with balanced precision and recall, making it the best-performing model. KNN and SVM performed well in detecting Spam and Updates but had some weaknesses in recall and specific categories. Naive Bayes showed the lowest performance, while Logistic Regression and Decision Tree struggled mainly with the Social email category.

Table 1

Algoritham	Primary Precision (%)	Promotion Precision (%)	Social Precision (%)	Spam Precision (%)	Updates Precision (%)	Accuracy (%)	Precision (%)	Recall (%)
Naïve Bayes	62	74	65	62	69	63	66	43
SVM	74	84	76	96	92	87	84	45
KNN	81	80	72	98	89	91	84	63
Random Forest	83	85	88	97	96	92	90	64
Multiple Logistic Regression	78	79	55	97	61	89	74	52
Decision Tree	79	79	78	98	90	90	85	65
AdaBoost	82	80	51	97	91	90	80	60

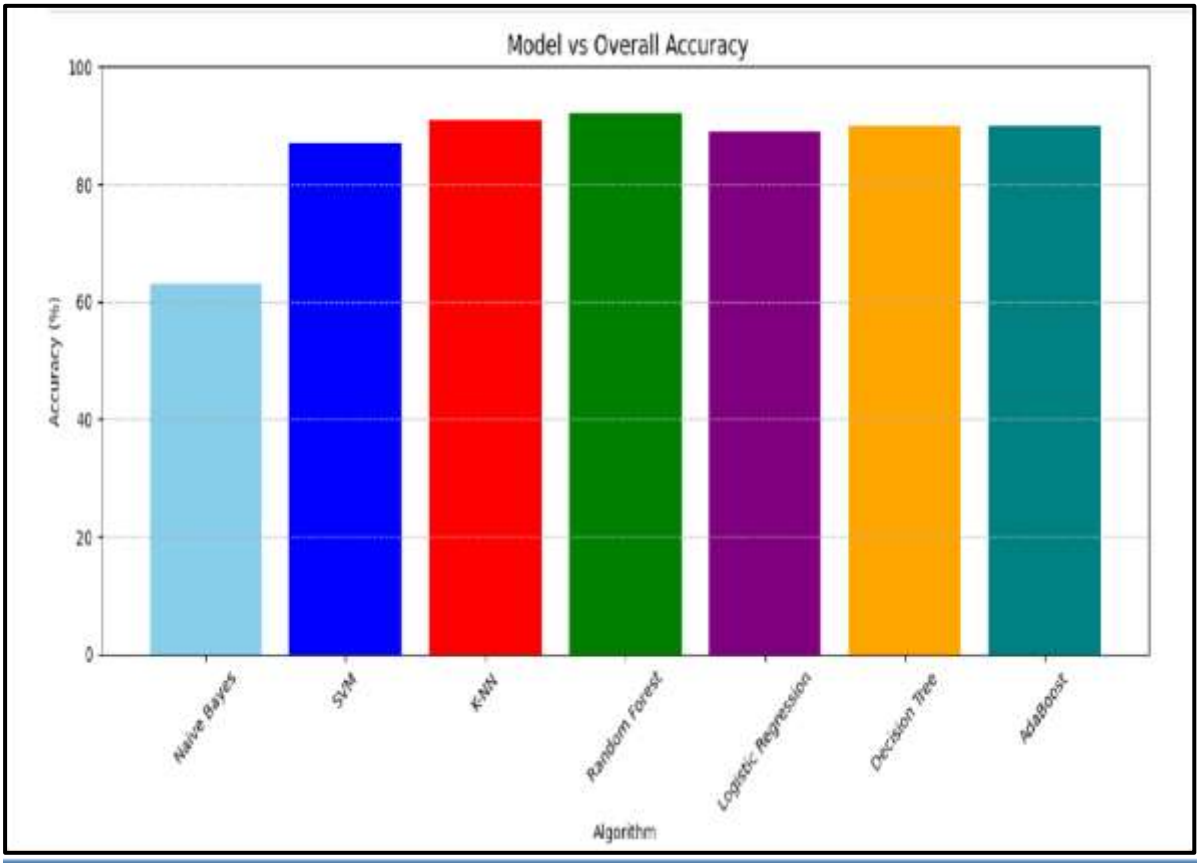

Figure 3

Figure 3 shows random Forest emerged as the most accurate and reliable model for email classification, while Naive Bayes showed the lowest performance with the highest classification errors.

5. Conclusion

This research concludes that combining NLP techniques with machine learning significantly improves email classification accuracy. Among the models tested, Random Forest proved to be the most reliable, achieving 92% accuracy and excelling across all email categories. KNN, SVM, and AdaBoost also performed well in spam and updates classification but struggled with overlapping or underrepresented classes like Social. Naive Bayes was the fastest but had the lowest accuracy, while Multiple Logistic Regression and Decision Tree offered solid results with specific limitations. The integration of keyword based classification and advanced feature engineering further enhanced performance, making the overall system both efficient and effective for real world spam detection and email organization.

References

1. Subhashini, G., & Mahalakshmi, G. (2024). Advanced SMS Spam Detection Using Integrated Feature Extraction. 4th International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE.
2. Alharbi, F., & Lee, S. (2020). Adaptive spam detection framework using NLP and classification-based techniques. Journal of Information Security and Applications, 54, 102581.
3. Zhang, Y., Jin, R., & Zhou, Z. (2017). Understanding NLP-based spam detection using keyword and semantic cues. Knowledge-Based Systems, 126, 70–84.