# SMS Spam Detection Using Machine Learning

Hanamant Klayani[1], Clitus Neil D souza[2], Nandkumar[3], Nitish Patil[4], Shreesha K R[5]
[1,3,4,5]*Department of Electronics and Communication Engineering, Srinivas Institute of Technology, Mangaluru 574143, Karnataka, India*
[2]*Assistant Professor, Department of Electronics and Communication Engineering, Srinivas Institute of Technology, Mangaluru – 574 143, Karnataka, India*
doi.org/10.64643/IJIRTV12I12-206815-459

***Abstract*—The problem of spam SMS messages has been a significant issue to mobile users which has caused privacy, and fraud as well as unwanted nuisance. In order to solve this problem, the project primarily aims to create a system that can automatically classify and detect irrespective of SMS messages as spam or legitimate messages through machine learning techniques. The dataset (Ham + Spam) has been cleaned, the stop words have been removed, messages have been lemmatized and the TF-IDF vectorizer has transformed the messages into numbers to make them SMS ready. Multiple Machine learning models like Logistic Regression Classifier, Naïve Bayes Classifier, Decision Tree Classifier, Random Forest, Linear SVM, KNN and a Voting Ensemble Classifier will be trained & analysed. The performance of each model is assessed using various parameters such as accuracy, precision, recall, F1 score, confusion matrix, and ROC curve analysis. Linear SVM was the most successful model in SMS spam classification in the project as it was the most accurate of algorithms. The results show that ML-based approaches algorithm can be efficiency used to identify spam messages and assist users to increase their safety, avoid fraud, and improve the security of mobile communication.**

## I. INTRODUCTION

SMS is a very popular form of communication and this has resulted in a surge in spam messages, scams, phishing and fraudulent offers. Spammers are continually changing message patterns to get past traditional rule-based filtering systems. This is the issue that is tackled in the present project by employing Machine Learning techniques to automatically classify SMS as Spam or Ham. It includes the following steps: Text Preprocessing, Transformation of Text Data into Numerical Data using TF-IDF, Training of Multiple Classifiers including LR, NB, DT, RF, KNN and Linear SVM. Evaluation of the models is performed by using accuracy, precision, recall and F1 score, where the linear SVM is the best model. The idea is to build a scalable, efficient, accurate system that can detect spam emails in real time and adapt to the changing spam patterns.

## II. METHODOLOGY

### A. Dataset Dscription and Preprocessing

The data is from Kaggle and has 5,574 SMS messages - of which 4,827 are ham and 747 are spam - with 2 main columns: Label and SMS Text. Errors, missing data or irrelevant columns are removed as data is first cleaned to ensure accuracy and reliability. The text is then preprocessed by removing the stop words, lemmatizing the text, and then converting it to numerical features using TF-IDF. The dataset is divided into 80% training data (4,457 messages) and 20% testing data (1,115 messages) with 80% used to train the machine learning models, and 20% used to test the models for performance. Different models like Naive Bayes, SVM and Random Forest are trained and tested on this data, with the performance of the models being assessed through evaluation metrics. Finally, the system enables users to enter an SMS message, which is then categorized as spam or ham.

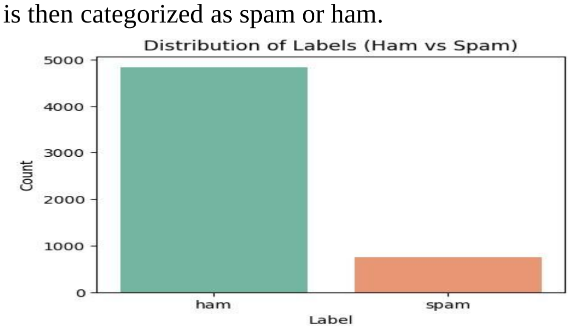


Fig 1: Distribution of Labels

B. Model Evaluation Metrics

Accuracy:

Measures the proportion of total predictions (spam and ham) that are correctly classified.

Formula:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision:

Indicates how many of the messages predicted as spam are actually spam.

Formula:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall:

Indicates how many actual spam messages were correctly identified by the model.

Formula:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-Score:

Harmonic mean of Precision and Recall, providing a balance between them.

Formula:

\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}

C. Confusion Matrix

The confusion matrix is a performance evaluation tool used to measure the accuracy of a classification model in categorizing SMS messages into Spam and Ham (Not Spam). It is represented as a 2 × 2 matrix that compares the actual labels with the predicted labels, showing both correct and incorrect classifications.

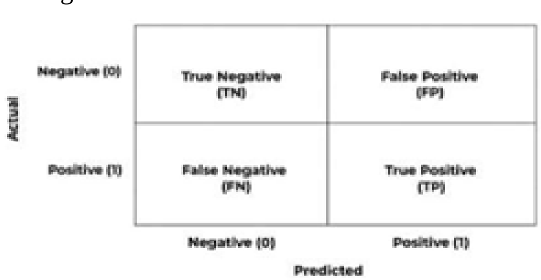


Fig 2: Confusion Matrix

III. RESULT AND DISCUSION

A. SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM) model is a very good model for detecting SMS Spam, with good accuracy of 964 ham and 134 spam messages, with very few misclassifications (only 2FP and 15FN). This is because it is based on building an optimum decision boundary, which maximizes the separation between classes. SVM can handle high dimensional and sparse data effectively, like TFIDF representation of text, and is effective at avoiding overfitting and generalizing the data. In general, SVM is shown to be a stable and efficient spam detection classifier.

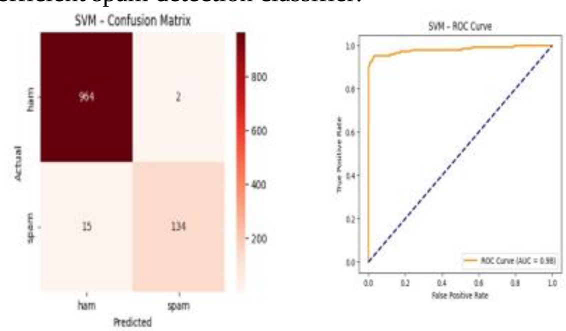


Fig 3: SVM - Confusion Matrix and Curve

B. Random Forest

The Random Forest (RF) Classifier correctly classifies 966 ham and 124 spam messages and has misclassified 25 spam messages as ham, which shows that the classifier has some false negative. It uses an ensemble of decision trees, which are combined to give out its prediction and reduces overfitting and increases the robustness over a single tree. Its spam detection performance is slightly less because of the difficulties in capturing minority class patterns, but it is still a balanced, stable and effective spam model for noisy text data.

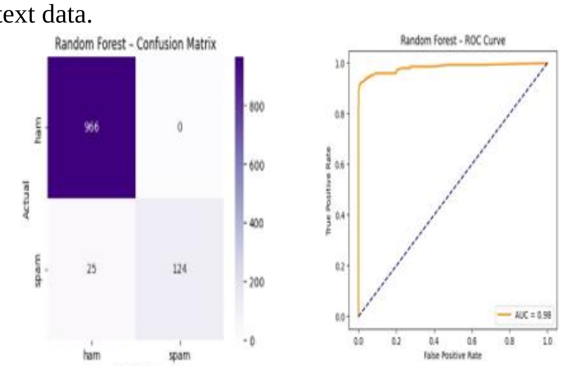


Fig 4: Random Forest - Confusion Matrix and Curve

C. Logistic Regression

Logistic Regression is a method that gives satisfactory and reliable results for both of the SMS classification types. It correctly labels 962 ham messages, and it correctly labels 110 spam messages and makes a limited number of misclassifications: 4 ham messages are labelled as spam, and 39 spam messages are labelled as ham. The Logistic Regression is a linear model that can be extremely useful when features are TF-IDF vectors converting text to vectors. It is able to solve the linearly separable data, and regularizes them to avoid overfitting. It is not as accurate as SVM but gives a stable performance, and easy to interpret.

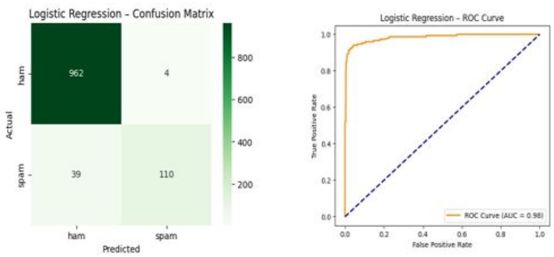


Fig 5: Logistic Regression- Confusion Matrix and Curve

D. Voting Ensemble

To achieve strong and stable performance, the prediction of a variety of models is computed and a Majority Voting classifier is used. It also adequately predicts 966 ham and 112 spam messages and it incorrectly classifies less messages than the weaker individual models. The tendency of ensemble methods is to enhance reliability because they minimize the overfitting factor along with the advantages and limitations of various algorithms are balanced. In this project, the Voting Classifier is also exceptional in terms of accuracy, great generalization and low classification error. The ensemble model has consistency and stability for different data patterns, although it is slightly more successful, SVM is also. This is why it is a good option to use in the real-life spam detection system.

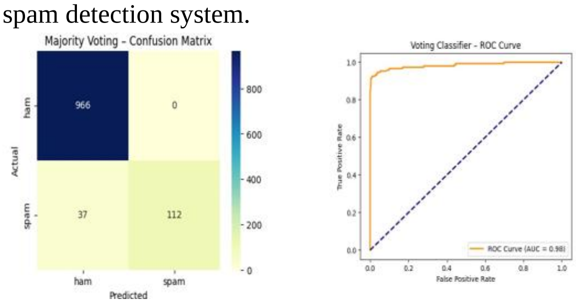


Fig 6: Voting Ensemble- Confusion Matrix and Curve

E. MODEL PERFORMANCE

Table 1: Model Performance Comparison

| MODELS | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Linear SVM | 98.48% | 0.98 | 0.98 | 0.98 |
| Voting Ensemble | 96.68% | 0.97 | 0.96 | 0.96 |
| Decision Tree | 95.61% | 0.95 | 0.94 | 0.94 |
| Random Forest | 97.76% | 0.98 | 0.97 | 0.97 |

F. DEPLOYMENT

Gradio is a Python library for developing simple and interactive web interfaces to test machine learning models. It offers a user-friendly interface including textboxes, buttons, etc. without using HTML, CSS, or JavaScript. Users can enter data, e.g. an SMS, and have the prediction of the model displayed immediately. Furthermore, Gradio allows for simple deployment of models via a web link, allowing for demonstrations and easy collaboration.

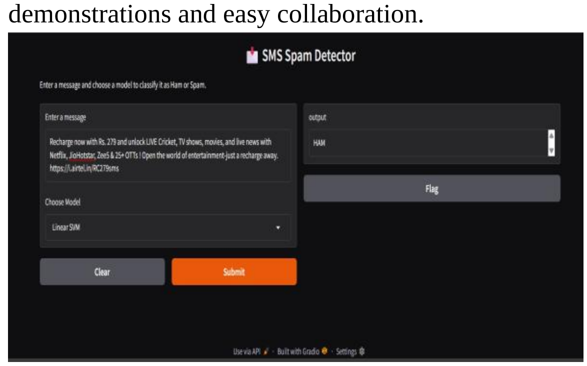


Fig 7: SMS Spam Detector Interface

## IV. CONCLUSION

This spam detection project is an example of how machine learning models could be used to accurately classify a message as Spam or Ham with a high degree of reliability. The system employs labelled. SMS data, pre-processes the data by removing noise, normalization and stop words, and then converts the text to numeric representation using TF-IDF to effectively perform pattern recognition. A number of algorithms such as Decision Tree, Logistic

Regression, Naïve Bayes, Random Forest, KNN, Linear SVM and Voting Ensemble were trained and tested and among them, Linear SVM was the most accurate and most robust and effective model for text classification. The study also emphasizes the role of the preprocessing in the improvement of the overall model performance. It can be beneficial for mitigating unwanted content such as spam, phishing, and unwanted messages, which can help users stay more private, secure, and receive better communications. Overall findings from the project indicate that the machine learning approach is efficient, effective and scalable and can be used in SMS based messaging platforms and telecommunication services. Further improvements could involve the use of deep learning models like LSTM or BERT, extending the capabilities to multiple languages, updating to counter new spam trends, and potentially rolling it out as a cloud-based API for greater accuracy and responsiveness.

## REFERENCES


[1] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. European Conference on Machine Learning, 137–142.

[2] Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. AAAI Workshop on Learning for Text Categorization, 55–62.

[3] Androutsopoulos, I., Koutsias, J., Chandrinos, K., & Spyropoulos, C. D. (2000). An evaluation of Naive Bayesian anti-spam filtering. Proceedings of the Workshop on Machine Learning in the New Information Age, 9–17.

[4] Ramos, J. (2003). Using TF-IDF to determine word relevance in document queries. Proceedings of the First Instructional Conference on Machine Learning.

[5] Cormack, G. V. (2008). Email spam filtering: A systematic review. Foundations and Trends® in Information Retrieval, 1(4), 335–455.

[6] Gómez Hidalgo, J. M., Bringas, G., Sánz, E., & García, F. (2006). Content-based SMS spam filtering. Proceedings of the 2006 ACM Symposium on Document Engineering, 27–32.

[7] Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.

[8] Powers, D. M. (2011). Evaluation: From Precision, Recall and F-Measure to ROC and Informedness. Journal of Machine Learning Technologies.

[9] Almeida, T. A., Hidalgo, J. M. G., & Yamakami, A. (2011). Contributions to the study of SMS spam filtering: New collection and results. ACM Symposium on Document Engineering, 259–262.

[10] Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. Information Processing & Management, 50(1), 104–112.

[11] Saha, S., & Ghosh, S. K. (2016). SMS spam filtering using text classification techniques. International Journal of Computer Applications, 139(6), 1–6.

[12] Zhang, X., & LeCun, Y. (2015). Text understanding from scratch. arXiv preprint arXiv:1502.01710.

[13] Dada, E. G., Bassi, J. S., Chiroma, H., et al. (2019). Machine learning for spam detection: Review, trends, and future directions. Heliyon, 5(11), e02711.

[14] Minaee, S., Kalchbrenner, N., Cambria, E., et al. (2021). Deep learning-based text classification: A comprehensive review. ACM Computing Surveys, 54(3), 1–40.

[15] Sharma, A., & Kumar, R. (2022). SMS spam detection using hybrid machine learning models. International Journal of Information Technology, 14(2), 789–797