International Multidisciplinary Research Journal Reviews (IMRJR)

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

Spam Email Detection Using Machine Learning

K. Meghna[1], J. Akash[2], O. Rushikesh[3], K. Harinath[4], P.V. Ramana Murthy[5]

Student, Department of IT. Malla Reddy Engineering College, Maisammaguda, Hyderabad-500100[1-4]

Professor, Department of IT. Malla Reddy Engineering College, Maisammaguda, Hyderabad-500100[5]

Abstract: Our project focuses on a comparative analysis of spam detection models using three datasets, including two custom-built ones, to improve detection accuracy. We prepared the data using preprocessing techniques such as tokenization, stemming, and stop word removal. Various models, including RNNs, SVM, Naive Bayes, and decision trees, were trained and compared based on accuracy and precision. Our goal is to identify the most effective methodology for detecting spam emails. The results aim to enhance spam detection systems by minimizing false positives and ensuring legitimate emails reach the user. Accurate spam detection can prevent phishing, malware, and other harmful activities. Our findings can contribute to the development of more precise and efficient spam detection technologies. This study has the potential to make email communication safer and more reliable.

Keywords: Naive Bayes, spams, Logistic regression, Bag of Words, Term Frequency- Inverse Document Frequency, non-spam (ham), accuracy, precision, recall, F1-score.

I.INTRODUCTION

Our project focuses on improving spam detection using machine learning (ML) techniques, addressing the increasing threat of email spam. With over 4.5 billion internet users relying on emails as a reliable communication medium, spam has become a growing concern, often leading to phishing, malware, and data theft. Spam emails, typically sent in bulk, consume network resources, disrupt users, and pose security risks.

Spam detection aims to identify and separate unwanted emails by analyzing the content, subject, and structure. Traditional methods, such as knowledge engineering, analyze IP addresses and apply predefined rules to detect spam, but these methods are time-consuming and less adaptable. ML-based approaches, however, eliminate the need for predefined rules, making them more efficient and accurate. Natural Language Processing (NLP), a key AI technique, is used to extract relevant information from email text and identify spam characteristics. Our project compares multiple ML models such as RNNs, SVM, Naive Bayes, and decision trees to determine the most effective spam detection approach. Preprocessing techniques like tokenization, stemming, and stop word removal are applied before training models. Spam detection helps minimize false positives and ensures that legitimate emails are delivered while blocking harmful ones. The findings from our research can improve spam detection systems, reduce security threats, and enhance email communication reliability. As spammers continue to develop sophisticated techniques, efficient spam detection models become essential to safeguard users from online threats. Our study highlights the potential of advanced ML models to provide accurate and effective spam filtering, contributing to safer and more secure digital communication.

II.LITRATURE SURVEY

To analyse existing research on spam email detection and classification techniques, aiming to identify effective methods and improve current systems. Various studies have explored approaches such as content-based filtering, feature selection, and classification using machine learning models. Jiaming Yang et al. (2011) applied binomial hypothesis testing and Support Vector Machine (SVM) to identify spam. Seyed Mustafa Pour Hashemi et al. (2014) proposed a hybrid feature selection approach using Chi-Square-2 and wrapper-based techniques, while Guangxi Li (2013) introduced collaborative online multitask learning for filtering. Tom Fawcett et al. (2003) utilized vivo-based spam filtering to address class imbalance and error issues. Sim hash-based email reflection was suggested by Venkata Reddy & Ravichandra (2014) to recognize spam features effectively. Harikrishna et al. (2014) applied statistical-based features, and Tanin pong & Ngamsuriyaroj (2009) developed an incremental filtering system. Bhat et al. (2011) used the Beaks-based approach with Random Forest, and Rohan et al. (2012) employed the Random Forest method to target malicious emails. Sarju et al. (2014) used structural criteria and classifiers like AdaBoost for detection, while Jafar Alqatawna et al. (2015) focused on content-based spam detection with decision trees and neural networks. Christina et al. (2010) suggested supervised learning techniques such as C4.5 and multilayer perceptron networks. Nadir Omer FadlElssied et al. (2014) proposed a hybrid K-means and SVM approach to reduce false positives. Kumar et al. (2015) applied neural networks with feature selection using Particle Swarm Optimization (PSO), and Chih-Hung Wu et al. (2009) analysed behavioural features for classification. Overall, the survey highlights the evolution of spam detection methods and their effectiveness.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

III. METHODOLOGY

Spam email detection using Naïve Bayes and Logistic Regression involves machine learning techniques that classify emails as spam or legitimate based on learned patterns. Naïve Bayes is a probabilistic algorithm that applies Bayes' theorem to calculate the likelihood of an email being spam based on the presence of specific words or features. It assumes that words in an email appear independently, making it computationally efficient and highly effective, often achieving high accuracy rates. It is particularly useful when working with large datasets due to its fast classification speed and ability to handle missing data. However, it may struggle with sophisticated spam techniques like adversarial word manipulation. On the other hand, Logistic Regression is a linear classification algorithm that predicts the probability of an email being spam based on weighted features. It works well when features are linearly separable and provides interpretable results, but its performance may degrade if spam emails contain complex, non-linear relationships. While Naïve Bayes is more efficient for large-scale spam filtering, Logistic Regression can perform better when carefully tuned with feature selection and regularization techniques.

3.1. Data Collection

Data plays an important role when it comes to prediction and classification, the more the data the more the accuracy will be. The data used in this project is completely open-source and has been taken from various resources like Kaggle For the purpose of accuracy and diversity in data multiple datasets are taken. 1 dataset containing approximately over 5573 mails and their labels are used for training and testing the application. In total dataset spam data is 12.6 percent and ham data is 87.4 percent.

spam	Get the official ENGLAND poly ringtone or colour flag on yer mobile for tonights game! Text TONE or FLAG to 84199. Opt
ham	Hahaha..use your brain dear
ham	Jus finish watching tv... U?
ham	K, fyi I'm back in my parents' place in south tampa so I might need to do the deal somewhere else
ham	Good morning, my Love ... I go to sleep now and wish you a great day full of feeling better and opportunity ... You are my
ham	Kothi print out marandratha.
ham	But we havent got da topic yet rite?
ham	Ok no problem... Yup i'm going to sch at 4 if i rem correctly...
ham	Thanks, I'll keep that in mind
ham	Aah bless! How's your arm?
ham	Dear Sir,Salam Alaikkum.Pride and Pleasure meeting you today at the Tea Shop.We are pleased to send you our contact n
ham	Gal n boy walking in d park. gal-can i hold ur hand? boy-y? do u think i would run away? gal-no, jst wana c how it feels wall
ham	What makes you most happy?
ham	Wishing you a wonderful week.

Figure: Sample Data

3.2. Data Preprocessing

Dataset cleaning is a critical preprocessing step in spam email detection, ensuring that training and testing data is accurate and relevant. It involves removing outliers, handling missing values, and eliminating unwanted features to enhance model performance. Once cleaned, datasets are merged to retain only key features: text (email content) and label (spam or non-spam). Textual data is then processed by removing tags, tokenizing sentences, eliminating stop words, and applying lemmatization using NLTK and Regex libraries. Feature vectors are generated using Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF). Bow represents text as numerical vectors based on word presence, while TF-IDF assigns importance to words by comparing term frequency across documents, ensuring that frequently occurring yet less informative words are downweighed. These techniques improve model accuracy and efficiency in spam classification.

3.3. Data Splitting

The data splitting is done to create two kinds of data Training data and testing data. Training data is used to train the machine learning models and testing data is used to test the models and analyse results. 80% of total data is selected as training data and remaining data is testing data.

3.4. Architecture

The architecture of spam email detection consists of multiple stages, beginning with data collection, where a large dataset of spam and non-spam emails is gathered. The next step is data preprocessing, which involves cleaning the data by removing duplicates, null values, and irrelevant features, followed by feature extraction, where techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) or word embeddings are used to convert text into numerical representations. After preprocessing, the data is split into training (80%) and testing (20%) datasets, where the training data is fed into a machine learning model such as Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), Random Forest, or Deep Learning models like Neural Networks. The model learns patterns that distinguish spam from non-spam emails. Once trained, the model is tested with the test dataset to evaluate performance using metrics like accuracy, precision, recall, and F1-score.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

Finally, the deployed spam detection system classifies incoming emails as either spam or non-spam, ensuring efficient filtering and improving email security.

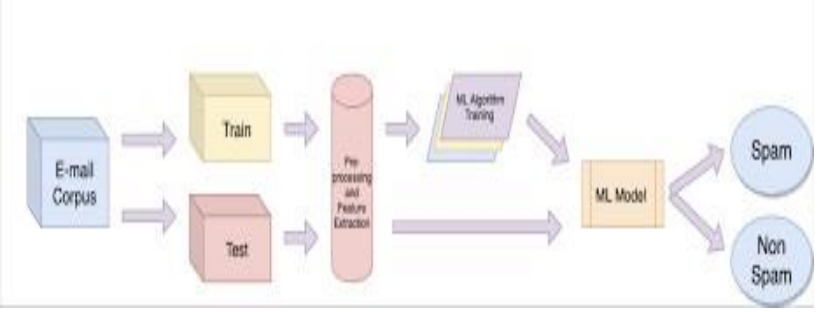

Figure: System Architecture

3.5. Naïve Bayes Classifier

A Naïve Bayes classifier is a supervised probabilistic machine learning model used for classification tasks, based on Bayes' Theorem. It assumes that all features used to predict the target are independent, which is rarely true in real-world data but often works well in practice, hence the term "naïve."
The formula is:

$$P(A|B) =P(B|A)\ P(A)P(B)P(A|B) = \frac{P(B|A)\ P(A)}\ {P(B)}\ P(A|B) =P(B)P(B|A)\ P(A)$$

Here, P(A|B) P(A|B) P(A|B) is the posterior probability, P(B|A) P(B|A) P(B|A) is the likelihood, P(A)P(A)P(A) is the prior probability, and P(B)P(B)P(B) is the evidence. Naïve Bayes is commonly used in text classification but treats all words as equally important, which can be a limitation. Despite this, it remains efficient and is often combined with other language processing techniques.

3.6. Logistic Regression

Logistic Regression is a supervised machine learning algorithm used to model the probability of a certain class or event, primarily for binary classification when the data is linearly separable. The relationship between features and the outcome is represented by the equation

$$z=b0+b1x1+b2x2+…+bnxnz = b\_0 + b\_1\ x\_1 + b\_2\ x\_2 + \ldots + b\_n\ x\_n$$

Here, zz represents the odds, calculated as the ratio of the probability of an event occurring to the probability of it not occurring. The odds are passed through a sigmoid function, defined as:

$$h(z)=11+e−zh(z) = \frac{1}{1 + e^{-z}}$$

The output is a probability between 0 and 1, which determines the class. Typically, 0.5 is the threshold, where values below it classifies as NO and values above it classifies as YES, though this threshold can be adjusted as needed.

3.6. Performance Evaluation

To ensure comprehensive assessment, multiple evaluation metrics were employed:

Accuracy

This metric measured the overall correctness of predictions. While effective for balanced datasets, accuracy alone was insufficient for analysing imbalanced outcomes. Accuracy=TP+TN+FP+FN /TP+TN

Process of hiding a secret audio/video/text within a larger one in such a way that someone cannot know the presence or contents of the hidden audio/video/text. Steganography is, many times, confused with cryptography as both the techniques are used to secure information. The difference lies in the fact that steganography hides the data so that nothing appears out of ordinary while cryptography encrypts the text, making it difficult for an out sider to infer anything from it even if they do attain the encrypted text. Both of them are combined to increase the security against various malicious attacks. The purpose of Steganography is to maintain secret communication between two parties. Using the LSB technique, which facilitates plain text hiding in an image as well as hiding files in an image. It works with JPEG and PNG formats for the cover image and always creates PNG Stego image due to its lossless compression.

International Multidisciplinary Research Journal Reviews (IMRJR)

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

Least Significant Bit Embeddings (LSB) are a general steganographic technique that may be employed to embed data into a variety of digital media, the most studied applications are using LSB embedding to hide one image inside another. In this image steganography software, we can hide the data using LSB embed techniques.
Steganography is the process of hiding a secret audio/video/text within a larger one in such a way that someone cannot know the presence or contents of the hidden audio/video/text. Steganography is, many times, confused with cryptography as both the techniques are used to secure information. The difference lies in the fact that steganography hides the data so that nothing appears out of ordinary while cryptography encrypts the text, making it difficult for an out sider to infer anything from it even if they do attain the encrypted text. Both of them are combined to increase the security against various malicious attacks. The purpose of Steganography is to maintain secret communication between two parties. Using the LSB technique, which facilitates plain text hiding in an image as well as hiding files in an image. It works with JPEG and PNG formats for the cover image and always creates PNG Stego image due to its lossless compression. Least Significant Bit Embeddings (LSB) are a general steganographic technique that may be employed to embed data into a variety of digital media, the most studied applications are using LSB embedding to hide one image inside another. In this image steganography software, we can hide the data using L

IV.RESULT AND DISCUSSION

The Naïve Bayes, based on Bayes' theorem, efficiently detected spam emails by assigning probabilities after preprocessing tasks like tokenization and stop-word removal. It had a high recall rate but a slightly higher false positive rate due to its independence assumption. Logistic Regression, using a sigmoid function, performed better by considering word correlations, resulting in a lower false positive rate but requiring more computational resources. While Naïve Bayes excelled in speed and recall, Logistic Regression offered better precision, making it preferable when reducing false positives is essential. Both models proved effective for spam detection.

```
classification report:
              precision    recall  f1-score   support

         ham       0.97      0.99      0.98      1445
        spam       0.95      0.80      0.87       227

    accuracy                           0.97      1672
   macro avg       0.96      0.90      0.93      1672
weighted avg       0.97      0.97      0.97      1672


confuion matrix:
 [[1436    9]
 [  45  182]]

Accuracy score: 0.9677033492822966
```

Figure: Logistic Regression Analysis

Spam Email Detection project using Logistic Regression. The displayed classification report presents precision, recall, F1-score, and support for both "ham" (non-spam) and "spam" emails. The model achieves a precision of 97% for ham and 95% for spam, indicating that most predictions are accurate. The recall for ham is 99%, meaning almost all ham emails are correctly identified, while spam recall is 80%, suggesting some spam emails are misclassified as ham. The overall accuracy of the model is 96.77%, meaning it correctly classifies most emails. The confusion matrix shows that 1,436 ham emails were correctly classified, while 9 were misclassified as spam. Similarly, 182 spam emails were detected correctly, while 45 were mistakenly labeled as ham, affecting recall. A heatmap visualization of the confusion matrix is also generated using Seaborn, though it is only partially visible in the output.

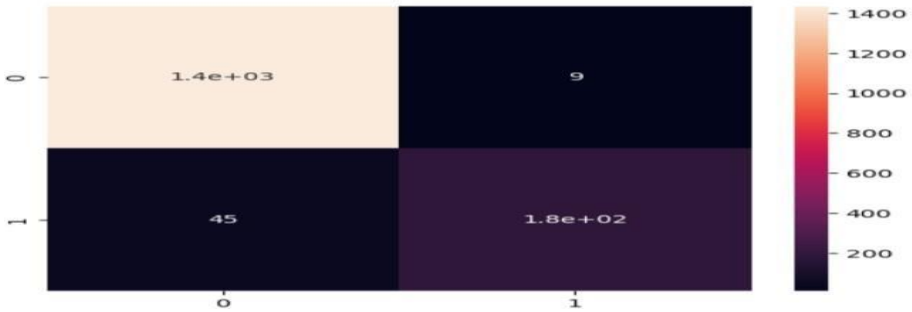

Figure: Confusion Matrix

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

```
Performance of Naive bayes:

classification report:
              precision    recall  f1-score   support

         ham       0.99      0.98      0.98      1445
        spam       0.89      0.91      0.90       227

    accuracy                           0.97      1672
   macro avg       0.94      0.95      0.94      1672
weighted avg       0.97      0.97      0.97      1672


confuion matrix:
 [[1420   25]
 [  20  207]]

Accuracy score: 0.9730861244019139
```

Figure: Naïve Bayes Analysis

The classification report evaluates the model's precision, recall, F1-score, and support for both "ham" (non-spam) and "spam" emails. The model achieves 99% precision for ham and 89% precision for spam, indicating that it effectively distinguishes between legitimate and spam emails. The recall for spam is 91%, meaning that most spam emails are correctly identified, while ham recall is 98%, suggesting a few false positives. The overall accuracy of the model is 97.3%, demonstrating strong performance. The confusion matrix shows that 1,420 ham emails were correctly classified, while 25 were misclassified as spam. Similarly, 207 spam emails were accurately detected, while 20 were incorrectly labeled as ham. Compared to other models, Naïve Bayes performs well due to its probabilistic nature and ability to handle text classification efficiently, though minor misclassifications exist.

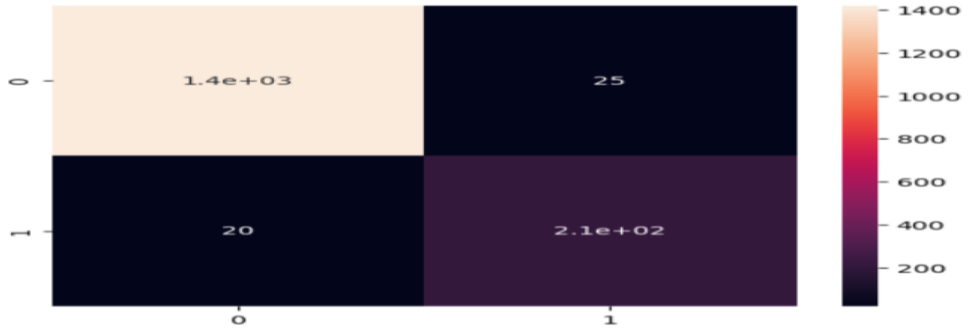

Figure: Confusion Matrix

Table: Comparison Of Algorithms Performance

ALGORITHM	ACCURACY	PRECISION	RECALL	F1 SCORE
Logistic Regression	96	0.95	0.80	0.87
Naïve Bayes	97	0.89	0.91	0.90

V.CONCLUSION AND DISCUSSION

The Spam Email Detection project successfully demonstrates the effectiveness of machine learning models in identifying spam emails with high accuracy. Through comprehensive data preprocessing, feature extraction, and model training, the study evaluates different classification techniques, particularly Naïve Bayes and Logistic Regression, to determine their suitability for spam detection. Naïve Bayes, with its probabilistic approach, achieves a higher recall for spam (91%), making it well-suited for detecting spam emails with minimal false negatives, although it sometimes misclassifies legitimate emails as spam. On the other hand, Logistic Regression, leveraging word relationships rather than independence assumptions, provides a higher precision (97%), ensuring fewer false positives but at the cost of slightly lower recall for spam emails. The overall accuracy for both models is approximately 97%, indicating their reliability in practical applications. However, misclassification issues remain, particularly with spam recall and false positives, which could be further minimized using advanced ensemble methods, deep learning techniques, or hybrid approaches. The results suggest that while Naïve Bayes is preferable for quick, high-recall filtering, Logistic Regression is better when precision is crucial. Integrating both models or fine-tuning hyperparameters could enhance spam detection accuracy, making email filtering systems more efficient and reducing user inconvenience due to incorrect classification. The Spam Email Detection project effectively demonstrates the application of machine learning algorithms to classify emails as either spam or ham (non-spam) with high accuracy. By leveraging Natural Language Processing (NLP) techniques, including text preprocessing, feature extraction (such as TF-IDF or CountVectorizer), and model training, the study evaluates the performance of different classifiers, specifically Naïve Bayes and Logistic Regression.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 4, April 2025
DOI 10.17148/IMRJR.2025.020403

The results indicate that both models achieve an overall accuracy of approximately 97%, making them highly effective for spam detection tasks.

Naïve Bayes, known for its probabilistic approach and assumption of word independence, exhibits a higher recall for spam (91%), meaning it successfully identifies the majority of spam emails while allowing a few false positives. This makes it suitable for real-world applications where detecting spam is a priority, even if it results in some legitimate emails being mistakenly classified as spam.

REFERENCES

[1]. C. Yang, R. Harkreader, and G. Gu, "Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers", In: Recent Advances in Intrusion Detection, Springer Berlin/Heidelberg, pp.318-337, 2011.

[2]. S. Kumar, and S. Arumugam, "A Probabilistic Neural Network Based Classification of Spam Mails Using Particle Swarm Optimization Feature Selection", Middle-East Journal of Scientific Research, Vol.23, No.5, pp.874- 879, 2015.

[3]. N. P. DíAz, D. R. OrdáS, F. F. Riverola, and J. R. MéNdez, "SDAI: An integral evaluation methodology for content-based spam filtering models", Expert Systems with Applications, Vol.39, No.16, pp.12487-12500, 2012.

[4]. A. K. Sharma, S. K. Prajapat, and M. Aslam, "A Comparative Study Between Naive Bayes and Neural Network (MLP) Classifier for Spam Email Detection", In: IJCA Proceedings on national Seminar on Recent Advances in Wireless Networks and Communications. Foundations of Computer Science (FCS), pp.12-16, 2014.

[5]. W. Ma, D. Tran, and D. Sharma, "A novel spam email detection system based on negative selection", In: Proc. of Fourth International Conference on Computer Sciences and Convergence Information Technology, ICCIT 09, Seoul, Korea, pp.987-992, 2009.

[6]. T. S. Guzzella, and W. M. Caminhas, "A review of machine learning approaches to spam filtering", Expert Systems with Applications, Vol.36, No.7, pp.10206- 10222, 2009.

[7]. N. Kumar, S. Sonowal, and Nishant, “Email spam detection using machine learning algorithms,” in Proceedings of the 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), pp. 108–113, IEEE, Coimbatore, India, July 2020.

[8]. G. Jain, M. Sharma, and B. Agarwal, “Optimising semantic lstm for spam detection,” International Journal of Information Technology, vol. 11, no. 2, pp. 239–250, 2019.

[9]. F Masood, G. Ammad, A. Almogren et al., “Spammer detection and fake user identification on social networks,” IEEE Access, vol. 7, pp. 68140–68152, 2019.

[10]. G. Chandrashekar, and F. Sahin, "A survey on feature selection methods", Computers & Electrical Engineering, Vol.40, No.1, pp.16-28, 2014.