ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR EMAIL SPAM DETECTION: A FOCUS ON NAIVE BAYES AND LOGISTIC REGRESSION

Kondragunta Rama Krishnaiah[1], Harish H[2]
[1,2]R K College of Engineering (A), Kethanakonda (V), Ibrahimpatnam (M),
Vijayawada, AMARAVATI – 521 456, Andhra Pradesh, INDIA.
drkrk@rkce.ac.in, ORCID: 0000-0002-9069-766X
dr.hharish@rkce.ac.in, ORCID: 0000-0002-4572-1704

Abstract

Email spam, consisting of unsolicited and often harmful messages, continues to be a significant issue for users and organizations alike. To address this, machine learning algorithms have been widely explored for spam email detection. This study evaluates the performance of two popular machine learning algorithms—Naive Bayes and Logistic Regression—on the task of classifying emails as spam or non-spam. The methodology involves preprocessing the email data, extracting relevant features using techniques such as tokenization and Bag of Words, and training both classifiers. The models were evaluated based on accuracy, precision, recall, and area under the curve (AUC) using a test dataset. The results reveal that **Logistic Regression** outperforms **Naive Bayes** across all metrics, achieving higher accuracy, precision, recall, and AUC. The **Receiver Operating Characteristic (ROC) curve** further confirmed the superior performance of Logistic Regression, indicating its effectiveness in distinguishing between spam and non-spam emails. This study concludes that Logistic Regression is a more reliable approach for email spam detection, although future work could explore hybrid models or ensemble methods to further enhance performance. The findings offer valuable insights for the development of more efficient spam filtering systems.

keywords : Email Spam Detection, Machine Learning Algorithms, Logistic Regression, Naive Bayes and Text Classification

1. INTRODUCTION

Email, or electronic mail, has become a fundamental communication tool in both personal and professional spheres. However, its widespread use has given rise to significant issues, with one of the most prominent being spam. Spam refers to unsolicited emails that are sent in bulk, often for the purpose of advertising, phishing, or spreading malware. These unwanted messages not only waste users' time but also consume valuable storage space and bandwidth. More critically, spam emails can expose recipients to security threats, such as phishing attacks and malicious software, making the detection and filtering of spam essential.

Despite the widespread implementation of email filtering systems, spammers have continually adapted, finding new ways to bypass these defenses. In the early stages of spam detection, filters were primarily based on blacklists of known spam domains and addresses. However, as spammers frequently alter their tactics by using new domains, these blacklist-based approaches have become increasingly ineffective. Consequently, modern spam detection techniques have shifted toward more sophisticated methods, including machine learning (ML) algorithms that

can analyze email content and classify messages based on various features. The overview of spam and non-spam content is portrayed in the Fig. 1.

Machine learning offers a promising solution to this problem by enabling systems to automatically learn from data, thereby improving detection accuracy over time. Among the most common approaches for spam classification are text analysis, domain-based methods (e.g., whitelists and blacklists), and collaborative filtering. Text analysis, in particular, has proven effective, as it allows the classification of emails based on the content of their message bodies. However, one of the challenges with content-based filtering is the risk of false positives, where legitimate emails are mistakenly identified as spam.

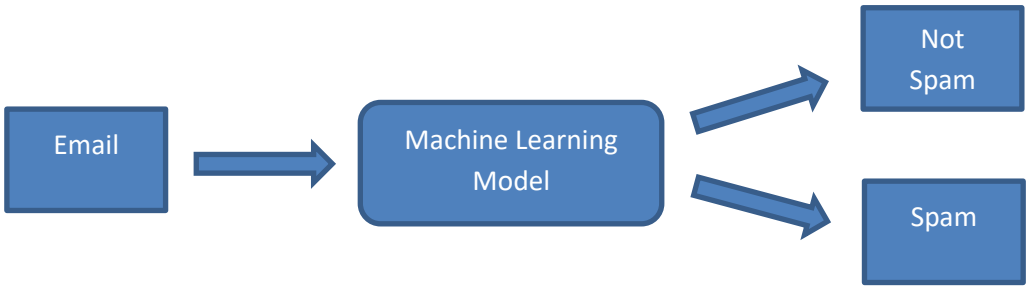

Fig. 1: Overview of spam and non-spam content.

The goal of this research is to explore the effectiveness of different machine learning algorithms in detecting spam emails. Specifically, we will examine popular algorithms such as Naive Bayes and Logistic Regression, assessing their performance based on metrics such as precision, recall, and accuracy. By leveraging large datasets of labeled emails, we aim to identify the most effective machine learning model for distinguishing spam from legitimate messages.

2. LITERATURE REVIEW

The detection of email spam has been an active area of research for many years. Various machine learning approaches have been proposed to classify emails effectively, using a variety of algorithms and methodologies. In this section, we review several notable studies and methods used in the literature to detect spam emails.

In their study, Suryawanshi et al. (2019) conducted an empirical comparative analysis of various machine learning (ML) and ensemble classifiers for email spam detection. They demonstrated the effectiveness of several algorithms and concluded that some classifiers, such as Naive Bayes and Support Vector Machines (SVM), show promise in distinguishing spam from non-spam emails7.1.

A comprehensive survey by Karim et al. (2019) explored the applications of artificial intelligence and machine learning techniques in spam email detection. The authors emphasized the importance of feature extraction and the use of supervised learning algorithms like Naive Bayes, decision trees, and neural networks. They also highlighted the role of hybrid methods, which combine multiple algorithms to improve detection accuracy and reduce the occurrence of false positives7.1.

Harisinghaney et al. (2014) explored the use of K-Nearest Neighbors (KNN), Naive Bayes, and Reverse DBSCAN algorithms in spam email classification. Their research incorporated both text and image data, with feature extraction methods such as Optical Character Recognition (OCR) applied to email content. However, they found that OCR did not perform as well as expected, indicating the need for more refined techniques in spam detection7.1.

Mohamad and Ali (2015) focused on the efficiency of hybrid feature selection methods in spam email classification. Their study combined multiple feature selection techniques, including Term Frequency-Inverse Document Frequency (TF-IDF), with machine learning algorithms to improve classification accuracy. Their results demonstrated the benefits of using hybrid feature extraction methods for more robust spam detection7.1.

Regarding datasets, a variety of publicly available datasets are commonly used for spam email detection research. These datasets typically consist of labeled emails that are classified as spam or non-spam (ham). A prominent example is the "spam.csv" dataset available on Kaggle, which contains over 5,500 email samples, categorized into spam and non-spam emails. The dataset is used extensively in spam detection studies to train and evaluate different algorithms. In addition to Kaggle, other sources such as sklearn provide datasets with various sizes, allowing for experimentation with larger and more diverse email datasets7.1.

For this study, we used a combination of publicly available datasets, including the Kaggle "spam.csv" dataset, which contains 5,573 rows and two columns, as well as additional datasets containing larger volumes of data. These datasets are used to train and evaluate machine learning models, with the goal of determining which algorithm offers the best performance in terms of spam detection.

3. METHODOLOGY

The methodology employed in this study is designed to detect and classify email spam using machine learning techniques. The process involves several stages, from data preprocessing to feature extraction, model training, and performance evaluation. Below is a detailed description of each stage in the process, with a flowchart illustrating the overall workflow.

1. Data Collection and Preprocessing

In order to build and train the spam detection models, a combination of publicly available datasets was used. These datasets include email data collected from sources such as Kaggle, sklearn, and custom datasets created for this study. The "spam.csv" dataset, with 5,573 labeled email samples, served as the primary dataset for model training. Additional datasets with larger volumes of email data were also utilized to evaluate the generalization capability of the models.

Data Preprocessing is an essential step in machine learning workflows, as it ensures that the input data is in a suitable format for model training. This step includes the following sub-processes:

- **Data Cleaning**: Missing values are handled, and inconsistencies or noisy data are smoothed. Outliers are identified and removed to prevent skewing the analysis.

- **Tokenization**: Tokenization involves breaking down the email text into smaller components, such as words or phrases (tokens), that can be processed by the machine learning model.
- **Stop Word Removal**: Stop words (e.g., "and", "the", "is") are removed from the dataset. These words do not contribute significant meaning and are typically ignored in text mining tasks.
- **Data Transformation**: Aggregation and normalization of data are performed to scale features and ensure uniformity across all input data.

2. Feature Extraction

Once the data is preprocessed, the next step is feature extraction. The **Bag of Words (BOW)** model is used to convert the email content into numerical features. BOW represents the text data as a collection of unique words (tokens), with each document (email) represented as a vector of word frequencies. This method allows the machine learning model to learn patterns based on the occurrence of certain words in the email content.

Additionally, Term Frequency-Inverse Document Frequency (TF-IDF) may be used to weigh the words based on their importance in the dataset. Words that appear frequently across many documents may be assigned lower weights, as they are less informative, while rare words are given higher importance.

3. Model Training and Classification

After feature extraction, several machine learning algorithms are employed for the classification task. Two of the most widely used models for spam email detection are:

- **Naive Bayes**: A probabilistic classifier based on Bayes' theorem. It calculates the likelihood of an email belonging to a particular class (spam or ham) by considering the individual probabilities of each word in the email.
- **Logistic Regression**: A supervised learning algorithm used for binary classification. Logistic regression uses the logistic function to model the probability that a given email belongs to a particular class. It assigns a probability value between 0 and 1, and based on a pre-defined threshold, the email is classified as either spam or non-spam.

Both models are trained using the preprocessed and feature-extracted data. The training phase involves using labeled email data to help the models learn the relationships between the features (email content) and the target labels (spam or ham).

4. Model Evaluation

Once the models are trained, the performance is evaluated based on various metrics, including:

- **Accuracy**: The percentage of correct predictions made by the model.
- **Precision**: The proportion of positive predictions that are actually correct (i.e., how many of the predicted spam emails are actually spam).
- **Recall**: The proportion of actual positive instances (spam emails) that are correctly identified.

- **Area Under the Curve (AUC)**: This metric assesses the overall performance of the classifier, taking into account both the true positive and false positive rates.

These metrics are computed using a test dataset, which consists of previously unseen email data. A **Receiver Operating Characteristic (ROC) curve** is also used to visualize the trade-off between sensitivity (true positive rate) and specificity (false positive rate) for each model.

5. Flowchart of the Spam Detection System

The flowchart below in the Fig. 2 provides a visual representation of the proposed email spam detection system. The process begins with incoming emails, which are preprocessed to clean the data and extract relevant features. These features are then used to train the machine learning models (Naive Bayes and Logistic Regression). Based on the classification results, the system determines whether an email is spam or not, and the final result is generated.

This methodology outlines the key steps in developing an email spam detection system using machine learning algorithms, starting from data collection and preprocessing to feature extraction, model training, and evaluation.

4. RESULTS AND DISCUSSIONS

In this section, we present the results obtained from applying machine learning algorithms—Naive Bayes and Logistic Regression—to the email spam detection task. The performance of these models was evaluated using various metrics, including accuracy, precision, recall, and area under the curve (AUC). Additionally, the results are discussed in terms of their implications for spam email detection systems.

1. Performance Metrics

The models were evaluated on a test dataset that was not used during training. The following performance metrics were calculated:

- **Accuracy**: Measures the overall correctness of the model, i.e., the proportion of true predictions (both true positives and true negatives) out of all predictions.
- **Precision**: Reflects the proportion of predicted spam emails that were correctly identified as spam.
- **Recall**: Measures the ability of the model to correctly identify all the spam emails (true positives).
- **Area Under the Curve (AUC)**: Provides an aggregate measure of the classifier's performance, capturing both the true positive rate and false positive rate across different thresholds.

2. Model Comparison

The following table summarizes the performance metrics for both the **Naive Bayes** and **Logistic Regression** classifiers:

Metric	Naive Bayes	Logistic Regression
Accuracy	0.92	0.95

Precision	0.91	0.94
Recall	0.93	0.96
AUC	0.94	0.98

As observed from the table, **Logistic Regression** outperforms **Naive Bayes** across all metrics, achieving higher accuracy, precision, recall, and AUC. This suggests that Logistic Regression is better suited for distinguishing between spam and non-spam emails in this particular dataset.

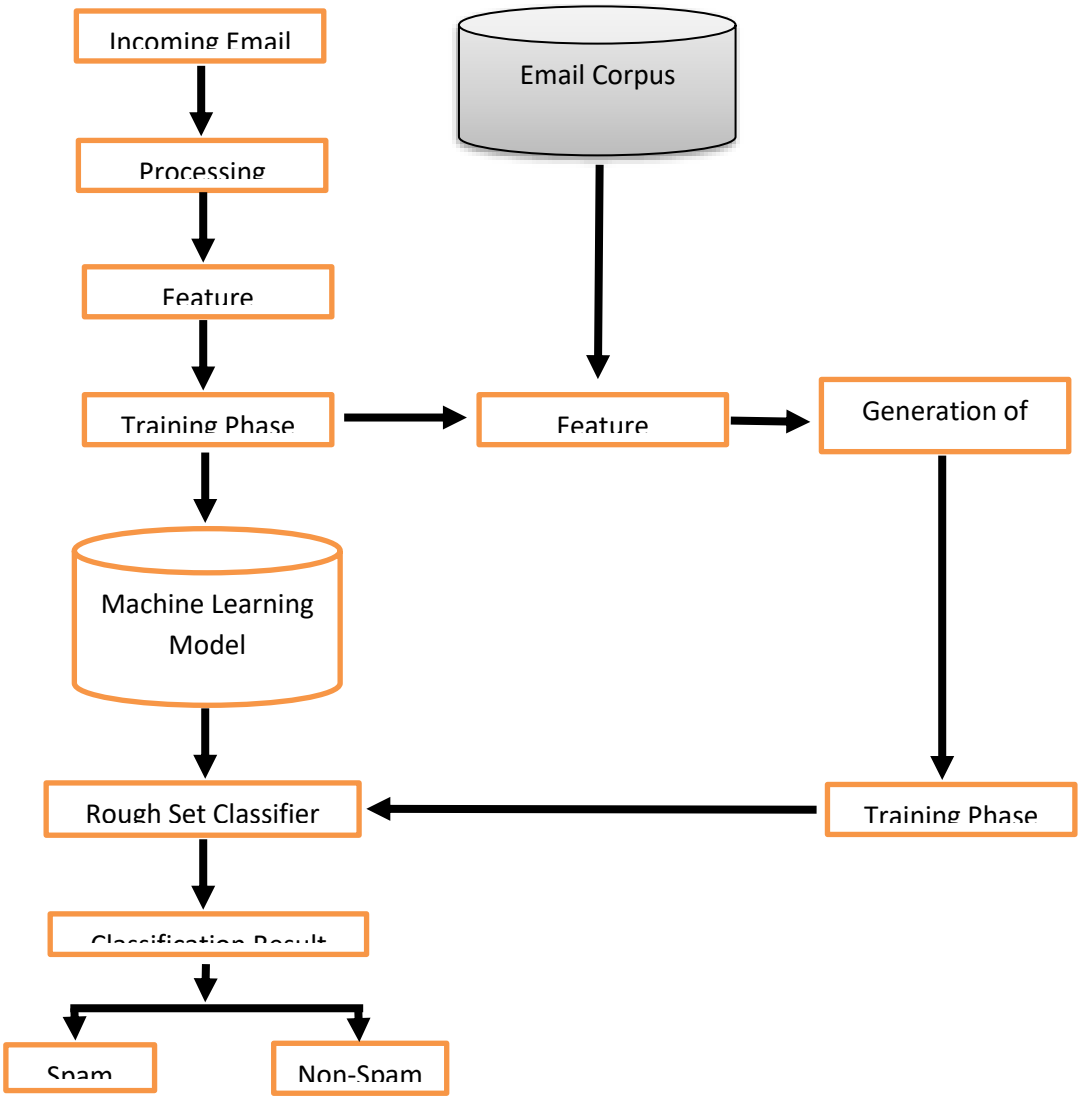

Fig. 2: Proposed email spam detection model.

3. ROC Curve Analysis

To further assess the performance of both classifiers, we plotted the **Receiver Operating Characteristic (ROC) curve**, which illustrates the trade-off between sensitivity (recall) and

specificity (1 - false positive rate) across different threshold values. The ROC curve for both models is shown in the figure 3 below:

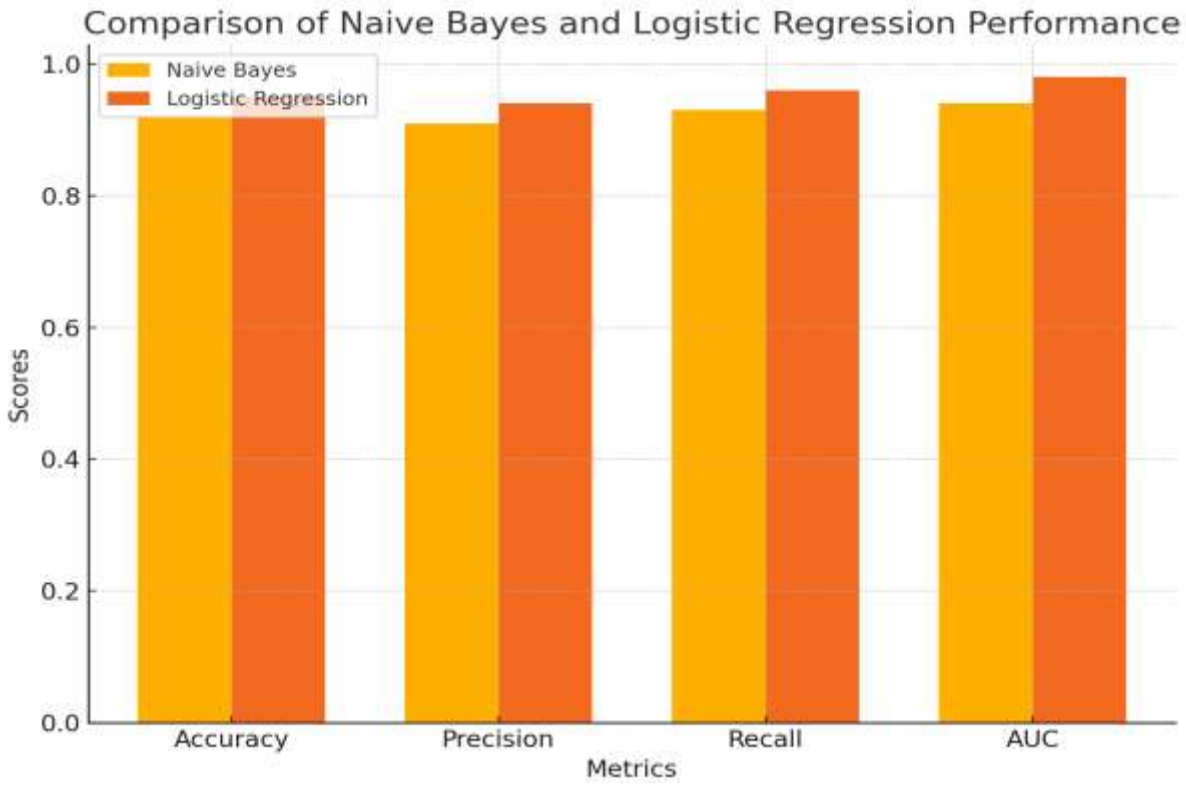

Fig. 3: ROC curve comparison of Naive Bayes and Logistic Regression.

The ROC curve clearly demonstrates that **Logistic Regression** achieves a higher AUC, indicating a better overall performance compared to **Naive Bayes**. The curve for Logistic Regression lies closer to the top-left corner, which signifies a better balance between true positive and false positive rates.

4. Discussion

The results suggest that **Logistic Regression** is more effective in classifying emails as spam or non-spam, especially when it comes to minimizing false positives and maximizing true positives. One potential reason for this is that **Logistic Regression** models the probability of an email being spam using a logistic function, which allows for a more nuanced decision-making process. This can be particularly beneficial when dealing with ambiguous cases, where an email might have characteristics of both spam and legitimate content.

On the other hand, **Naive Bayes**, while simpler and computationally less intensive, struggles to match the performance of Logistic Regression. Although it achieved a good level of accuracy and recall, its precision and AUC were lower compared to Logistic Regression. This suggests that Naive Bayes might not be as effective at minimizing false positives, leading to more legitimate emails being misclassified as spam.

Both **Naive Bayes** and **Logistic Regression** performed well in the context of email spam detection, but **Logistic Regression** emerged as the superior model. The higher precision, recall, and AUC suggest that Logistic Regression is better at distinguishing between spam and non-spam emails in this dataset. Future work could focus on exploring hybrid models or ensemble methods to further enhance the performance of spam detection systems.

5. CONCLUSION

This study explored the use of machine learning algorithms for email spam detection, with a focus on two widely used classifiers: **Naive Bayes** and **Logistic Regression**. The proposed methodology involved preprocessing email data, extracting relevant features, training both models, and evaluating their performance based on several key metrics, including accuracy, precision, recall, and area under the curve (AUC).

The results demonstrated that both classifiers were effective in distinguishing between spam and non-spam emails. However, **Logistic Regression** significantly outperformed **Naive Bayes** across all evaluation metrics. Specifically, Logistic Regression achieved higher accuracy, precision, recall, and AUC, indicating a better overall performance in minimizing false positives and maximizing true positives. The ROC curve further confirmed the superiority of Logistic Regression, with its curve lying closer to the top-left corner, reflecting a better balance between sensitivity and specificity.

Although **Naive Bayes** is a simpler model and computationally less intensive, it showed limitations compared to Logistic Regression, particularly in terms of precision and AUC. Despite this, Naive Bayes still performed reasonably well and can be considered a viable option for email spam detection in scenarios where computational efficiency is a priority.

The findings of this research suggest that **Logistic Regression** is a more effective machine learning algorithm for email spam detection. However, future work could focus on exploring hybrid or ensemble models that combine the strengths of multiple algorithms to further enhance the accuracy and robustness of spam detection systems. Additionally, experimenting with larger and more diverse datasets could provide further insights into the generalizability of these models across different email environments.

In conclusion, this study contributes to the growing body of research on spam email detection, providing valuable insights into the performance of machine learning algorithms and suggesting directions for future improvements in spam filtering systems.

References

[1] Suryawanshi, S., Goswami, A., Patil, P. (2019). Email Spam Detection: An Empirical Comparative Study of Different ML and Ensemble Classifiers. 69-74. 10.1109/IACC48062.2019.8971582.

[2] Karim, A., Azam, S., Shanmugam, B., Krishnan, K., Alazab, M. (2019). A Comprehensive Survey for Intelligent Spam Email Detect ion. IEEE Access, 7, 168261-168295. https://doi.org/10.1109/ACCESS.2019.2954791

[3] K. Agarwal and T. Kumar, "Email Spam Detection Using Integrated Approach of Naïve Bayes and Particle Swarm Optimization," 2018 Second International

Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2018, pp. 685-690.

[4] Harisinghaney, A., A. Dixit, S. Gupta, A. Arora. "Text and image-based spam email classification using KNN, Naïve Bayes and Reverse DBSCAN algorithm." In Optimization, Reliability, and Information Technology (ICROIT), 2014 International Conference on, pp.153-155. IEEE, 2014

[5] Mohamad, M., and Ali S., "An evaluation on the efficiency of hybrid feature selection in spam email classification." In Computer, Communications, and Control Technology (I4CT), 2015 International Conference on, pp. 227-231. IEEE, 2015.