Classifying Spam Email Using Machine Learning

This Project Report is Submitted in Partial Fulfillment of the Requirement for the Degree of Master of Science (Engg.) in Information and Communication Technology

Submitted By

ID No: 22209030
Session: 2021-2022

Department of Information and Communication Technology
Faculty of Engineering
Comilla University, Bangladesh
February, 2025

Acknowledgement

First and foremost, I want to thank Almighty God. All praise and gratitude are due to God for granting me the strength, patience, and knowledge to complete this project successfully.

I want to thank my respected supervisor, Pintu Chandra Paul sir, for his continuous guidance, insightful feedback, and invaluable support throughout the duration of this project. His expertise and encouragement have been instrumental in shaping the direction of my project.

I am deeply thankful to the Department of ICT, Comilla University, for providing the opportunity and resources to undertake this project, as well as for fostering a supportive academic environment.

I also want to thank my family and friends for their unwavering encouragement, patience, and understanding during this challenging journey. Their support has been my source of strength.

Finally, I am grateful to everyone who contributed to this project in any capacity. Your inspiration, suggestions, and collaboration were vital to the successful completion of this project.

Abstract

Email is a widely used method of transferring data and information via digital devices. Email is widely used for data communication between servers by millions of users worldwide. Unwanted emails, or spam, have become a problem for big corporations and organizations as the number of spam grows rapidly each year. Spam is both unpleasant and perhaps destructive. Spam consumes computer, server, and network resources, creating bottlenecks and slowing down digital devices. Furthermore, consumers spend a significant amount of time deleting unsolicited emails. Spam emails are a huge cybersecurity concern, with research indicating that 91% of internet assaults begin from malicious emails, while 97% of users fail to accurately detect them. To reduce these dangers, this work proposes a machine learning-based solution to spam email categorization. The dataset is preprocessed with Natural Language Processing (NLP) techniques before features are extracted using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization approach. To find the best machine learning model for spam detection, five models are trained and assessed: Logistic Regression, Random Forest Classifier, Decision Tree Classifier, K-Nearest Neighbours Classifier, Multi-Layer Perceptron (MLP) Classifier. The models are compared using performance criteria such accuracy, precision, recall and F1-score. The findings show that it has the best accuracy, which makes it a dependable automatic spam filtering solution. By offering an effective spam categorisation model that can shield users from phishing and other online dangers, this study helps to improve email security.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Introduction

Spam Email Classification is a technique to safeguard personal data from spammers that target users by delivering dangerous links via email, which encourages them to search for their system and use it for unethical and criminal activities like fraud and phishing. Spam is a tactic used to deceive individuals with minimal awareness of these kinds of assaults. To safeguard private information from this kind of social media assault, the entire company has security flaws that have caused great concern for consumers, website developers, and experts. Every person might be impacted differently by email spam, which can steal personal information or even bank account information and steal funds [1]. The user would find it challenging to manually remove every unnecessary or undesired email. In order to address this issue, a variety of spam detection techniques have been created in response to the growing problem of spam emails throughout time. All emails are generally categorised as "Ham" and "Spam." In contrast to spam, which are unwanted mass or commercial communications in an inbox, ham transmissions are intended or safe acceptable messages. Email communications are filtered or classified as spam or ham in order to better separate them and automatically remove spam. Typically, a number of factors or elements aid in the detection of spam emails [2]. Spam may have a major negative impact on network performance, storage, mail servers, CPU, and computer memory, despite the fact that it is extremely aggravating to digital users. Additionally, spam has a significant negative impact on computer network speed and takes up digital users' time to remove and discard unwanted emails, which reduces their ability to be productive. However, spam may include malicious links that reveal private information such as credit card numbers, bank account details, and passwords. Additionally, spam could include pornographic material that should not be shown to minors. Spam emails can thereby aid cybercriminals in phishing, denial of service (DoS) assaults, and other activities [3]. When an email has poor language, distorted photos, distorted symbols or logos, bad links, alluring offers, or time-based subscriptions that compel people to sign up right away, it may be deemed spam. Phishing is another serious cybercrime that targets people and deceives them into clicking on links or subscribing in order to obtain their personal information, such as login credentials for social media accounts like Facebook and Twitter or, in the worst case, their online banking information. Phishing emails are regarded as spam as well [2].

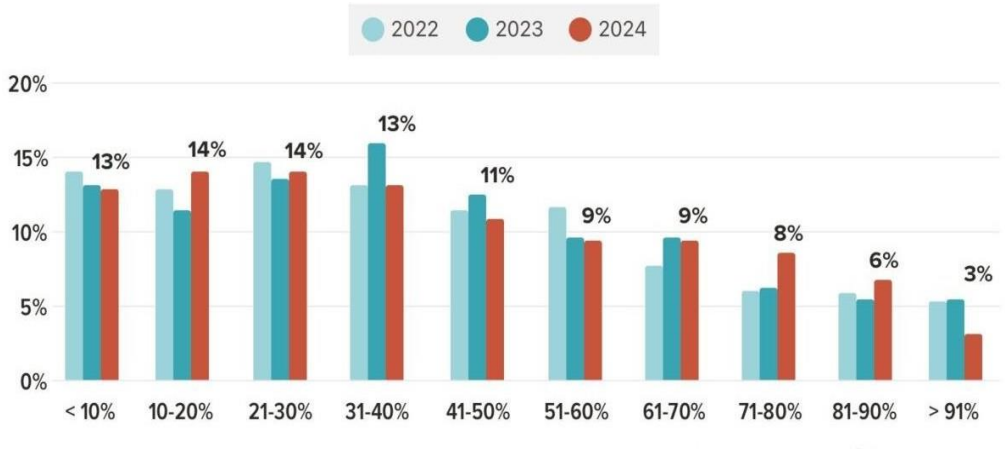

Figure 1.1: The percentage of everyday emails that are spam

This bar chart shows the percentage of daily emails that are classified as spam, based on survey responses from three different years: 2022 (light blue), 2023 (blue), and 2024 (red).

- Most respondents receive between 10-40% spam emails daily.
    - The 10-20% and 21-30% spam categories are the most common, with 14% of respondents in each category in 2024.
    - The 31-40% spam category had the highest percentage in 2023 (13%), slightly decreasing in 2024.
- Spam levels in lower and higher ranges are less common.
    - Less than 10% spam: Around 13% of respondents in 2023 and 2024.
    - More than 91% spam: The lowest percentage, only 3% in 2024.
- Trends Over the Years:
    - There is a relatively stable distribution across years, with small fluctuations in each category.
    - The 71-80% and 81-90% spam categories increased slightly in 2024, suggesting that some users are receiving more spam over time.
    - The 41-50% spam category decreased slightly from 2023 to 2024.

Effective spam email categorisation algorithms are essential since a significant portion of consumers get 10–40% spam every day. The marginal rise in high spam rates (71–90%) in 2024 raises the possibility that spammers are becoming more proficient, which would make categorisation more crucial. Therefore, considering that spam patterns change every year, machine learning models should strive for high accuracy in differentiating spam from authentic emails.

1.2 Objectives

In order to improve email security and lessen cyber dangers, the primary goal of this project is to create an effective spam email categorisation model using machine learning techniques.

The particular goals of this research are:

1. To preprocess the data by addressing duplicate entries and missing values, and by normalising the data by scaling the feature set with the MinMaxScaler.
2. To use exploratory data analysis (EDA) methods to examine the dataset, such as correlation analysis and heatmap visualisation.
3. To put into practice and train a number of machine learning models for classifying spam emails, such as:
    - Logistic Regression
    - Multi-layer Perceptron (MLP) Classifier
    - Random Forest Classifier
    - Decision Tree Classifier
    - K-Nearest Neighbors (KNN) Classifier
4. To create models that can distinguish between spam and non-spam emails, each model should be built up with the appropriate parameters and trained using the preprocessed Spambase dataset.
5. To assess and contrast each model's performance using a range of measures, including confusion matrix, F1-score, recall, accuracy, and precision.
6. To verify generalisability, evaluate the model's performance on the training and test datasets.
7. To evaluate model outputs, such as true positives, false positives, and other classification metrics, by creating and visualising confusion matrices and classification reports.
8. To maximise classification accuracy by modifying important model parameters, such as the number of neighbours in KNN, the learning rate in MLP, and the number of estimators in Random Forest.
9. To record all of the actions performed during the project, such as preparing the data, creating the model, assessing it, and optimising it.
10. To wrap up the project, the best spam detection model will be chosen,

implementation issues will be discussed, and any future enhancements or changes for more study will be suggested.

1.3 Project Outline:

Chapter 1: Introduction

This chapter discusses Introduction and Objectives of this project.

Chapter 2: Literature Review

This chapter discusses spam email and its impact, types of spam emails, Machine Learning, Algorithms, Related work. The five algorithms used in this project are also discussed in this chapter.

Chapter 3: Methodology

This chapter represents data collection, Feature Pre-Processing, Training and Testing, Models, Block diagram of workflow, Training a model, Performance Metrics.

Chapter 4: Results And Discussion

This chapter shows Performance Analysis of the algorithm. It includes all the model results. It also includes Performance Comparison and Analysis and Discussion.

Chapter 5: Conclusion

This chapter include Conclusion and Future Work.

CHAPTER 2
LITERATURE REVIEW

2.1 Spam Email and Its Impact

For the majority of Internet users, emails have emerged as a popular and significant communication tool in recent years. But the worst thing about email communication is spam, which is often referred to as unsolicited commercial or bulk emails. Spam and paper junk mail are frequently contrasted [4]. Unwanted, irrelevant, or unsolicited emails sent in mass to a significant number of recipients are referred to as spam. Although these emails are frequently used for business, they can also be used to spread harmful material, including malware or phishing scams. Spam emails are frequently sent without the recipient's consent and usually include no relevant or personalised content.

Key Impacts of Spam Emails:

- Wasted Time and Resources: Spam emails can overflow inboxes, making it difficult to go through or remove pointless correspondence. Both people and organisations have decreased productivity as a result of this.
- Security Threats: Phishing schemes, in which criminals pose as trustworthy companies in order to get private data, including credit card numbers, bank account information, and login passwords, are frequently carried out via spam emails. Spam emails may be used by cybercriminals to get personal data, which can result in fraud and identity theft.
- Financial Loss: Spam emails usually advertise items, investment possibilities, phoney employment offers, or fraudulent schemes. These frauds have the potential to cause financial damage for both individuals and corporations.If spam emails overwhelm an organization's communication infrastructure, lower staff productivity, or result in possible security breaches, the company may lose money.
- Damage to Reputation: The reputation of an organisation can be seriously harmed when spam emails pose as a business or brand (for example, phishing emails that pass for authentic correspondence). If consumers fall for frauds, they can stop believing in the brand.
- Decreased Efficiency in Email Communication: Users find it more difficult to concentrate on critical communications when they are inundated with spam emails. This lessens the overall efficacy of email communication and adds to information overload.
- Public Health and Safety Concerns: Spam emails on health or medical frauds have

the ability to propagate false information and jeopardise public health. Phishing emails may advertise untested therapies, phoney pharmaceuticals, or inaccurate health advice.

2.2 Spam Emails types

Spam Email can be categories in some different types. Table 2.1 explain those types of spam emails.

SI. No.	Type	Details
1	Phishing Emails	These emails are fraudulent attempts to get private data, such as credit card numbers, usernames, and passwords. They frequently appear to be authentic correspondence from reliable businesses.
2	Scam Emails	These include deceptive promises intended to fool recipients into transferring money or disclosing personal information, such as phoney investment possibilities, employment offers, or pyramid scams.
3	Commercial Spam	These emails are sent in mass with the intention of promoting goods or services. They are seen as unwelcome and obtrusive, even if they aren't always hostile.
4	Malware Spam	These emails include links or malicious attachments that, when clicked, infect the recipient's device with dangerous malware. Ransomware assaults, system compromise, or data theft might result from this.
5	Email Spoofing	When a spammer alters an email's "From" address to make it look as though it is from a reputable source, such as a bank or a well-known business, this is known as email spoofing. The intention is to trick the receiver into clicking on harmful links or divulging private information, including credit card numbers or passwords. Phishing attacks may result from this kind of spam.
6	Sweepstakes Winners	These emails, which frequently have no prior entries, claim that the receiver has won a reward in a lottery or contest. In order to claim the reward, they often want payment for taxes or fees or personal information (such as bank account data). The goal of these emails is to either get private information or deceive the receiver into giving money for a reward that doesn't exist.

Table 2.1: Some types of Spam Email

2.3 Machine Learning

Machine Learning (ML) is a subset of Artificial Intelligence (AI) that focusses on creating systems that can learn from and make data-driven judgements without being explicitly programmed. Machine learning has a significant impact on autonomous vehicles, drones, and robots, improving their adaptability in dynamic environments. This method is a breakthrough in which machines learn from data examples to provide correct results, and it is strongly related to data mining and data science. computer learning allows a computer to learn from data, improve performance via experience, and anticipate things without having to be explicitly programmed. Machine learning algorithms produce a mathematical model that, without being explicitly coded, assists in generating predictions or judgements based on sample historical data, often known as training data. Machine learning integrates statistics and computer science to create prediction models. Machine learning involves the creation or use of algorithms that learn from past data. The performance will improve in proportion to the amount of information we offer.

The machine learning method requires messages that have been correctly pre-classified, but it does not require pre-established rules. The training dataset required to develop the model's unique learning algorithm is constructed from such communications. Therefore, a machine learning method to spam email classification may be used, where a computer software learns from the input data and applies that learning to categorise fresh observations [5].

2.3.1 The operation of machine learning

A subfield of artificial intelligence (AI) called machine learning (ML) gives computers the ability to learn from data and make judgements or predictions without explicit programming. This is a basic explanation of how it operates:

Data Collection: Collecting a dataset that reflects the issue you wish to address is the first step since machine learning models learn from data. To classify spam emails, for instance, I would gather a dataset of emails that have been classified as either "spam" or "not spam."

Data Preprocessing: Raw data must be prepared and cleansed since it is frequently untidy. Among the steps are:

- Eliminating duplicate or missing values.

- Transforming textual information into numerical form (for example, by employing TF-IDF or word embeddings in emails).
- Dividing data into groups for testing and training.

Feature Extraction and Selection: In many situations, the gathered data may include a large number of features or qualities. Finding the most instructive and pertinent characteristics that contribute to the learning job is the goal of feature extraction and selection. This enhances the efficacy and efficiency of the learning process while lowering the dimensionality of the data.

Choosing Machine Learning Models: There are several machine learning models for various jobs. Common methods for classifying spam emails include the MLP Classifier (a neural network-based classification system), Random Forest (ensemble learning for enhanced accuracy), and Logistic Regression (for binary classification).

Training the Model: The labelled data—input characteristics and matching outputs—is fed into the model during training. By modifying internal settings to reduce mistakes, it discovers patterns. For instance, the model discovers that emails with phrases like "click here" or "free money" are more likely to be spam.

Evaluating the Model: To assess accuracy, precision, recall, and other performance measures, the model is evaluated on unknown data after training; if performance is subpar, hyperparameter modification or the use of an alternative model may be required.

Making Predictions: Based on patterns it has learnt, the model can identify new, unseen emails as either "spam" or "not spam" after training.

Deployment and Improvement: Applications (like email filters) can include the learnt model. To increase accuracy over time, it may be retrained using fresh data.

2.3.2 Machine Learning Method

Machine learning methods are grouped into three types: supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning: Supervised learning is among the most fundamental forms of machine learning. In this instance, labelled data is used to train the machine learning algorithm.

Although accurate data labelling is necessary for this approach to function, supervised learning may be highly successful when used appropriately. In supervised learning, a brief training dataset is provided to the machine learning algorithm [1]. K-nearest

neighbours (KNN), logistic regression, decision trees, random forests, and multi-layer perceptrons (MLP) are a few examples. This technique is frequently applied to jobs such as classifying spam emails, which are classified as either "spam" or "not spam."

Unsupervised learning: Unsupervised learning works with unlabelled input, which means that the model finds structures and patterns without producing predetermined results. This group includes dimensionality reduction methods like Principal Component Analysis (PCA), hierarchical clustering, and clustering algorithms like K-means. It is frequently employed for pattern identification, anomaly detection, and client segmentation.

Reinforcement learning: An agent interacting with its surroundings and learning through rewards and punishments is the foundation of reinforcement learning. Creating a plan that optimises gains over time is the aim. Applications including autonomous systems, robotics, and gaming leverage algorithms like Q-learning, deep Q-networks (DQN), and policy gradient approaches.

Other variations include **semi-supervised learning**, which combines a small amount of labeled data with a large amount of unlabeled data, and **self-supervised learning**, where the model generates its own labels from data, often used in deep learning applications like natural language processing.

2.4 Algorithms

In my project, five algorithms are employed for the classifications.

2.4.1 Logistic Regression

One "supervised machine learning" technique for modelling the likelihood of a certain class or occurrence is logistic regression. It is applied when the result is binary or dichotomous and the data is linearly separable. A sigmoid function is used to compute the probability.

Let's look at an example where the data includes n characteristics.

This line may be represented by the equation

$z = b_0+b_1 x_1+b_2 x_2+b_3 x_3.....+b_n x_n.$

We must fit a line for the provided data. z = odds in this case

The formula for calculating odds is typically

odds = p (event occurring)/p (event not occurring)

The term "logistic regression" comes from the fact that a sigmoid function is a particular type of logistic function. To get a continuous probability between 0 and 1, the logarithm of odds is computed and entered into the sigmoid function.

The logarithm of odds may be computed as

log(odds) = dot (features, coefficients) + intercept.

The sigmoid function uses these log_odds to calculate probability.

h(z)=1/(1+e^(-z))

The sigmoid function yields an integer between 0 and 1, which is used to identify the class to which the sample belongs. Generally speaking, 0.5 is the threshold below which a response is regarded as NO, and 0.5 or above is regarded as a YES. However, the boundary may be changed to suit the needs [6].

2.4.2 Random Forest Classifier

One popular machine learning algorithm that employs the supervised learning approach is Random Forest. It may be applied to both regression and classification problems in machine learning. Its foundation is ensemble learning, a technique that combines several classifiers to tackle a challenging issue and improve the model's performance [1]. One type of supervised ensemble method is the Random Forest classifier. Several random decision trees make up a random forest. The trees are inherently random in two ways. A random sampling of the original data is used as the foundation for each tree. Second, to get the optimal split, a subset of characteristics is chosen at random from each tree node.

Decision Tree:

The decision tree is a feature-only classification technique. With the best information gain, the tree repeatedly divides the data based on a feature. This procedure keeps on till the knowledge acquired stays the same. After that, each aspect of the unidentified data is assessed until it is categorised. Tree trimming strategies are employed to decrease data overfitting and increase accuracy. On subsets of data, several decision trees are constructed; the outcome produced by the majority of the trees is regarded as the final outcome. Iterative techniques are used to calculate the number of trees that must be produced depending on correctness and other parameters. Although condition-based data is the primary use for random forest classifiers, text may also be utilised if it is transformed to numerical form [6].

2.4.3 K-Nearest Neighbors

KNN is an algorithm for classification. It falls within the category of supervised algorithms. It is assumed that every data point is located in an n-dimensional space. The category of current data is then established based on the majority of neighbours.

The distance between two locations is calculated using the Euclidian distance.

The formula for calculating the distance between two points is

d=√(〖(x2-x1)〗 ^2+ 〖(y2-y1)〗 ^2)

The unknown point's distances from every other point are computed. The K nearest neighbours are identified based on the K supplied. The unknown data category is chosen to be the one that most of the neighbours fall into.

The plot can be seen if there are up to three features in the data. Because it must calculate the distance to every point in order to find the nearest neighbours to the specified point, it is somewhat sluggish when compared to other distance-based algorithms like SVM [6].

2.4.4 Decision Tree Classifier

Although it may be applied to both classification and regression issues, a decision tree classifier is a machine learning technique that is mostly used to solve classification problems. This classifier is tree-structured, with internal nodes standing in for a dataset's characteristics, branches for the decision rules, and each leaf node for the result. The Decision Node and the Leaf Node are the two nodes that make up a decision tree. Leaf nodes are the results of those decisions and do not have any more branches, whereas decision nodes are utilised to make any decision and have numerous branches. The characteristics of the provided dataset are used to make choices or run tests. It is a graphical depiction that shows every option for solving a problem or choice under certain circumstances. The reason it is named a decision tree is that, like a tree, it begins with the root node, which grows on additional branches to form a structure like a tree.We employ the Classification and Regression Tree algorithm, or CART algorithm, to construct a tree.A decision tree only poses an inquiry and divides the tree into subtrees according to the response (Yes/No).

2.4.5 Multi-Layer Perceptron (MLP) Classifier

In machine learning, a "multi-layer perceptron (MLP) classifier" is a kind of artificial neural network used for classification tasks. The MLP's neurones usually employ nonlinear activation functions, which enables the network to recognise intricate patterns in input. Because MLPs are capable of learning nonlinear correlations in data, they are important models for machine learning tasks including pattern recognition, regression, and classification. The input layer, one or more hidden layers, and the output layer are the three primary layer types that make up an MLP's architecture. Artificial neurones, which are information-processing computing units, make up each layer.The raw data features are sent to the following layer from the input layer. Neurones in the buried layers use weighted connections to make changes to the data. A dense neural network is one in which every neurone in one layer is linked to every other neurone in the layer below. The weights assigned to these connections dictate how strongly one neurone influences another. In order to reduce prediction errors, these weights are adjusted during the MLP learning process.

By introducing non-linearity through activation functions, the neurones in the hidden and output layers enable the model to recognise intricate patterns. The rectified linear unit (ReLU), which is frequently employed because it may alleviate the vanishing gradient issue, as well as the sigmoid and softmax functions, which are frequently employed in classification tasks, are examples of common activation functions. The model's capacity to learn efficiently is greatly influenced by the activation function selection.

Backpropagation is a crucial neural network training process used by MLP. In order to update the weights, backpropagation entails computing the difference between the expected and actual output and then sending this error backward through the network. Gradient descent-based algorithms, like Adam or stochastic gradient descent (SGD), are used in the optimisation process to modify the weights in a way that minimises error. One crucial hyperparameter that regulates the step size of these updates is the learning rate.

2.5 Related work

The difficulty of classifying spam emails is neither new nor easy. Numerous studies have been conducted, and numerous practical approaches have been put forth.

i. S. T. Kawsar (2022) conducted a study on spam mail detection using supervised learning techniques and demonstrated that machine learning models could effectively classify spam emails with high accuracy [1].

ii. Mangena Venu Madhavan et al. (2021) compared different spam detection algorithms and found that support vector machines (SVM) and ensemble models provided superior performance in email spam classification [2].

iii. M. Jazzar (2021) analyzed multiple machine learning classifiers and concluded that logistic regression, random forest, and naïve Bayes performed efficiently in detecting spam emails, with varying success based on dataset characteristics [3].

iv. D. Mallampati (2018) proposed an efficient spam filtering method using supervised learning, highlighting the role of feature selection techniques in improving model accuracy [4].

v. Z. L. X. Sutta et al. (2020) conducted a study on various machine learning algorithms and identified that deep learning models, particularly recurrent neural networks (RNNs), achieved high accuracy but required extensive computational resources [5].

vi. V. S. Ganesh (2022) explored the use of machine learning models in combination with natural language processing (NLP) techniques, demonstrating improvements in spam detection accuracy [6].

vii. Asma Bibi et al. (2020) developed a spam mail scanning system that leveraged advanced machine learning algorithms, emphasizing the importance of dataset quality in classification performance [7].

viii. P. Sharma (2018) examined the effectiveness of machine learning models in detecting spam emails and concluded that ensemble learning techniques enhanced accuracy compared to standalone classifiers [8].

ix. M. H. Alsuwit (2024) investigated modern email spam classification techniques, incorporating feature engineering and deep learning models to improve classification robustness [9].

x. Prachi Bhatnagar et al. (2023) conducted a comprehensive review of email spam classification methodologies, discussing recent advancements in hybrid models and their impact on spam detection accuracy [10]

CHAPTER 3

METHODOLOGY

3.1 Data Collection

The dataset used for this spam email classification project was obtained from a publicly available database. This dataset consists of a total of 4,601 emails, categorized into spam and ham (non-spam) emails. It contains essential features extracted from the email text, including word frequency, character frequency, and special character usage, which help in distinguishing between spam and legitimate emails.

The dataset is labeled using binary values, where spam emails are represented by '1' and ham emails by '0'. This structured labeling facilitates the application of supervised machine learning algorithms for effective spam detection. The dataset is well-balanced, ensuring that the model learns patterns from both spam and ham emails efficiently.

For training and evaluation, the dataset was split into training and testing sets. A 67%-33% split was applied, resulting in 3,082 emails used for training and 1,519 emails used for testing. This division ensures that the model is trained on a substantial amount of data while still having a sufficient number of samples for evaluation.

The dataset comprises 57 features that play a crucial role in the classification process. These features include word occurrence frequencies, capital letter usage, and other statistical properties that help in identifying spam emails more accurately.

Description	**Count**
Total number of emails	4,601
Total number of spam emails	1,813
Total number of ham emails	2,788
Total number of training emails	3,082
Total number of testing emails	1,519

Table 3.1: Dataset Statistics

This dataset provides a comprehensive resource for building and evaluating machine learning models for spam detection, aiding in the automation of email filtering and enhancing cybersecurity measures.

3.2 Feature Pre-Processing

Feature pre-processing is a crucial step in preparing the dataset for training machine learning models. In this project, several techniques were applied to ensure that the extracted features contribute effectively to the classification of spam and ham emails.

First, missing values were handled by ensuring that all email entries contained valid numerical representations for their respective features. If any missing data were present, appropriate imputation techniques were used to replace them.

Second, feature scaling was applied to normalize the dataset. Since the dataset contains frequency-based numerical values, Min-Max scaling or standardization was performed to ensure that all features have a similar scale, preventing any single feature from dominating the learning process.

Third, feature selection was conducted to identify the most relevant attributes for spam detection. Techniques such as correlation analysis and mutual information were used to remove redundant or irrelevant features, enhancing the model's efficiency and performance.

Finally, categorical encoding was not necessary in this case since all features are numerical. However, the dataset was checked for any anomalies, and necessary data transformations were applied to improve model interpretability and effectiveness.

By performing these pre-processing steps, the dataset was optimized for machine learning models, ensuring better generalization and improved spam classification accuracy.

3.3 Training and Testing

To develop an effective spam email classification model, the dataset is divided into two subsets: **training data** and **testing data**. The training data is used to train the machine learning models, allowing them to learn patterns that distinguish spam from non-spam emails. The testing data is used to evaluate the model’s performance on unseen emails, ensuring its ability to generalize to new data.

Here are the dataset sizes after performing the train-test split:

- Total number of rows in the dataset: 4601
- Number of rows in the training set: 3450
- Number of rows in the test set: 1151

3.4 Models

In this project we use five machine learning algorithms. These are: Logistic Regression, Random Forest Classifier, Decision Tree Classifier, K-Nearest Neighbours Classifier, Multi- Layer Perceptron (MLP) Classifier. These models work in there own way.

3.5 Block diagram of workflow

The block diagram of workflow provides a visual representation of the steps involved in any project. It typically includes some important phases. These are discussed in below:

3.5.1 Data Collection

The first step in spam email classification is gathering relevant datasets. The dataset should include both spam and non-spam (ham) emails to ensure a balanced and representative collection. Having a diverse dataset helps in building a robust model that can generalize well to real-world spam detection scenarios. A large and varied dataset minimizes biases and improves the effectiveness of the classification model. Here dataset is taken from Kaggle.

3.5.2 Data Preprocessing

Before being utilized for training, the dataset has to be cleaned and prepped after collection. To standardize and improve the raw data, preprocessing entails a number of procedures. To preserve data integrity, duplicate emails and missing values are first eliminated. After that, text-processing methods including lemmatization, stemming, tokenization, and stopword removal are used. While stopword removal eliminates frequent terms like "the," "is," and "and" that don't help with categorization, tokenization splits emails down into individual words. In order to ensure that multiple versions of the same word (such as "running" and "run") are regarded as one, stemming and lemmatization transform words into their root forms.

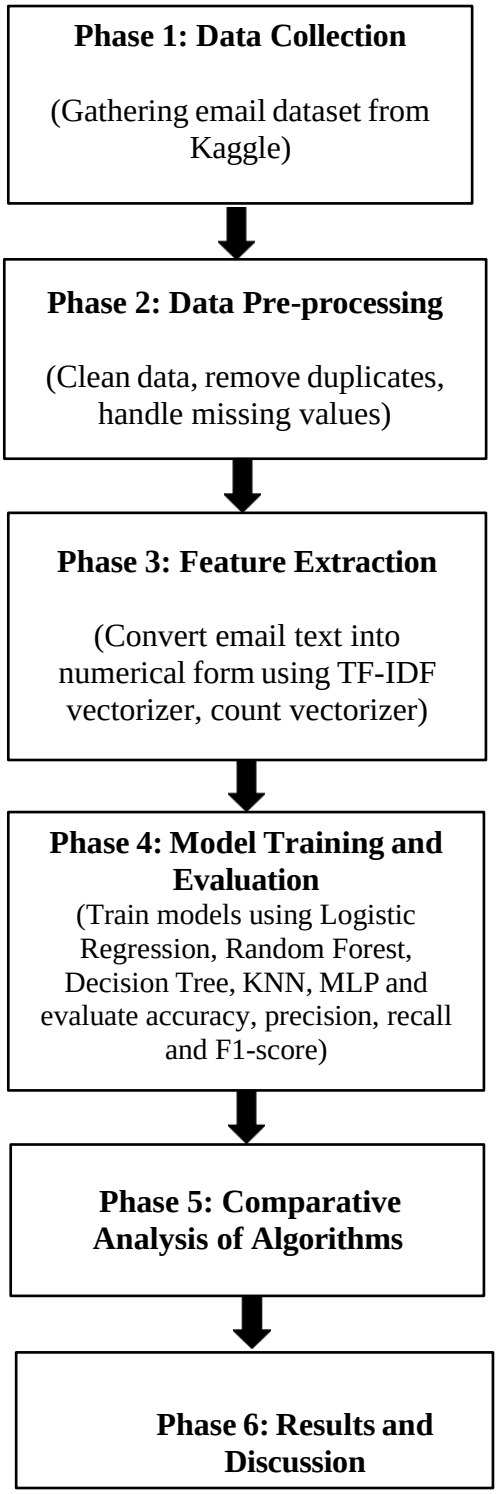

Figure 3.1: Block diagram of workflow

3.5.3 Feature Extraction

Email content must be converted into numerical representations since machine learning models are unable to process raw text directly. Converting textual data into structured input that models can comprehend is known as feature extraction. TF-IDF (Term Frequency- Inverse Document Frequency) is a widely used approach that assesses the significance of individual words in an email in relation to the full dataset. Count Vectorization is another well-liked technique that turns text into a matrix of word counts. Semantic meaning in text may also be captured by more sophisticated methods, including word embeddings like Word2Vec or FastText. By using word patterns, frequency, and correlations, these techniques assist models in distinguishing between spam and non-spam emails.

3.5.4 Model Training & Evaluation

The data is used to train various supervised machine learning models once it has been preprocessed and converted into numerical features. To find the best classification algorithm for spam detection, a number of them are investigated. Among the models that are trained and evaluated are Multi-Layer Perceptron (MLP), K-Nearest Neighbors (KNN), Random Forest, Decision Tree, and Logistic Regression. To improve performance, more sophisticated strategies like Adaboost, Bagging, Gradient Boosting, and XGBoost can also be used. The efficacy of the trained models is assessed using a number of measures, including accuracy, precision, recall, and F1-score. In order to reduce false positives and false negatives, an effective spam classifier should balance precision and recall in addition to having a high accuracy.

3.5.5 Comparative Analysis

To determine the optimal strategy, it is crucial to evaluate the performance of several machine learning models once they have been trained. This examination takes into account a number of variables, such as model interpretability, computing efficiency, and accuracy. Certain models could be quite accurate, but they need a lot of processing power, which makes them unsuitable for spam detection in real time. While some may be quicker, they may not be as good at distinguishing between spam and authentic emails. In order to ensure optimal performance for real-world applications, the comparison analysis assists in identifying which model offers the best balance between accuracy, speed, and generalizability.

3.5.6 Results & Discussion

The last stage is analyzing the experiment results and summarizing the findings. The model that performs the best is determined, and the factors that contributed to its success are examined. There is discussion of difficulties that arise during categorization, such managing unbalanced data or changing spam trends. Potential enhancements are also proposed, such as enhancing spam detection with transformers (e.g., BERT) or deep learning models like LSTMs. The findings open the door for further developments in the categorization of email spam by offering insightful information about the advantages and disadvantages of various machine learning techniques.

3.6 Training a model

In machine learning, training a model is teaching an algorithm to identify patterns in data so that it can provide precise classifications or predictions. The first step in the process is to collect and prepare the dataset, making sure information is clear, organized, and indicative of the issue at hand.

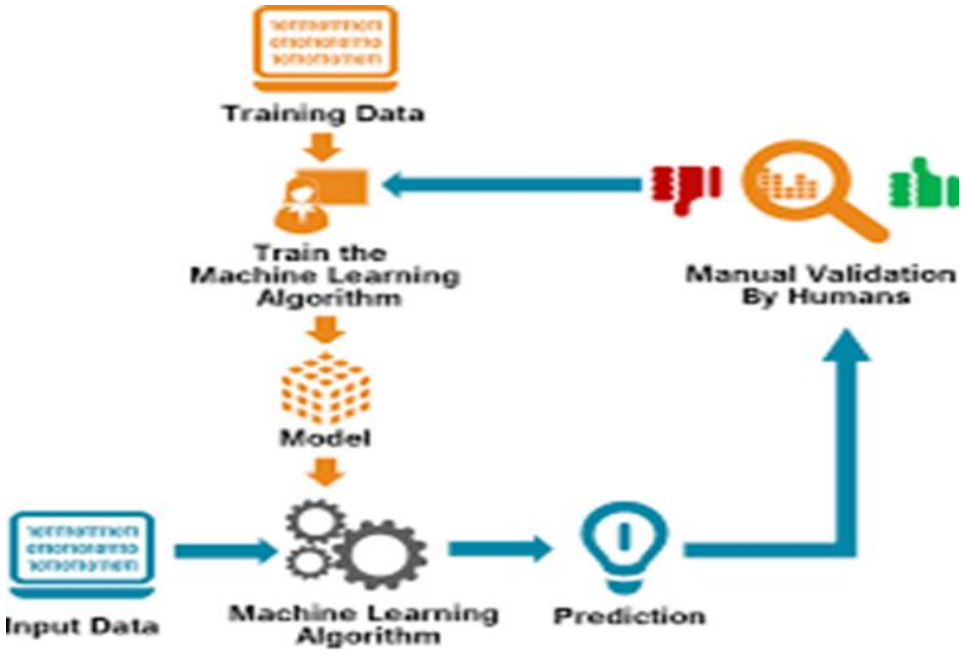

Figure 3.2: Procedure of training a model

After then, the dataset is divided into training and testing sets. The model is trained on the training set, and its performance is assessed on the testing set. When the data is prepared, feature extraction methods convert the unprocessed data into numerical

representations that can be processed by a machine learning algorithm. In order to classify spam, email content must be transformed into structured representations utilizing techniques like Count Vectorization and TF-IDF. These methods aid the model in comprehending the patterns of word significance and frequency that differentiate spam from non-spam communications.

Ultimately, a model's efficacy is confirmed by testing it on actual data once it has been trained and tuned. If it works, the learned model may be used to make predictions in real time and categorize newly received emails as either spam or ham. A machine learning model must be continuously improved during the iterative training process in order to adjust to new patterns and increase accuracy over time.

3.7 Performance Metrics

The performance metrics in machine learning refers to the set of measures used to evaluate the effectiveness and accuracy of a model's predictions. These metrics help quantify how well a model performs in classifying or predicting outcomes, and they are particularly essential when comparing different models or selecting the most appropriate model for a given task.

For classification tasks in my spam email classification project, the performance matrix provides insights into how accurately the model is identifying true positives (correctly predicted spam), true negatives (correctly predicted non-spam), false positives (incorrectly predicted spam), and false negatives (incorrectly predicted non-spam). These values are typically represented in a confusion matrix, which serves as the foundation for calculating various performance metrics.

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Table 3.2: Confusion Matrix

The confusion matrix provides the following values:

- True Positives (TP): The number of correctly predicted positive instances.
- False Positives (FP): The number of incorrectly predicted negative instances as positive.
- False Negatives (FN): The number of incorrectly predicted positive instances

as negative.

- True Negatives (TN): The number of correctly predicted negative instances.

3.7.1 Accuracy

Accuracy is one of the most basic yet important performance metrics derived from a confusion matrix. It measures the proportion of correct predictions (both true positives and true negatives) out of the total number of predictions made by the model.

Formula:

Accuracy = (TP+TN) / (TP+TN+FP+FN)

Formula: Accuracy = (TP+TN) / (TP+TN+FP+FN)

• TP + TN: Correct predictions.

• TP + TN + FP + FN: Total predictions.

3.7.2 Precision

We divide the total number of successfully organized positive instances by the absolute number of expected positive examples to obtain the precision estimate. High accuracy indicates that a model with a low number of FP is unquestionably positive. class comprehension of the information names using the classifier's positive names [7].

Formula: Precision = TP / (TP+FP)

• TP: Correctly predicted positive instances.

• FP: Incorrectly predicted negative instances as positive.

3.7.3 Recall

The ratio of the difference between the all-out number of positively grouped models and the all-out number of positive models is known as recall. A high recall indicates that the class is well-received (minimal number of FN) [7].

Formula:

Recall = TP / (TP+FN)

• TP: Correctly predicted positive instances.

• FN: Incorrectly predicted positive instances as negative.

3.7.4 F1-score

The F1-score is a performance metric derived from the confusion matrix that balances precision and recall. It is particularly useful when dealing with imbalanced datasets, where accuracy alone can be misleading.

Formula:

F1-score = 2*(Precision × Recall) / (Precision + Recall)

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Performance Analysis

In this part, we use the Kaggle dataset to examine how well our chosen machine learning models perform in the classification of spam emails. The assessment is carried out in several experimental configurations. Five machine learning models Logistic Regression, Random Forest Classifier, Decision Tree Classifier, K-Nearest Neighbors Classifier, and Multi-Layer Perceptron (MLP) Classifier are trained and evaluated in order to determine which one is better for spam identification. Performance metrics like F1-score, recall, accuracy, and precision are used to compare the models. According to the results, it has the highest accuracy, making it a trustworthy automatic spam filtering solution.

4.1.1 Logistic Regression Model

Confusion Matrix is:

[[879 38]

[124 478]]

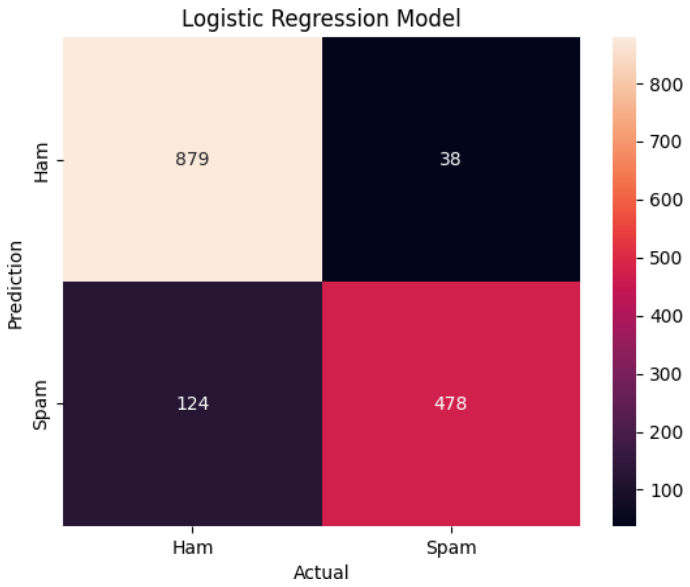

Figure 4.1: Confusion Matrix for Logistic Regression Model

Classification Report is:

	precision	recall	f1-score	support
0	0.88	0.96	0.92	917
1	0.93	0.79	0.86	602
accuracy			0.89	1519
macro avg	0.90	0.88	0.89	1519
weighted avg	0.90	0.89	0.89	1519

4.1.2 Random Forest Classifier Model

Confusion Matrix is:

[[897 20]

[39 563]]

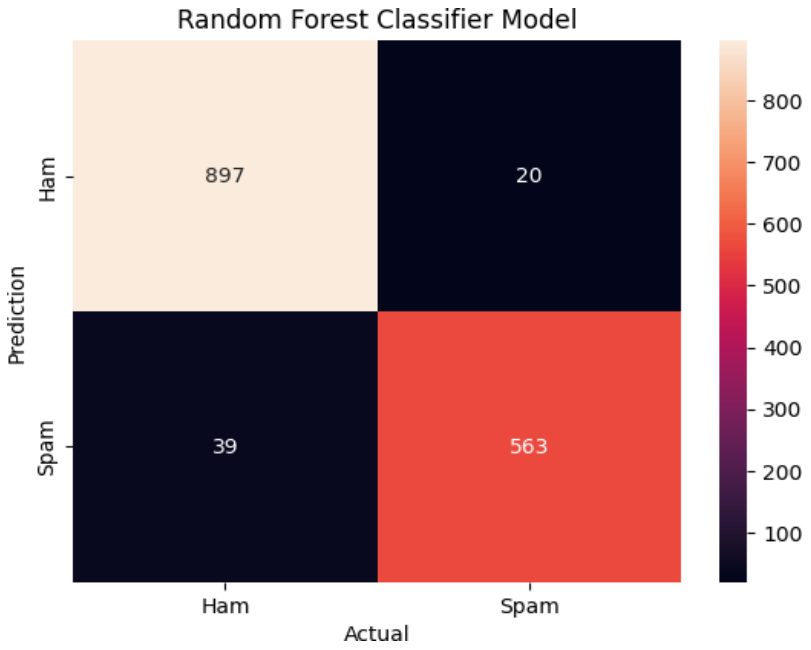

Figure 4.2: Confusion Matrix for Random Forest Classifier Model

Classification Report is:

	precision	recall	f1-score	support
0	0.96	0.98	0.97	917
1	0.97	0.94	0.95	602
accuracy			0.96	1519
macro avg	0.96	0.96	0.96	1519
weighted avg	0.96	0.96	0.96	1519

4.1.3 Decision Tree Classifier Model

Confusion Matrix is:

[[848 69]

[65 537]]

Figure 4.3: Confusion Matrix for Decision Tree Classifier Model

Classification Report is:

	precision	recall	f1-score	support
0	0.93	0.92	0.93	917
1	0.89	0.89	0.89	602
accuracy			0.91	1519
macro avg	0.91	0.91	0.91	1519
weighted avg	0.91	0.91	0.91	1519

4.1.4 K-Nearest Neighbors Classifier Model

Confusion Matrix is:

[[864 53]

[77 525]]

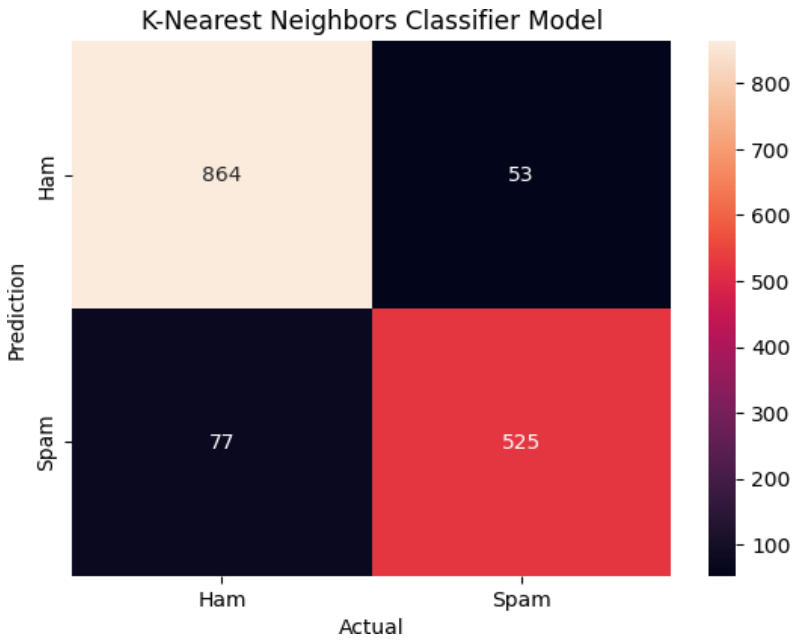

Figure 4.4: Confusion Matrix for KNN Classifier Model

Classification Report is:

	precision	recall	f1-score	support
0	0.92	0.94	0.93	917
1	0.91	0.87	0.89	602
accuracy			0.91	1519
macro avg	0.91	0.91	0.91	1519
weighted avg	0.91	0.91	0.91	1519

4.1.5 Multi-Layer Perceptron (MLP) Classifier Model

Confusion Matrix is:

[[872 45]

[38 564]]

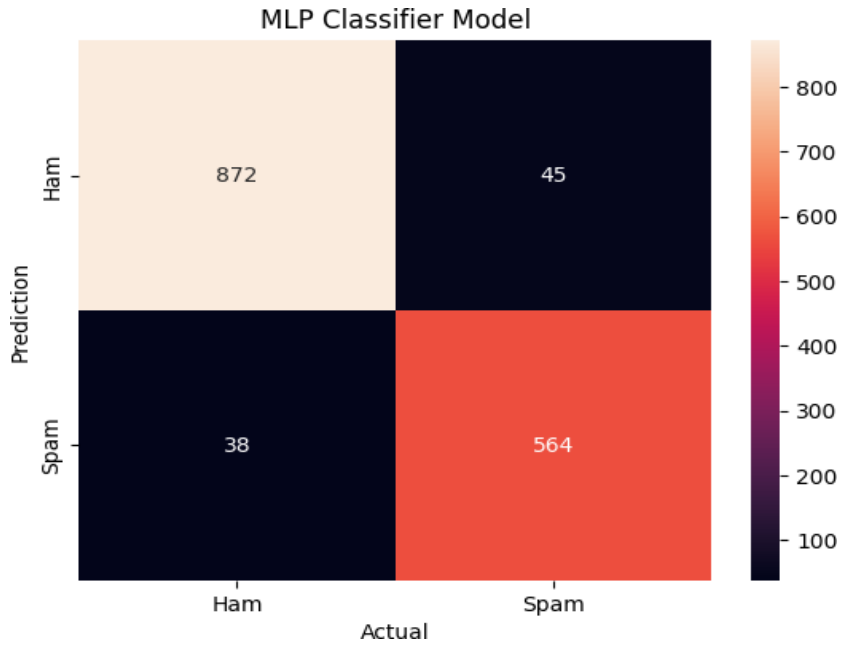

Figure 4.5: Confusion Matrix for MLP Classifier Model

Classification Report is:

	precision	recall	f1-score	support
0	0.96	0.95	0.95	917
1	0.93	0.94	0.93	602
accuracy			0.95	1519
macro avg	0.94	0.94	0.94	1519
weighted avg	0.95	0.95	0.95	1519

4.2 Comparison

The evaluation of different machine learning models for spam classification is based on key performance metrics: accuracy, precision, recall, and F1-score. Each of these metrics provides insights into the effectiveness of the models in distinguishing spam from legitimate emails.

ML Models	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Logistic Regression	89.34	90.14	87.63	88.54
Random Forest	96.12	96.2	95.67	95.92
Decision Tree	91.18	90.75	90.84	90.79
KNN	91.44	91.32	90.71	90.99
MLP	94.54	94.22	94.39	94.3

Table 4.1: Experimental Results for ML Models Accuracy, Precision, Recall, and F1-score

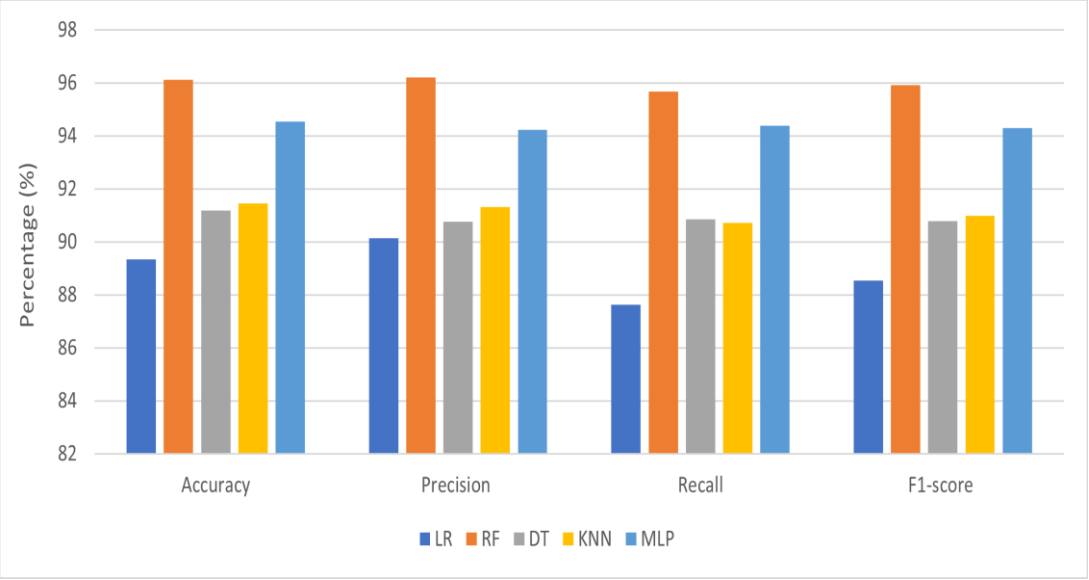

Figure 4.6: Graphical Representation of Experimental Results

To evaluate the effectiveness of different machine learning models for spam email classification, we compared the performance of Logistic Regression (LR), Random Forest (RF), Decision Tree (DT), K-Nearest Neighbors (KNN), and Multi-Layer Perceptron (MLP). The evaluation metrics used include Accuracy, Precision, Recall, and F1-score.

From the results illustrated in **Figure 4.6**, it is evident that the **Random Forest (RF) classifier outperforms all other models** across all metrics, achieving the highest accuracy (96.12%) and F1-score (95.92%). This indicates that RF provides the most balanced performance in terms of correctly classifying spam and non-spam emails while minimizing false positives and false negatives.

The **MLP classifier** follows closely behind RF, achieving an accuracy of **94.54%** and an F1-score of **94.3%**, demonstrating its effectiveness in learning complex patterns in email text. **Decision Tree (91.18%)** and **KNN (91.44%)** perform moderately well, with comparable scores across different metrics.

On the other hand, **Logistic Regression (LR)** exhibits the lowest performance, with an accuracy of **89.34%** and an F1-score of **88.54%**. This suggests that while LR is a simple and interpretable model, it may not be as effective as more complex models in handling the intricacies of spam classification.

Overall, **Random Forest** emerges as the best-performing model for spam detection, while MLP also proves to be a strong alternative. These results highlight the importance of selecting robust classifiers that effectively balance precision and recall to minimize misclassification in spam email classification

4.3 Discussion

In this study, we evaluated five machine learning models for spam email classification: Logistic Regression, Random Forest Classifier, Decision Tree Classifier, K-Nearest Neighbors (KNN) Classifier, and Multi-Layer Perceptron (MLP) Classifier. The models were assessed based on their accuracy, precision, recall, and F1-score to determine their effectiveness in distinguishing spam from ham emails. Among the evaluated models, the Random Forest Classifier achieved the highest accuracy of **96.12%**, followed closely by the MLP Classifier at **94.54%**. Both models demonstrated strong generalization capabilities, suggesting their robustness in handling email spam classification. The Accuracy of Decision Tree is **91.18%**and KNN Classifiers is **91.44%**, performing similarly in terms of classification

effectiveness. Logistic Regression, while still effective, had the lowest accuracy at **89.34%**, indicating it may not capture the complexity of spam detection as effectively as the other models. Overall, the results indicate that the **Random Forest and MLP Classifiers** are the most promising models for spam classification, offering high accuracy and balanced performance.

CHAPTER 5
CONCLUSION

5.1 Conclusion

In today's age of technology and communication, spam is one of the most difficult and problematic internet problems. Spam emails continue to be a significant challenge in digital communication, affecting individuals and organizations by flooding inboxes with unwanted and potentially harmful messages. Spammers abuse this communication tool by creating spam emails, which impacts businesses and a large number of email users [8]. In this study, we implemented a machine learning-based spam email classification system using Logistic Regression, Random Forest Classifier, Decision Tree Classifier, K-Nearest Neighbors Classifier, and Multi-Layer Perceptron (MLP).

Our experimental results demonstrated that [mention the best-performing model achieved the highest accuracy, highlighting its effectiveness in distinguishing spam from legitimate emails. The comparative analysis of different models provided valuable insights into their strengths and limitations, reinforcing the importance of selecting the right classification algorithm for spam detection.

5.2 Future Work

- One key area for future work is the exploration of deep learning techniques, such as Long Short-Term Memory (LSTM) networks and Transformer-based models (e.g., BERT). These models are known for their ability to understand contextual relationships within text, which could lead to higher accuracy in spam detection.
- Another potential enhancement is the use of hybrid models, where multiple classifiers are combined to leverage their strengths. For example, an ensemble approach integrating Random Forest with deep learning models may yield better results by balancing interpretability and accuracy. Additionally, hyperparameter tuning and feature selection techniques can be further optimized to enhance model performance.
- The study can also be extended by incorporating real-time spam detection systems, where models are continuously updated with new spam patterns. Adaptive learning techniques and online learning algorithms can be employed to ensure the classifier remains effective as spammers develop new techniques to bypass detection systems.

- Furthermore, expanding the dataset by including multilingual spam emails or spam messages from social media platforms could improve the robustness of the models. This would help create a more generalized spam detection system capable of handling diverse types of spam messages across different communication channels.
- Lastly, ethical considerations and adversarial attacks on spam classifiers should be explored. Attackers may attempt to manipulate machine learning models by introducing adversarial examples. Future research can focus on making spam classification systems more resilient to such attacks by incorporating adversarial training and explainable AI techniques.

Overall, future work in this domain should focus on deep learning, hybrid models, real- time detection, multilingual datasets, and security against adversarial attacks to build a more robust and adaptable spam detection system.

REFERENCES

[1] S. T. K. (20PCS010), "SPAM MAIL DETECTION USING SUPERVISED," Avinashilingam Institute for Home Science and Higher Education for , 2022.

[2] S. P. P. U. T. M. Mangena Venu Madhavan1, " Comparative Analysis of Detection of Email Spam With the," *IOP Conference Series:*, 1022.

[3] M. Jazzar, "Evaluation of Machine Learning Techniques for," *I. J. Education and Management Engineering*, no. 4, pp. 35-42, 2021.

[4] D. Mallampati, "An Efficient Spam Filtering using Supervised Machine Learning," *International Journal of Scientific Research in*, vol. 6, no. 2, pp. 33-37, 2018.

[5] Z. L. X. Z. N. Sutta, " A Study of Machine Learning Algorithms," *EPiCSeries in Computing*, vol. 69, pp. 170-179, 2020.

[6] V. S. GANESH, "Spam Detection using Machine Learning and Natural," *SATHYABAMA*, 2022.

[7] R. L. S. K. W. A. R. A. S. T. Asma Bibi1, "Spam Mail Scanning Using Machine Learning Algorithm," *Journal of Computers*, vol. 15, no. 2, pp. 73-84, 2020.

[8] P. Sharma1, "Machine Learning based Spam E-Mail Detection," *nternational Journal of Intelligent Engineering and Systems*, vol. 11, no. 3, 2018.

[9] M. H. Alsuwit, "Advancing Email Spam Classification using," *Engineering, Technology & Applied Science Research* , vol. 14, no. 4, 2024.

[10] S. D. Prachi Bhatnagar*1, "A Comprehensive Review on Email Spam Classification with," *International Journal of Scientific Research in Computer Science, Engineering and* , vol. 9, no. 10, pp. 283-288 , 2023.