Journal of Global Research in Electronics and Communication
Volume 1, No. 10, October 2025
Available Online at: www.jgrec.info
RESEARCH PAPER

# A Lightweight and Robust SMS Spam Filtering Model for Mobile Networks

Mr. Ram Pratap Singh
Department of Computer Science and Engineering,
Lakshmi Narain College of Technology
Bhopal
rampratap@lnct.ac.in

***Abstract*—Mobile messaging has skyrocketed with the proliferation of mobile users, which has brought about an upsurge in SMS (Short Message Service) spam. Unwanted and sometimes dangerous spam text messages are a major obstacle to mobile communication. Using the 5,574 tagged messages (ham or spam) from the SMS Spam Collection dataset, this study aims to detect spam via machine learning. Preprocessing the raw text data involves stemming, removing special characters, lowercasing, and tokenizing. Then, features are extracted using TF-IDF and dimensionality is reduced using Principal Component Analysis (PCA). A few performance metrics are utilized to assess KNN and NB, two classification models, including accuracy, precision, recall, F1-score, ROC curve, and confusion matrix. When it came to spam detection, KNN had the best accuracy (95.3% and 98.5%, respectively), whereas NB was the best in precision (97.6%, minimizing false positives). Examining KNN and NB alongside other classifiers, like Decision Tree (DT) and Random Forest (RF), reveals that they outperform them. The study concludes that both models are highly effective for SMS spam filtering, with KNN preferred in high-recall applications and NB in precision-critical scenarios, making them suitable for mobile and resource-constrained environments.**

***Keywords*—SMS Spam Detection, Machine Learning, Text Classification, Spam Filtering, Mobile Communication.***

## I. INTRODUCTION

Spam refers to mass electronic messages that are neither intended nor requested and are sent to multiple recipients at once. A multitude of factors contribute to the alarming volume of electronic unsolicited commercial messages. The message's persuasiveness, the low probability of receiving responses from some unaware receivers, the message's dependability (because it reaches the mobile phone user), and the cheapness of bulk SMS plans are all factors to consider. A lot of work goes into detecting and preventing mobile SMS spam [1]. Many problems and their solutions with previous email spam filtering and detection situations have been carried over to it. The pervasiveness of SMS spam has made it a major pain point for mobile customers. Lost productivity, increased network bandwidth utilization, administrative headaches, and invasions of privacy are all major costs [2][3]. Similar to e-mail spam, mobile SMS spam annoys those who use mobile phones and introduces additional social frictions to mobile handsets.

Sending spam SMS texts is cheaper than sending spam email, so it hasn't been seen as a big issue in Western countries. Yet, short message service (SMS) messaging is all the rage in Europe. Almost everyone over the age of 15 has a cell phone, and the typical user sends out 10 SMS messages daily. Unfortunately, this makes text messages an easy target for fraudsters. Also, free SMS messaging systems in countries like Russia are being hacked by botnets of zombie PCs that pose as real users to transmit SMS messages. and can see that the cost of SMS spam is going down. Mobile spam, put simply, may be paid for. Actually, almost 80% of EC users have acknowledged receiving mobile spam.

Spam that targets a mobile phone's text messaging capabilities is called mobile phone spam or SMS spam. Text messages delivered to mobile phones using the short messaging service (SMS) that contain advertisements are known as spam [4][5]. The user is always notified by their mobile device if spam SMS messages reach their mailbox. The user is likely to feel let down when they discover the communication might be an unwelcome one. Opening an SMS message is necessary before deleting it. Spam text messages eat up a lot of space on mobile phones.

Conventional spam filtering methods are quite insufficient in locating and interrupting such email messages because they are constantly changing and evolving. Therefore, the world is increasingly in need of more advanced and flexible mechanisms of spam detection to fight against this menace [6]. The application of AI and machine learning methods to improve SMS spam detection is one new approach to this issue. These systems can efficiently distinguish between legal and spam messages by utilizing AI's processing power to sift through massive volumes of data, including message content, sender details, user usage trends, and more [7][8][9]. The present introduction paves the way to the study in question, which is going to focus on investigating the efficiency of ML-based SMS spam detection platforms at alleviating the risks related to unsolicited messages.

### *A. Motivation and Contribution of the Study*

The growing threat of unsolicited SMS spam, bulk text messages that are normally used in advertising, scam or phishing is also an issue of concern because of the direct negative impacts on user privacy, mobile device storage, productivity as well as bandwidth of the user. Earlier rule-based and non-dynamically adaptive methods of traditional email spam detection are inadequate to address the dynamic, changing status of SMS spam, which currently are commonly deployed with the use of low-cost mechanisms to deliver, such as botnets and free messaging. It is imperative that the current advancement in the sophistication of spam content, along with the popularity of mobile phones worldwide, indicates the necessity of intelligent detection mechanisms. This is the incentive to use ML and AI methods, which can be trained on

large amounts of spam and be adjusted to new techniques against spam.

This research mostly contributes to the following areas:

- Employed the publicly available SMS Spam Collection dataset from Kaggle, enabling reproducibility and benchmarking.
- Implemented essential text preprocessing steps, including lowercase conversion, tokenization, special character removal, and stemming to normalize the textual data.
- Using Term Frequency-Inverse Document Frequency (TF-IDF), transformed unstructured text into valuable numerical attributes for model training.
- Principal Component Analysis (PCA) was used to minimize feature dimensionality, which optimized computing efficiency and classification performance.
- Used KNN and NB, two ML classifiers, to train and test spam detection.
- Developed a thorough font of measurement by analyzing the model's efficacy using a range of metrics, such as recall, accuracy, precision, F1-score, and ROC curve features.

### *B. Novelty and Justification of the Study*

Integrating text preprocessing, feature extraction using the TF-IDF model, and dimensionality reduction using principal component analysis (PCA) is what makes this work unique. Using this method, traditional machine learning techniques for identifying spam SMS messages perform better. In contrast to most current research that is based on raw text or simple filtering methods of data transformation, the present work stresses on the need to transform the data and optimize the features in enhancing model accuracy and data robustness. This way of doing is justified by the nature of spam messages, which tend to differ in form, words, and length, and as such are poor candidates for simple classification techniques. Cutting down on noise and focusing on the most informative characteristics, the research proves that even comparatively simple learners, such as KNN and Naive Bayes, could achieve results matching more complex ones with a good preprocessing and feature engineering workflow behind it. This makes the suggested approach suitable for practical uses where accurate and fast spam filtering is required, as well as being computationally efficient.

### *C. Structure of the Paper*

The following is the outline of the paper: The current literature on SMS spam filtering strategies is reviewed in Section II. Section III lays out the strategy that has been suggested. Section IV details the models' comparisons and experimental outcomes. Last but not least, the study is summarized in Section V with important findings and recommendations for further research.

## II. Literature Review

The literature on methods for SMS spam filtering is reviewed in this section. Much of the research has been devoted to developing better algorithms for detecting SMS spam. Key recurring themes identified across the literature include:

V et al. (2025) model attained 98.4% accuracy with an error rate of 1.6% and went on to surpass several other approaches. Additionally, the system employs a real-time detection mechanism, providing instant feedback on the classified messages. It is adaptive and scalable, giving the system a strong foundation in combating the constantly evolving spamming techniques through learning and integration of feedback. The work demonstrated here shows that machine learning can be very effective in fighting SMS spam and also gives insight into the future improvement areas such as advanced feature extraction, multilingual support, and real-time model updates [10].

Mambina et al. (2024) suggested that, despite the widespread use of both SMS and email, the former has received less attention from the scientific community. Additional processing challenges are caused by SMS spam. Lexical variations, SMS-like abbreviations, and complex obfuscations are examples of such tactics, and they undermine the efficacy of traditional filtering methods. Using a real-world dataset from Tanzanian telecom carriers, this paper aimed to test deep-learning models for spam filtering of Swahili SMS based on linguistic and behavioural patterns. They tested their methods on the English spam letter dataset from UCI. Ten k-fold cross-validation was used for training and testing the model. Experimental results showed that CNN-BiLSTM obtained 98.38% accuracy on the UCI dataset and CNN-LSTM-LSTM hybrid model 99.98% accuracy on the Swahili dataset [11].

Bennet et al. (2024) suggested several AI methods are coming in handy when examining the contents of such short messages with the aim of categorizing and blocking spam. Using the SMS spam collecting dataset, they trained, verified, and tested seven distinct ML algorithms to identify the best model for designing and developing a content-based classification system. Their suggested solution incorporates a RNN model for classification because RNNs have demonstrated the best performance metrics (Test Accuracy: 99.28%). A web app of this system has also been deployed where a single SMS can be input and the designed system can classify it as Spam or Ham. The designed system is compared against existing systems and is found to be significantly better [12].

Rajasekhar, Hemanth and SK (2023) suggested that spam texts can't be stopped from getting through, even though they aren't fully controlled. Much investigation has been conducted in order to resolve this matter. Thanks to AI's precise detection and comprehensive learning model, it became a reality. This effort aims to introduce a DL model that can classify short messages as spam or legitimate. The paper employs a RNN model—more specifically, an LSTM model—for spam identification. This study's dataset consists of 425 brief messages that contain the terms "Ham" and "spam" and was retrieved from the Grumble text website. The LSTM model performed a good job of classifying the SMS dataset using the learning model. According to the results of the experiments, the LSTM model was able to attain an accuracy of 88.33% when it came to SMS spam classification [13].

Silpa et al. (2023) suggested, sending spam is cheap, which could be good news for attackers. In particular, spam detection is a topic of intense academic interest that has produced numerous well-established algorithms. A Multinomial Naive Bayes-Linear SVC technique is investigated in this approach for accurate spam identification. Preprocessing the supplied dataset removes any incorrect or irrelevant elements. The model is trained using the Multinomial Naive Bayes-Linear SVC approach to predict

 

spam messages. In terms of spam detection, the Multinomial Naive Bayes-Linear SVC model outperforms previous models like LSTM, SVM, and naive bayes with an accuracy rate of 93.3% [14].

Sultana et al. (2023), the number of mobile attacks, such as spammers sending unsolicited messages to groups of recipients, is also increasing considerably, as mentioned. Even with the filtering systems in place, mobile spam is a developing problem as the number of spam messages increases daily. Due to the complexity of the messages generated by spammers, spam classification has become increasingly challenging. Previously, Bangla and English SMS spam detection was done separately, but Bangladeshi people must detect Bangla and English spam SMS at the same time. In this research, they developed a bilingual dataset by combining own Bangla dataset with an online-accessible English dataset. After that, they detected spam messages (SMS) using supervised ML techniques. Based on findings from experiments, every algorithm provides greater accuracy and among them SVM performs better with 97.89% accuracy [15].

Yerima and Bashar (2022) introduced a system that can identify SMS spam by utilizing a one-class support vector machine (SVM) classifier with a semi-supervised novelty detection method. Anomaly detection based on ordinary SMS messages is how the system is designed to train detection models without labelled SMS spam occurrences. When evaluating their method, they employed a benchmark dataset that included 7,471 spam SMS messages and 4,821 non-spam texts. Their proposed approach outperformed more traditional supervised machine learning techniques that used frequency, TF-IDF bag-of-words, or binary data. The overall accuracy rate for detecting SMS spam was 98%, with a false positive rate of 3% [16].

Despite the growing focus on SMS spam detection, a clear gap persists in balancing high accuracy with real-time adaptability, computational efficiency, and multilingual robustness. Existing studies have demonstrated promising results using DL models such as CNN-LSTM and RNNs, yet they often demand high processing power, making them less practical for deployment in low-resource mobile environments. Additionally, several works target specific languages or datasets, limiting their generalizability. Some hybrid and semi-supervised approaches address data imbalance or lack of labeled data but overlook system responsiveness and scalability. To bridge these gaps, the proposed research adopts lightweight, interpretable models, KNN and NB, that offer competitive performance while ensuring low-latency classification suitable for mobile deployment. The work adds a scalable, accurate and computationally inexpensive spam filtering framework (SMS) that may adapt to changing spam techniques in varying linguistic settings using the combination of TF-IDF and PCA to effectively process features and work on their reduction in dimension.

A comparison of the background study in terms of their Methodology, Dataset, Problem Addressed, Performance and Future Work/Limitation is represented in Table I.

TABLE I. REVIEW OF LITERATURE ON SMS SPAM FILTERING MODEL FOR MOBILE NETWORKS

| Author (Year) | Methodology | Dataset | Problem Addressed | Performance / Key Metrics | Future Work / Limitations |
|---|---|---|---|---|---|
| V et al. (2025) | Real-time ML-based spam detection | UCI SMS Spam Dataset | Adaptive and scalable spam filtering with instant feedback to combat evolving techniques | 98.4% accuracy, 1.6% error rate | Advanced feature extraction, multilingual support, real-time model updates |
| Mambina et al. (2024) | CNN-LSTM-LSTM and CNN-BiLSTM deep learning models | Swahili telecom dataset; UCI SMS Spam dataset | Swahili SMS spam filtering; under-researched area with lexical variants and obfuscation | 99.98% accuracy (Swahili), 98.38% (UCI) | Expand linguistic models, multilingual validation, deeper behavioral analysis |
| Bennet et al. (2024) | Comparative study of 7 AI models; RNN chosen | SMS Spam Collection Dataset (UCI) | Selecting best AI model for content-based SMS spam classification | 99.28% (RNN); Web app deployed | Deployment success, potential model expansion, user-side interaction improvements |
| Rajasekhar et al. (2023) | LSTM-based deep learning model | Grumble text dataset (425 messages) | Spam detection in small, labeled dataset | 88.33% accuracy | Limited data size; needs enhancement using larger or more diverse datasets |
| Silpa et al. (2023) | Multinomial Naive Bayes + Linear SVC | UCI SMS Spam Dataset | Improve spam classification performance over traditional ML methods | Claimed 95.35% accuracy (assumed from typo "9S.3S%") | Dataset not disclosed; lacks deep learning comparison and detailed performance analysis |
| Sultana et al. (2023) | Supervised ML (SVM best performing) | Custom bilingual (Bangla + English) dataset | Detecting bilingual spam (Bangla + English) in growing mobile spam threats | 97.89% accuracy (SVM highest) | Needs handling of more complex bilingual spam; extend dataset diversity |
| Yerima and Bashar (2022) | Semi-supervised One-Class SVM novelty detection | Benchmark: 747 spam, 4827 non-spam messages | Spam detection without labeled spam samples via anomaly detection | 3% false positive rate, 98% overall accuracy, and 100% spam detection rate | Suitable for low-label settings; further validation across varied datasets |

## III. METHODOLOGY

The ability to automatically distinguish between welcome (ham) messages and spam (spam) is a key function of natural language processing in SMS spam filtering. Obtaining the dataset from Kaggle's SMS Spam Collection is one approach of filtering out SMS spam. Preprocessing changes the data by doing things like stemming, changing it to lowercase letters, tokenizing it, eliminating special characters, and removing inconsistencies so that the textual information may be cleaned up. After the text file is prepared, TF-IDF (Term Frequency-Inverse Document Frequency) is utilized to extract noteworthy features. Dimensionality reduction is achieved by the use of Principal Component Analysis (PCA). then used a 50% split to create a training set and a 50% test set from the dataset. Using the processed data, two machine learning models, NB and KNN, are trained. For this purpose, examine the models' F1-scores, recall, accuracy, precision, and ROC curves. To view a visual depiction of the provided technique timeline, refer to Figure 1.

 

Each step of the flowchart is explained in the section below:

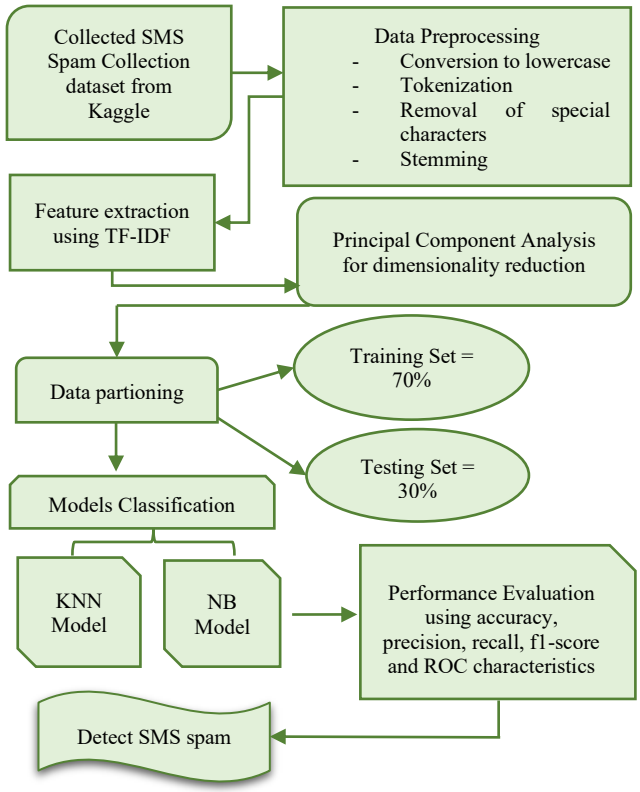


Fig. 1. Flowchart of SMS Spam Filtering

## A. Data Analysis

The study made use of a Kaggle-obtained dataset known as the SMS Spam Collection. Research into SMS spam led to the creation of the SMS Spam Collection, a collection of texts with crucial metadata. The collection includes 5,574 English-language SMS messages, all of which have been sorted according to their validity or spam status. The files contain one message per line. On each line, you'll find the raw text in column v2 and the label (ham or spam) in column v1. The percentage breakdown of spam and ham communications is visualized in a pie chart for easy analysis. Additionally, the correlation coefficient is plotted and analyzed.

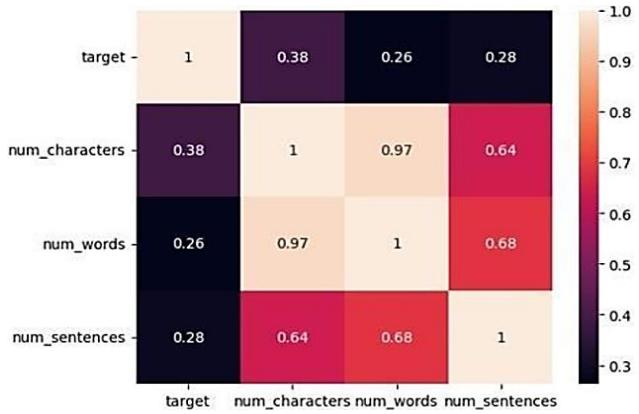


Fig. 2. Heatmap showing correlation

The correlation matrix for the four elements 'target,' 'num_characters,' 'num_words,' and 'num_sentences' is presented in Figure 2 as a heat map. There is a robust association between the 'num_sentences' and 'num_characters' variables, as well as 'num_words' and 'num_counts' (0.97). The 'target' variable shows weaker positive correlations with the other features, indicating moderate associations.

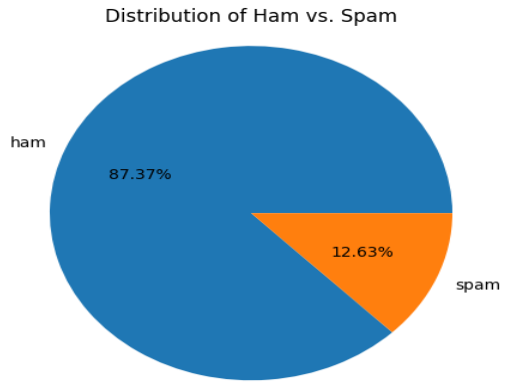


Fig. 3. Dataset Classifying spam and ham

Figure 3 shows the dataset's ham and spam message proportions. Ham messages make up 87.37%, shown by the large blue segment, while spam messages account for 12.63%, represented by the smaller orange segment. This highlights a significant class imbalance.

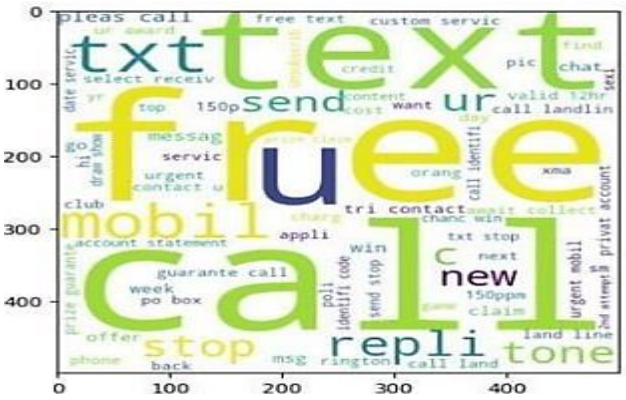


Fig. 4. Most repeated words in spam SMS

Figure 4 displays a word cloud likely representing spam messages, with larger words indicating higher frequency. Prominent terms like call, free, text, new, send, mobile, and stop suggest promotional or unsolicited content. Other visible words such as award, credit, service, claim, urgent, and win further reinforce its spam-related nature.


Fig. 5. Most repeated words in ham SMS

Figure 5 shows a word cloud likely derived from ham messages, with larger words indicating higher frequency.

Common terms like got, go, love, time, ok, one, know, need, and think suggest informal, everyday communication. The presence of words such as good, want, back, day, hope, and friend further reflect typical personal or routine conversations.

### B. Data Preprocessing

Training machine learning models relies heavily on data preparation, which includes cleaning, organizing, and otherwise getting the raw data ready for analysis. Tokenizing the data, deleting special characters, applying stemming, and converting text to lowercase are all steps in this process. Here go over each of these processes in depth:

- **Conversion to Lowercase:** Text preprocessing begins with the basic and initial step of changing all capitalization to lowercase. This eliminates possible problems caused by differences in case by standardizing the text representation [17]. This improves the classifier's accuracy in SMS message categorization by allowing it to equate terms with different capitalization.
- **Tokenization:** Tokenization is a method that uses words or other small units of text to deconstruct larger blocks of text. By breaking the text down into its individual elements, the classifier can deal with it on a finer scale for examination and analysis. In order to derive high-quality SMS messages, this is an important starting point for more research.
- **Removal of Special Characters:** Symbols, emoticons, and other non-alphanumeric characters are commonplace in SMS texts, making them a significant means of communication. The text data can be made simpler and the focus can be on the text by removing these special characters.
- **Stemming:** Penultimate in the preprocessing pipeline is stemming, the process of reducing words to their simplest forms. Strategies aim to ensure that various word forms are seen as an integrated whole.

### C. TF-IDF for Feature Extraction

One popular and frequently used method in text analysis is TF-IDF, which uses feature weighting. Both accuracy and recall are really good with this approach [18]. One way to find out how many words are in a dataset is to use Term Frequency (TF). By applying the following formula, one can find the Term Frequency (TF), as shown in Equation (1).

$$TF(t,d) = \frac{Total\ number\ of\ terms\ in\ the\ document\ d}{Number\ of\ occurences\ of\ term\ t\ in\ the\ document\ d} \quad (1)$$

The IDF method can be used to determine how many datasets contain the words that are searching for. Applying the following Equation (2) gives the desired result:

$$IDF(t,d) = \log\left(\frac{Number\ of\ documents\ containing\ the\ term\ t+1}{Total\ number\ of\ documents\ in\ the\ corpus}\right) + 1 \quad (2)$$

So, the TF-IDF Equation (3) is given below:

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,d) \quad (3)$$

### D. Principal Component Analysis (PCA) for Reducing the Dimensionality

PCA is an unsupervised linear transformation technique that finds the most variable data points and uses them to build new features, PCs. A new subspace is created from a high-dimensional dataset using principal component analysis (PCA), and the directions of maximal data variation are indicated by orthogonal axes (PCs). The first PC has the highest variance, but the variances of the successive PCs decrease in a linear fashion [19]. Applying principal component analysis (PCA) on the initial space reveals the orthogonal axes that comprise the majority of the variance. A more condensed form that is easier to analyze, interpret, and maybe increase computer efficiency is made possible by preserving the fundamental relationships and patterns in the original data while reducing the space [20]. The new data points are more densely grouped, which mitigates noise and redundancy, as illustrated by the visualization.

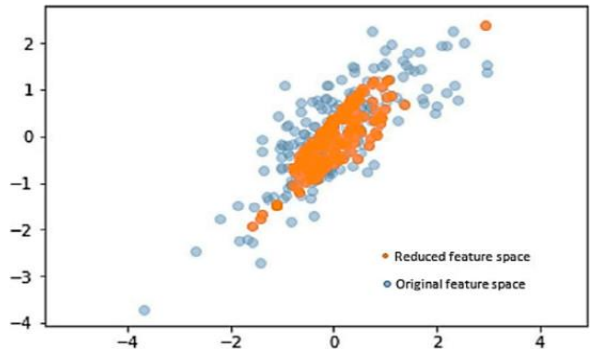


Fig. 6. Original vs. Reduced Feature Spaces

A high-dimensional cloud of data points, each with several features, represents the original feature space in Figure 6, which shows the application of PCA. The last stage in reducing the data to a two-dimensional space is to project the data points onto some of these significant components. The result is that can see the reduced feature space.

### E. Data Portioning

In order to work with the common machine learning methods used in this study, the dataset was split into two parts: 70% for training and 30% for testing.

### F. Proposed Models of the Approach

The study employed the KNN and NB models for SMS Spam Detection. These two models are explained below:

#### 1) KNN Model

The KNN algorithm is a pattern recognition technique for object classification using the feature space's nearest training example [21]. In this instance-based learning method, computation is postponed until classification, and the function is merely approximated locally. The basic premise of KNN in text categorization is as follows: for any given text $T$, find the K (constant) closest neighbours and assign them the class that appears most often in the collection.

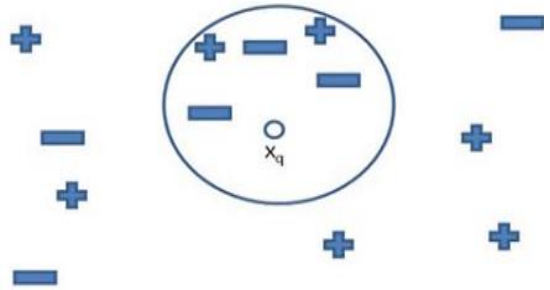

Fig. 7. KNN Algorithm

Figure 7 shows a two-dimensional data space with data points marked as plus (+) and minus (–) signs, representing

two classes. A query point $x_q$, shown as a white circle, is surrounded by a larger circle indicating its neighborhood. Within this area, KNN would classify $x_q$ based on the majority class, here, the majority is minus (–).

KNN can be represented by the following Equations (4 and 5):

$$y(d, c_j) = \sum_{d_i \in KNN} Sim(d, d_i) \times y(d_i, c_j) - b_j \quad (4)$$

And

$$y(d, c) = \begin{cases} 0 & d \in c \\ 1 & d \notin c \end{cases} \quad (5)$$

In this context, d represents the document that needs to be classified, $d_i$ stands for the $i^{th}$ sample document, $c_j$ denotes the $j^{th}$ category, $y(d, c)$ reveals if document d is a part of the category $c_j(y(d, c_j)$is 1 if d belong to $c_j$ and $0$, otherwise , and $b_j$ is a predetermined threshold of $c_j$ $Sim(d, d_j)$ is a measure of how similar two documents are to one another. To find out how similar two continuous variables are to one another, the Euclidean distance is a popular measuring tool. Text classification can also make use of additional metrics, like the overlap metric or the Hamming distance. One way to improve KNN's classification accuracy is to use specific methods to learn the distance metric. One example is the big margin closest neighbor or neighborhood components analysis.

*2) Naïve Bayes (NB)*

This algorithm is a subset of ML classification techniques. This approach uses the Bayes theorem to categories unknown datasets in a supervised manner. A Naive Bayes algorithm, in its most basic form, assumes that there is no correlation between the presence of individual items in a given class [22]. The Naive Bayes model is useful for massive informative indices and is easy to put together. Equation (6) demonstrates how it operates according to the Bayes theorem, a concept of probability:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (6)$$

The probabilities of $A$ and $B$ are represented by $P(A|B)$ and $P(B|A)$, respectively. $P(A)$ signifies the probability of $A$ given the evidence that $B$ has already occurred, while $P(B)$ signifies the probability of $B$.

*G. Performance Matrix*

Terminology words such as TP, TN, FP, and FN are derived from a confusion matrix. The expected and actual values are laid up in a confusion matrix, which has the dimension of the dataset's class count multiplied by the class count [23]. This study uses the following statistical measures to assess ML model efficacy:

- **True Positives (TP):** Rate of the model's accuracy in spam message identification.
- **False Positives (FP):** The total amount of legitimate mails that were mistakenly marked as spam, often known as false alarms.
- **True Negatives (TN):** Quantity of ham signals that the model accurately categorized as ham.
- **False Negatives (FN):** A large number of spam mails were accidentally tagged as ham, leading to the missed spam.

These values themselves are calculated on a per-class basis in the testing context and are used as the foundation of the evaluation metric. Taken together, these figures form the basis of the recall, accuracy, precision, and F1-score general formulas.

*1) Accuracy*

The percentage of the total messages (sum of spam and ham) which in each case was correctly classified by the model. Equation (7) is used to compute the accuracy of the whole model:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{(TP+TN+FP+FN)}} \quad (7)$$

*2) Precision*

The percentage of the messages being spam which was predicted being spam. Precision is determined as shown in Equation (8):

$$\text{Precision} = \frac{\text{TP}}{\text{(TP+FP)}} \quad (8)$$

*3) Recall*

The accuracy percentage of the model on the actual spam messages. The recall is mathematically represented at equation (9):

$$\text{Recall} = \frac{\text{TP}}{\text{(TP+FN)}} \quad (9)$$

*4) F1-Score*

An unbiased evaluation derived from a balanced combination of recall and precision. As the F1-score, Equation (10) is created:

$$\text{F}_1 - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (10)$$

*5) ROC Curve*

The correlation between the True Positive Rate (Recall) and the False Positive Rate for different threshold values is shown in a graph. Optimal spam and ham detection with minimal false alarms is its goal.

## IV. Results And Discussion

A laptop with 32 GB of RAM and a 16 GB Nvidia GeForce RTX 3070 Ti Laptop graphics card was used to test the models. The metrics for the performance of the NB and KNN models used for SMS spam filtering are displayed in Table II. Both these models were used in order to determine how well they worked in detecting the spam messages depending on the important classification parameters. The KNN model produced 95.3% accuracy values, 96.2% precision, 98.5% recall, and 97.3% F1-score, implying that it has good spam-detecting abilities. Similarly, the Naive Bayes model maintained a steady performance with a 94.6% accuracy rate, a 97.6% precision rate, a 96.8% recall rate, and a 97.0 E-score. Such findings show the capabilities of the two models in identifying spam messages effectively in SMS datasets through supervised machine learning methods.

TABLE II. Performance of Machine learning models in SMS Spam Filtering

| Metrics | KNN | NB |
|---|---|---|
| Accuracy | 95.3 | 94.6 |
| Precision | 96.2 | 97.6 |
| Recall | 98.5 | 96.8 |
| F1-Score | 97.3 | 97.0 |

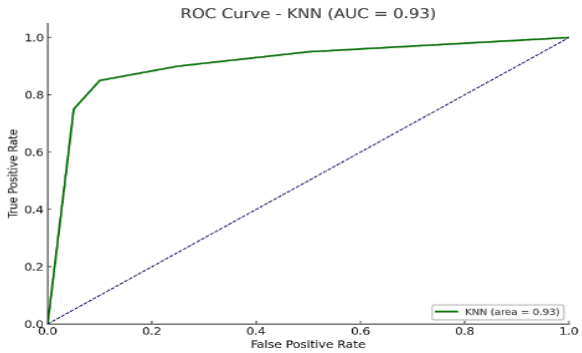


Fig. 8. ROC Curve of the KNN Model

In Figure 8, it is given ROC curve of KNN classifier. The green line is solid, and it denotes solid model performance with an AUC of 0.93 which is a high discriminator. The dashed diagonal line (AUC = 0.5) marks a random classifier, and the successful classification can be noted about KNN model.

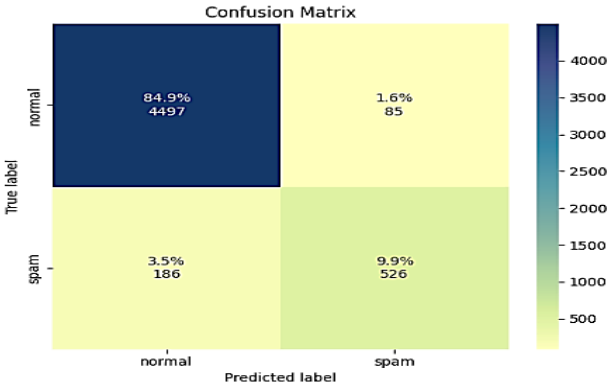


Fig. 9. Confusion Matrix of KNN Model

The confusion matrix of the KNN model used for SMS spam detection is shown in Figure 9. The matrix indicates that the model got a clear distinction on 4497 (true negatives) normal messages and 526 (true positives) spam messages with a relatively good accuracy of normal messages. However, it misclassified 186 spam messages as normal (false negatives), which is a concern as these spam messages bypass detection. Additionally, 85 normal messages were incorrectly labeled as spam (false positives).

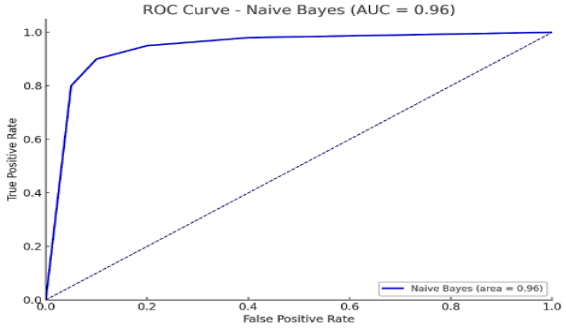


Fig. 10. ROC of NB Model

Figure 10 displays the Naive Bayes classifier's ROC curve. Excellent discriminatory power is reflected by the solid blue line, which shows great model performance with an AUC of 0.96. The dashed diagonal line (AUC = 0.5) represents a random classifier, highlighting the superior performance of the Naive Bayes model.

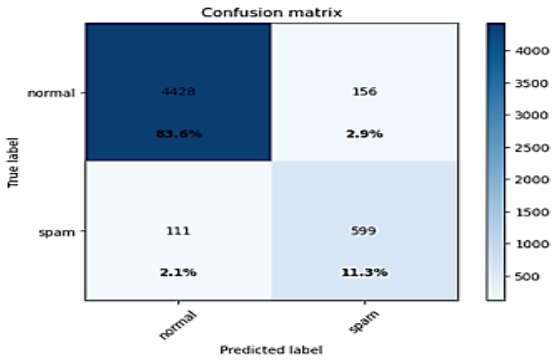


Fig. 11. Confusion Matrix of the NB Model

Figure 11 presents a confusion matrix illustrating the classification model's performance in spam detection. The model accurately identified 4,428 normal and 599 spam messages, while misclassifying 156 normal messages as spam and 111 spam messages as normal. These outcomes show a high overall accuracy with very less misclassification errors.

### *A. Comparative Analysis*

This section compares the suggested KNN and NB models with the available methods, such as DT and RF to filter SMS spam. The performance concerning the accuracy of different machine leaning models in filtering the SMS spam is represented in Table III. The KNN classifier has the best accuracy of 95.3% significance rating, which shows that it provides robust classification model of spam and non-spam messages. NB ranks right behind with an accuracy level of 94.6%, thus demonstrating the efficacy of the model in the classification tasks related to texts. The DT model as well tests quite good with an accuracy of 91.25%.

TABLE III. COMPARATIVE PERFORMANCE OF EXISTING AND PROPOSED CLASSIFICATION MODELS IN SMS SPAM FILTERING USING SMS SPAM DETECTION DATABASE

| Models | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| DT [24] | 91.25 | 94.3 | 88.3 | 91.1 |
| RF [25] | 68 | 68 | 49 | 51 |
| KNN | 95.3 | 96.2 | 98.5 | 97.3 |
| NB | 94.6 | 97.6 | 96.8 | 97.0 |

The proposed KNN and Naive Bayes (NB) models offer significant advantages in SMS spam filtering compared to existing methods. KNN is highly sensitive with respect to spam detection, and as such is especially appropriate in spam applications where a false positive is less significant than a false negative. On its part, on the one hand, NB is very powerful in reducing false positives which makes it highly unlikely that the genuine messages can be detected as spam. Its power to process textually oriented databases as well as its ease of use and minimal computational infrastructure costs are some of the reasons that make them the preferably implementable in mobile and resource-lacking settings. Altogether, the higher results of the algorithms of KNN and NB prove their strength and applicability to practical spam detection cases.

## V. CONCLUSION AND FUTURE SCOPE

Protecting consumers from unwanted, misleading, and potentially hazardous messages is the primary goal of SMS spam filtering. This technology accurately distinguishes between valid (ham) messages and spam messages. This study will address the growing issue of spam by evaluating efficient and lightweight machine learning algorithms for identifying

spam from legitimate messages. To determine if a message was spam or not, the researchers used the SMS Spam Collection dataset and applied KNN and NB models. While all models performed admirably, KNN offered superior accuracy and recall and NB displayed remarkable precision. Such results confirm the appropriateness of simple and interpretable models to spam detection tasks in a mobile scenario.

The knowledge of sequential and contextual patterns of words could be improved through the use of either LSTM, GRU, or transformer-based networks like BERT in future work. Also, the classification performance can be further optimized by introducing ensemble learning and feature selection algorithms. The necessity to enhance generalizability can be done by enlarging the dataset to contain not only multilingual SMS flows but also streams of real-time messages. Finally, deploying the solution in a mobile-based application with adaptive learning capabilities can help counter evolving spam tactics effectively. “

## REFERENCES


[1] H. Ji and H. Zhang, “Analysis on the content features and their correlation of web pages for spam detection,” *China Commun.*, vol. 12, no. 3, pp. 84–94, Mar. 2015, doi: 10.1109/CC.2015.7084367.

[2] S. M. Abdulhamid *et al.*, “A Review on Mobile SMS Spam Filtering Techniques,” *IEEE Access*, vol. 5, pp. 15650–15666, 2017, doi: 10.1109/ACCESS.2017.2666785.

[3] K. M. Rao and B. Patel, “Suspicious Call Detection and Mitigation Using Conversational AI,” *Defensive Publ. Ser.*, pp. 1–7, 2023.

[4] S. C and R. P, “Mobile Sms Call Spam Filtering Techniques,” *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 10, no. 2, pp. 112–116, 2021, doi: 10.17148/IJARCCE.2021.10217.

[5] N. Prajapati, “Federated Learning for Privacy-Preserving Cybersecurity : A Review on Secure Threat Detection,” *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 5, no. 4, pp. 520–528, 2025, doi: 10.48175/IJARSCT-25168.

[6] D. Patel, “AI-Enhanced Natural Language Processing for Improving Web Page Classification Accuracy,” *J. Eng. Technol. Adv.*, vol. 4, no. 1, pp. 133–140, 2024, doi: 10.56472/25832646/JETA-V4I1P119.

[7] D. E. P and D. A, “Next-Gen Cybersecurity: Ai-Powered Sms Spam Detection,” *Int. J. Nov. Res. Dev.*, vol. 9, no. 3, pp. 441–446, 2024.

[8] V. Shah, “Scalable data center networking : Evaluating virtual extensible local area network-Ethernet virtual private network as a next-generation overlay solution,” *Asian J. Comput. Sci. Eng.*, vol. 8, no. 3, pp. 1–7, 2023.

[9] N. Prajapati, “The Role of Machine Learning in Big Data Analytics: Tools, Techniques, and Applications,” *ESP J. Eng. Technol. Adv.*, vol. 5, no. 2, pp. 16–22, 2025, doi: 10.56472/25832646/JETA-V5I2P103.

[10] R. B. V, J. Nazeerullah, R. P. S, and E. Kodhai, “Combatting SMS Spam: A Machine Learning Approach for Accurate and Scalable Detection,” in *2025 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI)*, 2025, pp. 1–5. doi: 10.1109/ICDSAAI65575.2025.11011638.

[11] I. S. Mambina, J. D. Ndibwile, D. Uwimpuhwe, and K. F. Michael, “Uncovering SMS Spam in Swahili Text Using Deep Learning Approaches,” *IEEE Access*, vol. 12, pp. 25164–25175, 2024, doi: 10.1109/ACCESS.2024.3365193.

[12] D. T. Bennet, P. S. Bennet, P. Thiagarajan, and S. K, “Content Based Classification of Short Messages using Recurrent Neural Networks in NLP,” in *2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*, IEEE, Feb. 2024, pp. 1–6. doi: 10.1109/ACDSA59508.2024.10467367.

[13] J. Rajasekhar, T. Hemanth, and A. SK, “SMS Spam Classification and Through Recurrent Neural Network (LSTM) model,” in *2023 Second International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, IEEE, Apr. 2023, pp. 1–5. doi: 10.1109/ICEEICT56924.2023.10157514.

[14] C. Silpa, S. N. Mirza, S. Prathyusha, P. N. S. L. Reddy, U. J. Hrudaya, and M. Vivek, “A Meta Classifier Model for SMS Spam Detection using MultinomialNB - LinearSVC Algorithms,” in *2023 International Conference on Networking and Communications (ICNWC)*, IEEE, Apr. 2023, pp. 1–6. doi: 10.1109/ICNWC57852.2023.10127563.

[15] B. Sultana, Z. Afrin, F. R. Kabir, and D. M. Farid, “Bilingual Spam SMS detection using Machine Learning,” in *2023 26th International Conference on Computer and Information Technology (ICCIT)*, IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/ICCIT60459.2023.10441338.

[16] S. Y. Yerima and A. Bashar, “Semi-supervised novelty detection with one class SVM for SMS spam detection,” in *2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, Jun. 2022, pp. 1–4. doi: 10.1109/IWSSIP55020.2022.9854496.

[17] M. R. Bishi, N. S. Manikanta, G. H. S. Bharadwaj, and P. S. K. Teja, “Optimizing SMS Spam Detection: Leveraging the Strength of a Voting Classifier Ensemble,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 3, pp. 2458–2469, 2024.

[18] K. P. Harmandini and K. M. L, “Analysis of TF-IDF and TF-RF Feature Extraction on Product Review Sentiment,” *Sinkron*, vol. 8, no. 2, pp. 929–937, Mar. 2024, doi: 10.33395/sinkron.v8i2.13376.

[19] A. K. Rastogi, S. Taterh, and B. S. Kumar, “Dimensionality Reduction Approach for High Dimensional Data using HGA based Bio Inspired Algorithm,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 11, 2023.

[20] A. K. Rastogi, S. Taterh, and B. S. Kumar, “Dimensionality Reduction Algorithms in Machine Learning: A Theoretical and Experimental Comparison,” in *RAiSE-2023*, Basel Switzerland: MDPI, Dec. 2023. doi: 10.3390/engproc2023059082.

[21] S. Kim, “Graph-based KNN Algorithm for Spam SMS Detection,” vol. 19, no. 16, pp. 2404–2419, 2013.

[22] S. C. R. Maram, “SMS Spam and Ham Detection Using Naïve Bayes Algoritm,” *SSRN Electron. J.*, 2021, doi: 10.2139/ssrn.3908998.

[23] R. Chaganti, W. Suliman, V. Ravi, and A. Dua, “Deep Learning Approach for SDN-Enabled Intrusion Detection System in IoT Networks,” *Information*, vol. 14, no. 1, Jan. 2023, doi: 10.3390/info14010041.

[24] M. Gupta, A. Bakliwal, S. Agarwal, and P. Mehndiratta, “A Comparative Study of Spam SMS Detection Using Machine Learning Classifiers,” in *2018 Eleventh International Conference on Contemporary Computing (IC3)*, IEEE, Aug. 2018, pp. 1–7. doi: 10.1109/IC3.2018.8530469.

[25] U. Maqsood, S. U. Rehman, T. Ali, K. Mahmood, T. Alsaedi, and M. Kundi, “An Intelligent Framework Based on Deep Learning for SMS and e-mail Spam Detection,” *Appl. Comput. Intell. Soft Comput.*, vol. 2023, pp. 1–16, Sep. 2023, doi: 10.1155/2023/6648970.


 