UMT PRESS
Journal of Mathematical Sciences and Informatics
Journal Homepage: https://journal.umt.edu.my/index.php/jmsi
eISSN: 2948-3697
DOI: http://doi.org/10.46754/jmsi.2026.06.007

AN INTERACTIVE SPAM DETECTION MODEL USING AN ENSEMBLE ALGORITHIM

GBOTOSHO AJIBOLA, OZOH PATRICK* AND ABOLARINWA MICHAEL OLUGBENGA

Faculty of Computing and Information Technology, Osun State University, Osogbo, Nigeria.

**Corresponding author: patrick.ozoh@uniosun.edu.ng*

ARTICLE INFO

Article History:
Received: 7 July 2025
Revised: 3 December 2025
Accepted: 11 March 2026
Published: 15 June 2026

Keywords:
Spam messages, SMS, preprocessing, TF-IDF, feature extraction.

ABSTRACT

Short Message Service (SMS) spam remains a significant issue in mobile communication systems. This study presents a framework for identifying spam messages in SMS communications using an ensemble of machine learning techniques for classification. The proposed framework involves a data preprocessing procedure to prepare the raw SMS dataset, followed by feature extraction using the Term Frequency–Inverse Document Frequency (TF-IDF) technique to represent textual data numerically. This enables the model to capture relevant characteristics from the processed data. During this experiment, the Ensemble model achieved 97% accuracy, outperforming Support Vector Machine (SVM) K-Nearest Neighbour (KNN), and Random Forest (RF), which achieved accuracies of 70.93%, 73.21%, and 79.18%, respectively. The proposed approach enhances user security in mobile communication and provides a comprehensive solution to the persistent issue of SMS spam, providing an effective spam detection system.

2020 Mathematics Subject Classification:

Introduction

An interactive Spam Detection Model refers to the adoption of adaptive learning techniques to filter through, detect and discard or quarantine unsolicited, bulk, and unwanted messages. Bouke *et al.* [7] applied machine learning techniques to a framework for investigating limitation detection techniques for secure and reliable email communication. A great number of people use mobile devices. The volume of Short Message Service (SMS) communications has significantly expanded, thereby allowing spammers to send messages for their own individual gain. Gadde *et al.* [10] applied different machine learning techniques to identify unsolicited, bulk, and unwanted SMS messages to develop a detection model.

As the popularity of the SMS platform has increased, so too has the number of unsolicited commercial advertisements via the platform. In Africa, especially in Nigeria, unsolicited, bulk, and unwanted SMS messages have been on the rise especially among young people. While not yet rampant, there is evidence of an uptick in the number of these unwanted messages [17]. Ejirika and Omotehinwa [9] revealed that SMS spam accounts for 21% of all text messages received. Some of these messages are unsolicited advertisements for various goods and services [14]. These spam messages can be particularly frustrating for Nigerian

mobile phone users, who pay a fee to receive them. In some cases, users have reported receiving as many as 50 spam messages [13].

The situation is similar in other African nations In Kenya, for example, SMS spam is estimated to account for about 10% of all text messages received [12]. This has led to an increase in demand for spam-filtering software, which is still not widely available in the region. Furthermore, SMS spam can be more disruptive than those received via email because, in some countries, the the recipient may be charged a fee such messages. As a result, detecting such messages has become an important research problem [24].

Technological advancements in mobile messaging have also contributed to an increase in spam SMS messages. Wang [22] proposed a technique to classify SMS messages. which outperformed current techniques. Future work suggests applying the model to larger datasets to enhance its performance.

Messaging through mobile phones has become increasingly important for the dissemination of information. Nonetheless, an increase in malicious messages created significant challenges for users. Airlangga [2] presented a parameter optimisation technique, in which different models were applied to detect unsolicited messages with projections aimed at improving accuracy. Future studies were recommended to investigate class imbalance and enhance model performance. These results provide important knowledge for evolving more accurate SMS spam detection systems and for building robust models, while enhancing the end user experience.

This study utilises traditional machine learning methods such as Support Vector Machine (SVM) SVM, K-Nearest Neighbour (KNN), and Random Forest (RF),, which have been applied in classification tasks, but lack the adaptability required for dynamic offloading in Space-Air-Ground Integrated Networks (SAGIN) environments, combine satellite, aerial (UAVs/HAPS), and terrestrial networks to provide seamless, wide-coverage, and low-latency 6G communication. This study contains five sections. Section 1 consists of the background study, introduction, and contributions to knowledge. Section 2 contains related works, which also discusses different feature extraction methods for SMS spam detection. The techniques used in this study are discussed in Section 3. This section looks at the data collection methods, algorithms, and the evaluation of results. Meanwhile, Section 4 summarises the key findings of this study, and looks at the performance of the proposed method against more conventional methods. Section 5 contains a summary of the findings of this research paper as well as the conclusions drawn from the findings of the study.

This research paper, aims to:

a) Enable individuals and businesses to filter out unwanted SMS spam messages.
b) Evaluate the SMS spam detection system proposed in this research paper.

Literature Review

Filtering Unwanted SMS Spam Messages

Several studies have been conducted on unsolicited SMS messages. Ahmed *et al.* [1] used Naive Bayes and decision trees to classify unsolicited SMS messages as spam. The article achieves an accuracy of 99.28% using Naive Bayes and an accuracy of 98.12% using decision trees. The strength of this study lies in its enormous dataset and the high level of accuracy

achieved. Nallamothu and Khan [16] used a combination of techniques to classify SMS messages as spam. The authors achieved an accuracy of 97.6%, lower than that of [21]. The Nallamothu study used a simple and effective combination of algorithms. However, the study did not evaluate or consider the impact of feature selection on the results.

Meanwhile, Srinivasarao and Sharaff [21] used adaptive spam detection algorithms. Al-Kabbi *et al.* [3] pointed out that rule-based approaches do not detect new or previously unseen spamming tactics. Additionally, as noted by De Goma *et al.* [8], the scarcity of annotated data also poses a significant challenge. The diversity of spamming tactics, including the use of different languages, keywords, and sender identities, requires the development of a generalisable spam detection algorithm. The increased use of web applications to deploy such algorithms ensures that a large number of consumers can link their opinions with potential users and producers through various sales platforms. Qazi *et al.* [19] identifies existing methodologies of various review platforms for different groups, where important assessments are applied. The study indicates that there are many filtering methods applicable to resolving SMS spam detection issues.

The present study provides a wider understanding of the challenges in spam identification, as well as substantial enhancements using machine-learning methods. These will act as a benchmark for future studies to improve the accuracy of the classifications, which has great potential for detecting unsolicited SMS messages [11]. Al-shanableh *et al.* [4] provided various methods with which to investigate the optimisation of the spam detection model.

Developing an SMS Spam Detection System

Space-Air-Ground Integrated Network (SAGIN)

The Space-Air-Ground Integrated Network (SAGIN) refers to a system that combines satellite, aerial, and terrestrial networks to provide seamless, low-latency, network coverage. This integrated network presents a more competent and effective mode of information transmission and communication. However, SAGIN is more susceptible to attacks than conventional networks, as it dramatically expands the attack surface for malicious actors, and introduces new vulnerabilities through heterogeneous connectivity.

As such, the network infrastructure requires automated, intelligent filtering for 6G networks better malicious message detection algorithms that are more accurate, have better real-time performance outcomes, and lower resource consumption rates to assist SAGIN achieve its full potential as a robust and reliable global network for next-generation communication systems [23].

This network is characterised by a highly mobile architecture, which presents real-time system performance challenges, which would benefit greatly from Open issues message detection system. This study proposes an artificial intelligence model to showcase model architecture and analyse major challenges affecting the network, especially with regard to spam detection, as the growing volume of SMS messages and higher incidences of malicious message dissemination is likely to burden the network infrastructure, which will degrade network performance and impact end user experiences. The study presents future trends and identifies major technical challenges. The purpose of this is to present a guide and a set of rules for the development of a new generation of intelligent and adaptive SAGIN architectures,

capable of creating an efficient network for next-generation communication systems. The application of artificial intelligence presents an important control attribute, which can improve automation and efficiency [24]. A key technique for developing an intelligent learning model used for executing multiple functions simultaneously in SAGIN is developed to control resources and services across the network. This study will help facilitate the integration of Space-Air-Ground Integrated Networks (SAGIN) and Artificial Intelligence (AI) for future studies into improving wireless networks.

Additionally with the increase in the number of Internet of Things (IoT) devices and users, the collision probability increases, further reducing system performance. Wu *et al.* [25] investigated device access in SAGIN-assisted high-altitude platforms, focusing on power allocation for IoT devices to optimise system access potential. The study involved applying the Advantage Actor-Critic (A2C) algorithm, which is a reinforcement learning algorithm that combines the strengths of policy-based and value-based methods to achieve stable and efficient training, which improves the probability of the IoT terminal access with low complexity. A system of cybersecurity-enabled models is presented in [27]. The article presents SAGIN architecture with a unique digital twin (DT) model. Furthermore, a case study describing AI is presented, where open research issues are described.

The Need for a Secure and Efficient Offloading in the SAGIN Environment

Enhancements in wireless communication technologies have boosted the use of smart devices such as smartphones, home equipment, and monitoring sensors [23]. These systems require the ability to store and analyse quantities of real-time data, which imposes a considerable strain on the network infrastructure, considerably increasing the load on these systems. This is important to improvements with sixth-generation networks, contributing to making global connectivity possible, more importantly, ensuring the communication requirements of remote locations in need of cellular network infrastructure are available.

The SAGIN environment concentrates on introducing dynamic strategies for establishing hosting functions for mobile devices, offloading computations, and controlling the relationships for allocating computing resources. The aim is to reduce the time-averaged network cost, particularly the uncertainty of device environments, speeds, and types.

Wu *et al.* [25] devised a deep learning perception-aided online technique to solve issues in an environment filled with uncertainties. The reliability of the proposed method is validated using enlarged numerical simulations to quantify its performance with respect to different network parameters. The Space-Air-Ground Integrated Networks (SAGIN) provide effective solutions for a high-quality communication network and meeting ever larger computing demands.

Motivation for the Use of Multi-Agent Reinforcement Learning

Reinforcement Learning (RL) is increasingly linked to SMS spam detection because traditional, static machine learning models are failing to keep pace with the evolving, sophisticated tactics of spammers. RLcan solve some of the problems faced by SAGIN with regard to SMS spam detection by applying a simple architecture to the dynamics of the problem.

Reinforcement learning offers a dynamic approach to this problem by enabling spam filters to learn and improve through operational feedback, rather than relying solely on periodic manual retraining which has been applied successfully to address issues in networks linked to finance, robotics, natural language processing, and telecommunications.

The objective of reinforcement learning is to enable an agent to work in an environment that models the objectives that must be accomplished. This system guides the agent into making decisions that optimise upcoming rewards. Nonetheless, there is a need to solve real-world challenges. This cannot be achieved by single active agent, but by a multi-agent system, which allows many agents to learn at the same time.

Limitations in Existing Offloading Methods, Security Models

As the volume of data and the number of cyberthreats increase, existing cybersecurity systems can become overworked, making them less effective and reducing the speed at which they ae able to analyse security concerns. Existing offloading methods, which reduce CPU loads and improve performance by shifting tasks to specialized hardware, low traffic networks depend on rigid rules and predetermined attack signatures, which limits its effectiveness in detecting evolving threats and vulnerabilities. Existing techniques cannot comprehend the full structure of a peculiar network environment, the particular methods used by attackers, and the unpredictable nature of the attacks. This sometimes leads to unfinished threat detection and ineffective answers to unfolding and or complex cyberattacks, especially via SMS over mobile networks.

While reinforcement learning possesses significant potential for resolving a host if cybersecurity issues, it still encounters limitations. Some of these limitations are particular to reinforcement learning and are important for dealing with challenges in cybersecurity environments.

Reinforcement learning is valuable in cybersecurity because it supports adaptive learning and helps systems respond to evolving threats. However, its effectiveness is often limited by the complexity of large networks and the need for substantial amounts of high-quality data, which may not always be available.

Additional challenges include integrating reinforcement learning into rigid systems, maintaining robustness in real-world environments, and adapting to changing network conditions without poor generalisation. Moreover, the decisions produced by reinforcement learning models must remain interpretable so that stakeholders can understand and trust the outcomes.

Security Modelling: Threat Models, Secure Communication Protocols, Trust Mechanisms

The objective of threat modelling is to choose, communicate, and interpret threats and their eventual reduction to stakeholders. An evaluation of likely attacks, probable attack careers, and likely assets that can be attacked should be provided.

Security modelling is the systematic process for identifying, evaluating, and reducing evolving threats to a system. It vigorously identifies threats to the security of a system before they are activated, ensuring risk reduction and enhanced security. Threat model identifies,

communicates, and comprehends threats and relief for the protection of the system. Threat modelling uses theoretical structures, system diagrams, and testing to safeguard data.

MARL Algorithm: Choice Rationale, Training Process, Convergence Criteria

Multi-Agent Reinforcement Learning (MARL) is linked to the need for SMS spam detection because traditional, static spam filters cannot cope with the increasingly intelligent, adaptive, and distributed nature of spam attacks. MARL provides achievable answers to complex and evolving environments with decision-making attributes that interact with the systems in a network. This method investigates problems encountered by high-dimensional observations and action in cybersecurity and provides practical solutions for implementation. However, the MARL algorithm must be applied carefully to achieve the requirements of the agents. Cybersecurity requires a cautious choice of algorithms that relate to the requirements of the system and the level of operations among agents. The Multi-Agent Reinforcement Learning algorithm manages challenges effectively and removes issues such as conflicting actions. The complexity can increase remarkably; thus, the MARL algorithm needs maximal domains for a small number of nodes.

Cybersecurity involves several important real-world challenges, particularly in large-scale environments where some devices may lack dedicated incident response support. In the Multi-Agent Reinforcement Learning (MARL) framework, agents share and split data during training while operating independently during deployment. This approach enhances learning efficiency, reduces communication overhead during implementation, and lowers overall training complexity.

Simulation Setup: Platform, Parameters, Scenarios (e.g., Low-Earth Orbit Satellite Handoffs, UAV Mobility)

Future communication networks are designed to allow an increased number of services and systems by presenting their uninterrupted use, providing a high-capacity data connection using current communication and networking technologies that improve efficiency.

The SAGIN environment simulation setup is designed to produce an enhanced network performance [27]. The network comprises of satellites and the mobility of UAVs. The simulation platform analyses system models. This will assist in improving the SAGIN simulation platform.

Comparison of Techniques

The techniques considered in this section are summarised in the table below to identify the reasons for choosing the methods used in this study and to discuss the advantages and disadvantages of each technique.

Table 1: Characteristics of past machine learning methods

Method	Advantages	Disadvantages
Naive Bayes Algorithm	1. The performance evaluation is less complex 2. Reduced overfitting 3. Takes care of missing data 4. Useful for classification and regression functions 5. Naive Bayes Algorithm gives knowledge to features that are important for predictions	1. Requires much memory and needs many resources 2. Less interpretable than individual decision trees
Maximum Entropy Algorithm	1. Develops training instances for a general model 2. Approximations are made to the target function 3. These algorithms can easily adapt to new data collected over time	1. The method exhibits characteristics of lazy learning 2. Relies on storing data 3. The complexity of the hypothesis can grow with the data 4. Each query consists of beginning the new model from scratch, leading to high classification costs 5. A huge memory is required to store data
Support Vector Machine	1. SVM is adequate for high-dimensional spaces and image classification analysis 2. Handles non-linear relationships 3. Improves accuracy 4. SVM is adequate for text classification	1. Overfitting of models 2. Slow for big datasets, impacting performance 3. Adjusting parameters needs careful tuning 4. SVM is difficult with noisy datasets, reducing effectiveness
K-Means Clustering	1. Approximations are made to the target function 2. K-Means Clustering is adequate for text 3. Handles non-linear relationships	1. K-Means Clustering 2. Adjusting parameters needs careful tuning 3. The complexity of the hypothesis can grow with the data

This study reviewed the literature, using an online data repository. The study compares the Naive Bayes Algorithm, the Maximum Entropy Algorithm, K-Means Clustering, and the Support Vector Machine, with results indicating that the Naive Bayes Algorithm was the most widely used technique, followed by the K-Means Clustering, the Maximum Entropy Algorithm, and the Support Vector Machine. An Ensemble model that integrates these four techniques will enhance the accuracy of the findings of this study.

This study proposes an ensemble model be devised to detect spam messages in mobile applications, as the decision-making of a larger group is superior to that of a single expert.

Methodology

This section is an overview of the methodology used.

Data

The dataset is obtained from publicly available sources, with no class imbalance.

Description of Techniques

Naive Bayes Algorithm

This technique consists of features that are conditionally independent of the class label, simplifying the computation process. The algorithm performs well in real-world applications, specifically text classification and spam filtering.

Step 1: Divide the original data into two classes
Step 2: Build machine learning models using parameters
Step 3: Estimate the probability distribution for the two classes
Step 4: Combine the techniques using conditional independence
Step 5: Predict classification algorithms

The sentiment extraction method consists of statistics, machine learning, and classification algorithms.

Maximum Entropy Algorithm

It is given as [18].

$$\log_{\overrightarrow{p}}(p) = log \prod_{x \in X} p(x)^{\overrightarrow{p(x)}} = \sum_{x \in X} p(x) \log p(x) \quad (1)$$

Support Vector Machine

It is given as [22].

$$\begin{aligned} TS_i &= \max\left\{\left|\frac{\overline{x}_{ik} - \overline{x}_i}{m_k s_i}\right|, \quad k = 1, 2, \ldots K\right\} \\ \overline{x}_{ik} &= \sum_{j \in C_k} \overline{x}_{ij}/n_k \\ \overline{x}_i &= \sum_{j=1}^{n} x_{ij}/n \\ s_i^2 &= \frac{1}{n-K} \sum_k \sum_{j \in C_k} (x_{ij} - \overline{x}_{ik})^2 \\ m_k &= \sqrt{1/n_k + 1/n} \end{aligned} \quad (2)$$

K is the number of iterations. The number of samples is given as n.

K-Means Clustering

This is given as [15]. The weights are assigned as in Equation 3. Dt are constants, and N is sample.

$$\frac{1}{m_k}\sum_{i=1}^{m_k}\left\|x^i - \mu_c k\right\|^2 \quad (3)$$

Ensemble

Ensemble learning is a collection (or ensemble) of basic learners or models, which improve the final prediction. The ensemble technique employed in this study is known as a stacking classifier an ensemble machine learning technique that combines multiple base-level classification models (Level-0) via a meta-classifier (Level-1) to improve overall prediction accuracy. It is shown in the Figure 1 below.

Input: Training data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{m}$ $(\mathbf{x}_i \in \mathbb{R}^n, y_i \in \mathcal{Y})$
Output: An ensemble classifier H

1: Step 1: Learn first-level classifiers
2: **for** $t \leftarrow 1$ to T **do**
3: Learn a base classifier h_t based on $\mathcal{D}$
4: **end for**
5: Step 2: Construct new data sets from $\mathcal{D}$
6: **for** $i \leftarrow 1$ to m **do**
7: Construct a new data set that contains $\{\mathbf{x}'_i, y_i\}$, where $\mathbf{x}'_i = \{h_1(\mathbf{x}_i), h_2(\mathbf{x}_i), \ldots, h_T(\mathbf{x}_i)\}$
8: **end for**
9: Step 3: Learn a second-level classifier
10: Learn a new classifier h' based on the newly constructed data set
11: **return** $H(\mathbf{x}) = h'(h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_T(\mathbf{x}))$

Figure 1: Stacking pseudocode

Comparison and Validation

The results in this study are compared with results from [28], [29], and [30].

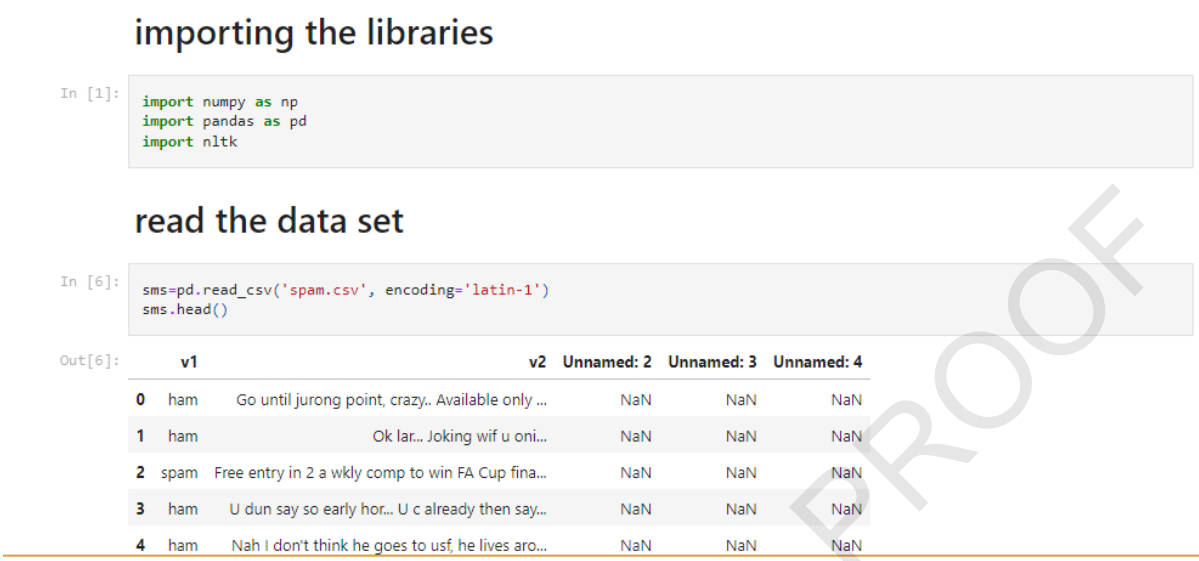
$$\text{RMSE} = \sqrt{\sum\{e_t\}^2}$$

$$MAPE = \frac{\sum \frac{|e_t|}{x_t}}{n}$$

et are the errors at period t, t is the current period, and x_t is the current value for n observations.

Discussion of Results

Results for data collection include obtaining data loaded into the machine learning environment in a text file. This is shown in Figure 2.



importing the libraries

In [1]:
```
import numpy as np
import pandas as pd
import nltk
```

read the data set

In [6]:
```
sms=pd.read_csv('spam.csv', encoding='latin-1')
sms.head()
```

Out[6]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN

Figure 2: Reading data

For training purposes, the pre-processed dataset is divided. The spam and non-spam labels are converted into numerical values. Figure 3 shows the training data.

Figure 3: Training data

Figure 4 shows the code for implementing the TFIDF technique. This figure shows the importance of obtaining insights into what characteristics.

```
Out[118…  <5572x8672 sparse matrix of type '<class 'numpy.int64'>'
          with 73916 stored elements in Compressed Sparse Row format>

In [111…  print(x_train.shape)
          print(x_test.shape)

          input=text[5571]

(4457, 8672)
(1115, 8672)
```

implementation of ML Model

```
In [62]:  from sklearn.neural_network import MLPClassifier

          model=MLPClassifier()
          model.fit(x_train, y_train)

Out[62]:  MLPClassifier(activation='relu', alpha=0.0001, batch_size='auto', beta_1=0.9,
                        beta_2=0.999, early_stopping=False, epsilon=1e-08,
                        hidden_layer_sizes=(100,), learning_rate='constant',
                        learning_rate_init=0.001, max_fun=15000, max_iter=200,
                        momentum=0.9, n_iter_no_change=10, nesterovs_momentum=True,
                        power_t=0.5, random_state=None, shuffle=True, solver='adam',
                        tol=0.0001, validation_fraction=0.1, verbose=False,
                        warm_start=False)
```

Figure 4: Implementation of the TFIDF technique

Evaluation of Results

The results are as follows. Table 2 shows performance evaluation against the metrics for the SVM, Naive Bayes Algorithm, Maximum Entropy Algorithm, K-Means Clustering, and Ensemble models.

Table 2: Performance evaluation result

Classifiers	Accuracy	Precision	Recall	F1-Score	Time(sec)
Support Vector Machine (SVM)	70.93	56.48	98.67	71.84	48.37
Naive Bayes Algorithm	76.82	65.50	80.97	72.42	34.91
Maximum Entropy Algorithm	79.18	67.26	86.90	75.83	31.29
K-Means Clustering	73.21	69.43	81.29	71.28	29.93
Ensemble	97.41	97.73	95.31	96.51	46.64

The F1-score ranges between 0 and 1, or equivalently between 0% and 100%. It is calculated using a confusion matrix. The study achieves an accuracy of the Ensemble 97.41% compared to Naive Bayes Algorithm (76.82%), Maximum Entropy Algorithm (79.18%), K-Means Clustering (73.21%), SVM (70.93%), and F1-scores of 96.51%, 72.42%, 75.83%, 71.28%, and 71.84% respectively, for a real-world spam detection model.

Results of Comparisons

This section computes results based on the models described in Section 3 (from previous studies [28] and [30]). the findings of these computations are presented in Table 3.

Table 3: Techniques

Zhang et al. (2023)		Zimba et al. (2024)		Ensemble Model	
Actual data	Estimates	Actual data	Estimates	Actual data	Estimates
39.59	46.11	39.59	43.56	39.59	44.18
43.60	48.43	43.60	46.77	43.60	47.91
40.74	44.67	40.74	43.14	40.74	43.94
42.75	48.95	42.75	46.32	42.75	47.01
39.18	46.21	39.18	43.34	39.18	44.22
34.79	40.66	34.79	37.21	34.79	38.54
39.68	45.79	39.68	43.11	39.68	44.32
40.84	44.41	40.84	42.32	40.84	42.99
42.03	45.76	42.03	43.01	42.03	43.87
41.78	46.46	41.78	43.32	41.78	44.58
42.94	46.42	42.94	43.11	42.94	44.36
38.12	43.68	38.12	39.92	38.12	40.93
35.91	40.43	35.91	37.88	35.91	38.03
40.95	45.74	40.95	42.43	40.95	43.46
41.56	47.32	41.56	44.47	41.56	45.84
42.34	45.46	42.34	43.13	42.34	43.79
42.64	46.34	42.64	43.73	42.64	44.74
42.15	46.85	42.15	43.58	42.15	44.36
36.90	40.85	36.90	37.41	36.90	38.58
35.76	39.47	35.76	36.74	35.76	37.74
41.58	45.27	41.58	42.82	41.58	43.43
42.15	46.11	42.15	43.15	42.15	44.85
40.92	44.32	40.92	41.89		

Table 4 shows root mean square error (RMSE) for [28], Ensemble model, [30] to be 0.873, 0.596, and 0.704, respectively. The Mean Absolute Percentage Error (MAPE), a statistical metric used to measure the accuracy of forecasting models for the Ensemble model is 0.921%, which is the smallest compared to [28] and [30], with the MAPE values of 1.957% and 1.199%, respectively.

Table 4: Evaluation results

	Zhang *et al.* (2023)	**Ensemble model**	**Zimba *et al.* (2024)**
RMSE	0.873	0.596	0.704
MAPE (%)	1.957	0.921	1.199

The results show that Ensemble, which combines multiple machine learning models—often weak learners—to create a single, stronger, and more accurate predictive model, reducing variance, bias, and errors is a more efficient technique compared to [28] and [30], and the combination of the four methods considered will improve the accuracy of results obtained. This study proposes an Ensemble model to detect spam messages.

Conclusions

This study introduces an approach for filtering messages to enhance security the network. The objectives scope, data sources, and scope of this study are clearly outlined. The study includes the data collection, cleaning, the methods utilised, and the resulting findings. Overall, this research contributes to a safer, more productive, and enjoyable mobile messaging experience for users worldwide.

The knowledge gained from this study supports spam messages and promote a safer and more reliable mobile messaging experience. spam detection models. The study concludes that the Ensemble model achieved the highest accuracy based on the evaluation metrics used. The study has achieved its objectives by developing an effective spam detection system. The study's findings contribute to advancements in text classification and communication security, benefiting users and businesses worldwide. The knowledge gained from this study will help efforts to combat spam messages and promote a safer and more reliable mobile messaging experience. Additionally, this article presents an improved evaluation of spam detection models.

Authors' Contributions

All authors contributed to conception and design of this research paper. All authors read and approved the final manuscript. Conceptualisation: Ozoh Patrick; Methodology: Ozoh Patrick, Gbotosho Ajibola; Formal analysis and investigation: Ozoh Patrick, Gbotosho Ajibola; Writing –original draft preparation: Ozoh Patrick; Writing — review and editing: Gbotosho Ajibola, Abolarinwa Michael; Supervision: Ozoh Patrick.

Acknowledgements

The authors would like to thank the referees for their careful reading of the article and for the valuable comments provided.

Conflict of Interest Statement

The authors declare that there is no conflict of interest.

References

[1] Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine learning techniques for spam detection in email and IoT platforms: Analysis and research challenges. *Security and Communication Networks*, *2022*, Article 1862888. https://doi.org/10.1155/2022/1862888

[2] Airlangga, G. (2024). Optimizing SMS spam detection using machine learning: A comparative analysis of ensemble and traditional classifiers. *Journal of Computer Networks, Architecture and High Performance Computing*, *6*(4), 1942–1951. https://doi.org/10.47709/cnahpc.v6i4.4822

[3] Al-Kabbi, H. A., Feizi-Derakhshi, M. R., & Pashazadeh, S. (2023). Multi-type feature extraction and early fusion framework for SMS spam detection. *IEEE Access*, *11*, 123756–123765.

[4] Al-shanableh, N., Alzyoud, M. S., & Nashnush, E. (2024). Enhancing email spam detection through ensemble machine learning: A comprehensive evaluation of model integration and performance. *Communications of the IIMA*, *22*(1), Article 2. https://doi.org/10.58729/1941-6687.1451

[5] Bharadiya, J. P. (2023). A comparative study of business intelligence and artificial intelligence with big data analytics. *American Journal of Artificial Intelligence*, *7*(1), 24–30. https://doi.org/10.11648/j.ajai.20230701.14

[6] Bouke, M. A., Alramli, O. I., & Abdullah, A. (2025). XAIRF-WFP: A novel XAI-based random forest classifier for advanced email spam detection. *International Journal of Information Security*, *24*, Article 5. https://doi.org/10.1007/s10207-024-00920-1

[7] De Goma, J., Bravo, J. A., Prudente, S., & Rondilla, R. F. (2024, February). Detection of SMS spam messages using TF-IDF vectorizer and deep learning models. In *Proceedings of the 2024 9th International Conference on Intelligent Information Technology* (pp. 245–249).

[8] Ejirika, E. R., & Omotehinwa, T. O. (2024, April). Analysis of machine learning models for spam email detection and real-time integration. In *Proceedings of the International Conference on Science, Engineering and Business for Driving Sustainable Development Goals 2024 (SEB4SDG)* (pp. 1–10). IEEE.

[9] Gadde, S., Lakshmanarao, A., & Satyanarayana, S. (2021, March). SMS spam detection using machine learning and deep learning techniques. In *Proceedings of the 2021 International Conference on Advanced Computing and Communication Systems (ICACCS)* (Vol. 1, pp. 358–362). IEEE.

[10] Hadi, M. T., & Baawi, S. S. (2024, January). Email spam detection by machine learning approaches: A review. In *Proceedings of the International Conference on Forthcoming Networks and Sustainability in the AIoT Era* (pp. 186–204). Springer.

[11] Kalolo, C., & Mbelwa, J. (2023, November). Comparative analysis of machine learning models for detecting mobile messaging spam in Swahili SMS. In *Proceedings of the 2023 International Conference on the Advancements of Artificial Intelligence in African Context (AAIAC)* (pp. 1–7). IEEE.

[12] Kalyani, V. V., Sundari, M. R., Neelima, S., Prasad, P. S. S., Mohan, P. P., & Lakshmanarao, A. (2024, April). SMS spam detection using NLP and deep learning recurrent neural network variants. In *Proceedings of the 2024 International Conference on Cognitive Robotics and Intelligent Systems (ICC-ROBINS)* (pp. 92–96). IEEE.

[13] Mambina, I. S., Ndibwile, J. D., Uwimpuhwe, D., & Michael, K. F. (2024). Uncovering SMS spam in Swahili text using deep learning approaches. *IEEE Access*, *12*, 25164–25175.

[14] Naeem, M. Z., Rustam, F., Mehmood, A., Ashraf, I., & Choi, G. S. (2022). Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms. *PeerJ Computer Science*, *8*, Article e914.

[15] Nallamothu, P. T., & Khan, M. S. (2023). Machine learning for spam detection. *Asian Journal of Advances in Research*, *6*(1), 167–179.

[16] Oyeyemi, D. A., & Ojo, A. K. (2024). *SMS spam detection and classification to combat abuse in telephone networks using natural language processing*. arXiv. https://arxiv.org/abs/2406.06578

[17] Patel, D., Saxena, S., Verma, T., & Student, P. G. (2016). Sentiment analysis using maximum entropy algorithm in big data. *International Journal of Innovative Research in Science, Engineering and Technology*, *5*(5).

[18] Qazi, A., Hasan, N., Mao, R., Abo, M. E. M., Dey, S. K., & Hardaker, G. (2024). Machine learning-based opinion spam detection: A systematic literature review. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3417438

[19] Saeed, V. A. (2023). A method for SMS spam message detection using machine learning. *Artificial Intelligence & Robotics Development Journal*, *3*(1), 214–228. https://doi.org/10.52098/airdj.202366

[20] Srinivasarao, U., & Sharaff, A. (2023). Machine intelligence based hybrid classifier for spam detection and sentiment analysis of SMS messages. *Multimedia Tools and Applications*, *82*(20), 31069–31099.

[21] Wang, C., Pang, M., Wu, T., Gao, F., Zhao, L., Chen, J., Wang, W., Wang, D., & Zhang, P. (2024). Resilient massive access for SAGIN: A deep reinforcement learning approach. *IEEE Journal on Selected Areas in Communications*, *43*(1), 297–313. https://doi.org/10.1109/JSAC.2024.3460030

[22] Wang, L. (Ed.). (2005). *Support vector machines: Theory and applications* (Vol. 177). Springer Science & Business Media.

[23] Wang, L., Fan, M., Yang, N., Ma, X., Liang, Y., & Zhang, H. (2025). Toward intelligent space-air-ground integrated network: Architecture, challenges, and emerging directions. *Journal of Communications and Information Networks*, *10*(2), 87–102.

[24] Wu, C., Wang, X., Hu, Y., Han, S., Meng, W., & Niyato, D. (2025). Towards intelligent SAGIN: Leveraging big AI models and SDN for end-to-end automation. *IEEE Network*.

[25] Xie, S., Wang, G., Lin, S., & Yu, P. S. (2012, August). Review spam detection via temporal pattern discovery. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 823–831). ACM.

[26] Yin, Z., Luan, T. H., Cheng, N., Hui, Y., & Wang, W. (2022). *Cybertwin-enabled 6G space-air-ground integrated networks: Architecture, open issue, and challenges*. arXiv. https://arxiv.org/abs/2204.12153

[27] Zhang, Q., & Fiorella, L. (2023). An integrated model of learning from errors. *Educational Psychologist*, *58*(1), 18–34.

[28] Zhang, Y., & Guo, D. (2015). *Zhang functions and various models*. Springer.

[29] Zimba, A., Phiri, K. O., Kashale, C., & Phiri, M. N. (2024). A machine learning and natural language processing-based smishing detection model for mobile money transactions. *International Journal on Information Technologies & Security*, *16*(3).