**Journal of Information Systems Engineering and Management**
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

# Detecting Arabic SMS Scam Messages Using a Hybrid Ensemble Machine Learning Algorithms

Adnan H. Zawari[1], Bassma S. Alsulami[1], Fahad A. Alqurashi[1]
[1]*Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*
*azawri@stu.kau.edu.sa*

**ARTICLE INFO**

Received: 26 Dec 2024

Revised: 14 Feb 2025

Accepted: 22 Feb 2025

**ABSTRACT**

Short Messaging Service (SMS) remains a critical communication tool, with over five trillion messages sent globally each year. Despite the rise of internet-based messaging services, SMS retains its importance due to its high open rate, with 97% of messages read within fifteen minutes [1]. However, this ubiquity has also made SMS a prime target for scammers. SMS scams, commonly known as "smishing," pose significant threats, leading to widespread financial losses and privacy breaches for millions of users annually. This study proposes a hybrid ensemble learning model for Arabic SMS scam detection, integrating stacking and voting techniques to leverage multiple classifiers. A comprehensive dataset of scam and non-scam Arabic SMS messages was collected and preprocessed to ensure high-quality training data. The selected base models—Logistic Regression, Random Forest, and Gradient Boosting—were trained independently, and their outputs were combined through a meta-learner for final predictions. Experimental results show that the proposed model achieves 91.89% accuracy and a 91.55% F1-score, outperforming traditional classifiers and standalone ensemble models. This approach enhances detection accuracy and provides a more reliable solution for identifying scam messages in Arabic SMS communication.

**Keywords:** Arabic SMS Scam Detection, Hybrid Ensemble Learning Algorithms, Scam Message Text Classification

## 1. INTRODUCTION

Short Message Service (SMS) remains a widely adopted communication medium used across personal, commercial, and governmental domains. Its simplicity, low cost, and universal compatibility with mobile devices have made it a reliable channel for time-sensitive information [2]. However, the increasing reliance on SMS has also led to a surge in SMS-based scams and spam, where malicious actors exploit the medium to deceive users into revealing sensitive information or engaging in fraudulent transactions [3].

The personal nature of SMS communication and the widespread adoption of mobile devices make SMS an attractive attack vector for cybercriminals [4]. Additionally, the ease of spoofing sender identities and the lack of robust security mechanisms in traditional SMS infrastructure further exacerbate the risks associated with SMS scams [5]. Conventional detection methods, such as keyword-based filtering and sender reputation checks, have proven ineffective in combating the evolving tactics of scammers, who frequently alter their strategies to bypass these security measures [6].

Arabic presents significant linguistic challenges for natural language processing (NLP) and machine learning models. Its complex morphology, contextual word variations, and wide range of dialects require tailored preprocessing and feature extraction strategies. In addition, most existing research and datasets in SMS spam detection focus on English-language content, leaving Arabic-speaking users particularly vulnerable due to the lack of robust, localized detection systems and high-quality labeled data.

To address these challenges, this research proposes a machine learning-based Arabic SMS scam detection system using a hybrid ensemble learning approach. A dataset was developed by collecting Arabic SMS messages through

public participation and manually annotating them as scam or non-scam. Since many scam submissions were in image format, Optical Character Recognition (OCR) was used to extract their text. The final dataset was preprocessed through text normalization, stopword removal, and placeholder substitution to prepare it for effective model training.

The core of the proposed solution is a hybrid ensemble model that combines stacking and soft voting. First, a soft voting classifier is constructed using Logistic Regression, Random Forest, and Gradient Boosting. This voting classifier is then used as the meta-learner within a stacking classifier, allowing the model to benefit from both probability-based decision aggregation and layered learning. This structure enhances classification robustness and improves overall detection accuracy.

By addressing linguistic complexity, data scarcity, and model adaptability in Arabic SMS scam detection, this research contributes a scalable and practical solution to improving mobile communication security in Arabic-speaking regions.

The remainder of this paper is organized as follows: Section 2 reviews related work on SMS scam detection and machine learning techniques used in prior studies. Section 3 describes the data collection process. Section 4 presents the data preprocessing steps applied to prepare the dataset. Section 5 details the development of machine learning models, including the proposed hybrid ensemble. Section 6 presents the experimental results and discusses model performance. Finally, Section 7 concludes the paper and outlines potential directions for future work.

## 2. RELATED WORK

The widespread use of mobile communication has increased the risk of SMS scams, or "smishing," where attackers deceive users into disclosing sensitive information. Early detection systems relied on rule-based methods like keyword filtering and blacklisting but struggled to adapt to evolving scam tactics and obfuscation techniques.

To address these challenges, machine learning approaches have gained popularity for their ability to learn patterns from data and improve detection accuracy. Classifiers such as Logistic Regression, Support Vector Machine, Random Forest, AdaBoost, and Gradient Boosting have shown strong results in identifying scam messages. Ensemble methods like bagging, boosting, and stacking further enhance performance by combining multiple models, while deep learning techniques such as LSTM and transformer-based models like BERT offer state-of-the-art results, albeit with higher data and resource requirements.

Before the rise of machine learning, SMS scam detection relied on rule-based systems using predefined patterns, keyword filters, and sender metadata. Keyword filtering flagged messages with terms like “lottery” or “urgent,” but scammers quickly bypassed these using misspellings, special characters, or vague wording, requiring constant updates to remain effective [7]. Blacklisting and whitelisting filtered messages based on sender numbers or domains, but spammers used spoofed or frequently changing numbers to evade detection, and maintaining these lists was error-prone [8]. Sender reputation analysis introduced a more dynamic approach by evaluating sender behavior and user reports, yet it often suffered from delayed data accumulation and occasionally penalized legitimate senders [9]. While these traditional methods offered early protection, their static nature made them vulnerable to evolving tactics, highlighting the need for adaptive, data-driven techniques—ultimately leading to the adoption of machine learning models.

The increasing complexity of SMS scam tactics has driven the shift from static rule-based methods to adaptive, machine learning-based approaches capable of learning patterns from labeled data. Sjarif et al. [10] applied TF-IDF with Random Forest, achieving 97.5% accuracy, outperforming other models like SVM and Naïve Bayes. Similarly, Luo et al. [11] used the Kaggle SMS Spam Collection to compare Logistic Regression, Decision Tree, and KNN, with Logistic Regression yielding 99% accuracy and the lowest processing time. Gadde et al. [12] explored various vectorization techniques with both machine learning and LSTM, finding SVM and LSTM performed best across embeddings. Navaney et al. [13] confirmed SVM’s reliability, achieving 97.4% accuracy. Baaqeell and Zagrouba [14] tested Logistic Regression, SVM, AdaBoost, and a hybrid SVM-KMeans, where the hybrid achieved the highest accuracy. Airlangga [15] compared several classifiers and showed SVM 98.57% and an ensemble voting method 98.48% offered the best results, stressing the power of ensemble learning and the need to address class imbalance. Shirani-Mehr [16] found SVM with a linear kernel most effective 98.86%, followed by AdaBoost and Random Forest.

Journal of Information Systems Engineering and Management
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Ghourabi et al. [17] introduced a CNN-LSTM hybrid model trained on both English and Arabic SMS, achieving 98.37% accuracy, with Random Forest showing perfect precision but slightly lower recall. Oyeyemi and Ojo [18] used BERT embeddings with traditional classifiers and found Naïve Bayes + BERT most effective 97.31%, while Logistic Regression, Gradient Boosting, and Random Forest also performed well. Uddin et al. [19] compared RoBERTa with traditional models, noting RoBERTa achieved the highest performance 99.84% but at a greater computational cost. De Luna et al. [20] combined multiple datasets and found Logistic Regression 95.61% and Random Forest 95.23% slightly outperformed SVM and ensemble models. Finally, Gomaa [21] assessed machine learning and deep models on the UCI dataset, where RMDL achieved 99.26% accuracy, outperforming all others, though Gradient Boosting and Random Forest remained practical for their lower complexity.

## 3. DATA COLLECTION

To develop a robust Arabic SMS scam detection model, a dataset was collected through a survey designed to gather real-world SMS messages. The survey invited participants to submit screenshots or share the text of SMS messages they had received, whether they identified them as scams, non-scams, or were uncertain about their classification. It was distributed via social media platforms and direct messaging channels, enabling broad participation across diverse demographics. This approach increased dataset diversity and helped capture a wide range of scam patterns.

A total of 673 SMS messages were submitted by participants. The messages were manually categorized into two primary classes: (1) Scam messages, which exhibited clear fraudulent intent such as phishing attempts, impersonation, or fraudulent financial schemes, and (2) Non-scam messages, which were legitimate communications from banks, service providers, or personal contacts with no deceptive content. Participants were also allowed to submit messages they were uncertain about. These messages were initially labeled as “undefined” and later manually reviewed and categorized during preprocessing to ensure dataset integrity.

## 4. DATA PREPROCESSING

Effective preprocessing of Arabic SMS messages is essential for improving the accuracy and efficiency of machine learning models. This section details the preprocessing steps applied to the dataset, including text extraction, normalization, and feature refinement.

After manually reviewing and reclassifying all messages, the dataset was refined to ensure correct labeling. Messages initially classified as undefined were reassessed and assigned to either the scam or non-scam categories. Following this step and the removal of unhelpful messages, the dataset was reduced to 538 messages, comprising 40% scam and 60% non-scam content.

Since many participants submitted scam messages as screenshots, Optical Character Recognition (OCR) was used to extract Arabic and English text from the images [22]. The process involved three steps: image preprocessing with OpenCV to enhance clarity by adjusting contrast, removing noise, and binarizing the images; text extraction using Tesseract OCR configured for Arabic and English; and manual review to verify and correct the extracted text.

To ensure consistency and reduce data sparsity, several normalization techniques were applied. Arabic diacritics were removed, and letters with multiple forms were unified to standardize spelling variations. Common Arabic stopwords—terms that do not carry significant meaning—were removed using predefined lists [23], and all text was converted to lowercase for uniformity.

Non-alphabetic characters such as punctuation and symbols were eliminated unless contextually relevant, such as in URLs. Emojis were removed or mapped to sentiment labels when meaningful. Additionally, repeated letters were normalized to their base form to minimize noise and ensure consistent representation of words.

To enhance model generalization and prevent overfitting to specific identifiable data, sensitive components were replaced with placeholders. Phone numbers were replaced with PHONE_NUMBER, URLs with DOMAIN_NAME (retaining only the domain), monetary values with MONEY_AMOUNT, and one-time passwords (OTPs) with OTP_CODE. This substitution ensures the model learns generalizable linguistic and structural patterns rather than memorizing unique values.

Journal of Information Systems Engineering and Management
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Finally, for feature extraction, Term Frequency-Inverse Document Frequency (TF-IDF) was selected based on its strong performance in similar studies. TF-IDF evaluates the importance of a word within a document relative to a larger corpus and is calculated as follows:

- $\mathrm{TF(w)} = \frac{\text{(Number of times word } w \text{ appears in a document)}}{\text{(Total words in the document)}}$
- $\mathrm{IDF(w)} = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing } w}\right)$
- $\mathrm{TF-IDF(w)} = \mathrm{TF(w)} \times \mathrm{IDF(w)}$

This method assigns higher weights to terms that distinguish scam messages from legitimate ones, while reducing the influence of common, non-informative words [24].

## 5. MODEL DEVELOPMENT

The model development phase involved training and evaluating multiple machine learning algorithms to identify the best-performing models for Arabic SMS scam detection. The selection of these models was based on their demonstrated performance in prior literature, ensuring a strong foundation for detecting scam messages. A diverse set of classifiers was selected to compare different learning paradigms, including linear models, ensemble methods, and boosting techniques. Additionally, ensemble learning strategies, such as stacking and voting, were explored to improve classification performance.

### A. Standalone Machine Learning Models

Logistic Regression (LR) is a widely used linear model for binary classification. It estimates the probability that a given input belongs to a particular class using the sigmoid activation function [25]. LR was chosen as a baseline model due to its simplicity and efficiency in text classification tasks.

Support Vector Machine (SVM) is a powerful supervised learning algorithm that finds the optimal hyperplane to separate different classes by maximizing the margin between data points [26]. SVM is particularly effective in text classification due to its ability to handle high-dimensional spaces and small datasets.

Random Forest (RF) is an ensemble learning method that constructs multiple decision trees and aggregates their predictions to improve accuracy and reduce overfitting [27]. RF is known for its robustness and interpretability in classification tasks.

Gradient Boosting (GB) is a boosting algorithm that builds a strong classifier by sequentially improving weak learners [28]. It minimizes prediction errors iteratively, making it well-suited for handling complex patterns in textual data.

AdaBoost (AB) is another boosting technique that assigns higher weights to misclassified instances, forcing subsequent models to focus on difficult examples [29]. It enhances overall model performance by combining multiple weak classifiers.

### B. Ensemble Learning Models

Stacking Ensemble is a meta-learning technique that combines multiple base classifiers to enhance predictive performance. Unlike traditional ensemble methods such as bagging or boosting, stacking learns how to optimally combine multiple models by introducing a second-level model, known as the meta-learner, which makes the final prediction based on the outputs of the base learners [30]. In this study, the models chosen for stacking were selected based on their superior performance among the standalone classifiers, ensuring that only the best-performing models contributed to the ensemble. Specifically, Logistic Regression, Random Forest, and Gradient Boosting were used as base learners because they demonstrated the highest classification accuracy and balanced performance across precision, recall, and F1-score.

Voting Ensemble aggregates predictions from multiple classifiers and assigns the final label based on either majority voting (hard voting) or probability averaging (soft voting) [31]. The selected models for this ensemble were Logistic Regression, Random Forest, and Gradient Boosting, as they demonstrated the highest accuracy in individual

Journal of Information Systems Engineering and Management
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

evaluations. In soft voting, rather than each model making a fixed class decision, they generate probability distributions over the possible classes, and the final prediction is determined by averaging these probability scores and selecting the class with the highest combined probability. Additionally, by averaging predictions, soft voting enhances model stability and improves generalization, making the ensemble more reliable in real-world classification tasks.

The Hybrid Ensemble Model proposed in this study combines stacking and voting techniques to maximize predictive performance and enhance classification robustness. Figure 1 illustrates the structure of the proposed model, where the first level consists of a stacking classifier utilizing Logistic Regression, Random Forest, and Gradient Boosting as base learners. The predictions from these base models are then used to train the meta-learner, which applies soft voting aggregation to refine the final classification decision. This hybrid approach leverages the strengths of both stacking and voting: stacking enhances feature representation by combining diverse classifiers, while voting stabilizes the final decision through probability averaging. Stacking enables base models to independently learn decision boundaries—Logistic Regression captures linear relationships, Random Forest handles complex patterns using decision trees, and Gradient Boosting iteratively refines predictions to minimize errors. By integrating these diverse learning strategies, the stacking framework reduces individual model weaknesses and improves overall generalization. The meta-learner, functioning as a voting classifier, further enhances decision-making by assigning greater weight to models with higher confidence in their predictions. This reduces individual model biases, balances misclassifications, and improves the stability and reliability of the final classification output.

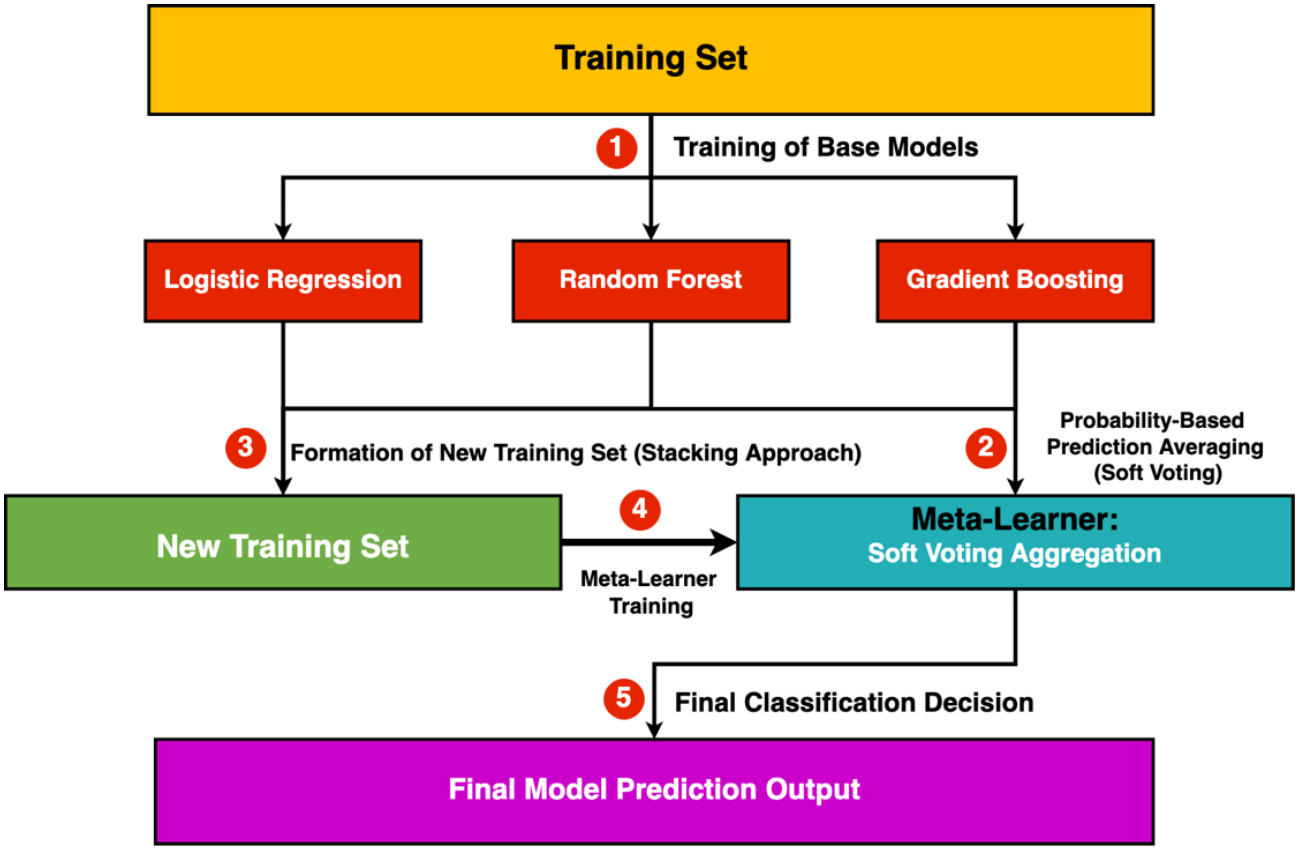


*Figure 1 Structure of the Hybrid Ensemble Model.*

## C. Model Optimization and Implementation

To enhance performance, hyperparameters were fine-tuned using Grid Search and Random Search. Grid Search exhaustively evaluates predefined parameter combinations, while Random Search explores a subset of the hyperparameter space to improve efficiency [32]. The dataset was split into 70% training and 30% testing data a commonly used ratio in text classification tasks with limited data to ensure both sufficient learning and reliable

**Journal of Information Systems Engineering and Management**
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

evaluation. All models were implemented using Scikit-learn, a widely adopted Python library that provides robust support for classification algorithms, ensemble learning, and hyperparameter tuning.

## 6. RESULTS AND DISCUSSION

To comprehensively evaluate the effectiveness of the models in detecting scam messages, several widely adopted performance metrics in text classification were employed. These include accuracy, precision, recall, F1-score, ROC-AUC score, and the confusion matrix, each offering insights into different aspects of model performance. Accuracy measures the proportion of correctly classified messages, both scam and non-scam, and is calculated using Equation (1), which considers true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Precision, defined in Equation (2), assesses how many messages predicted as scams are indeed scams. Recall, shown in Equation (3), evaluates the model's ability to identify actual scam messages. The F1-score, given by Equation (4), represents the harmonic mean of precision and recall, providing a balanced metric particularly useful in imbalanced datasets. The ROC-AUC score reflects the model's overall ability to discriminate between scam and non-scam messages. Additionally, the confusion matrix offers a clear breakdown of the model's prediction outcomes by displaying the counts of TP, FP, TN, and FN. Together, these metrics offer a well-rounded evaluation, accounting for both prediction accuracy and error types.

1) $Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$

2) $Precision = \frac{TP}{TP + FP}$

3) $Recall = \frac{TP}{TP + FN}$

4) $F1 - Score = 2 \times \frac{Precision\ X\ Recall}{Precision + Recall}$

Table 1 presents the detailed classification performance of all models across accuracy, precision, recall, F1-score, and ROC-AUC. The results clearly demonstrate that ensemble learning approaches outperform individual classifiers in detecting Arabic SMS scam messages. The proposed Hybrid Ensemble Model, which combines stacking and soft voting, achieved the best overall performance with an accuracy of 91.89%, a recall of 87.84%, and an F1-score of 91.55%, indicating its superior ability to generalize across different types of messages.

*Table 1 Classification performance of the five standalone machine learning models*

| Model | Accuracy | Precision | Recall | F1-Score | ROC-AUC |
|---|---|---|---|---|---|
| Logistic Regression | 84.46% | 90.48% | 77.03% | 83.21% | 95.76% |
| SVM | 80.41% | 94.12% | 64.86% | 76.80% | 91.49% |
| AdaBoost | 79.73% | 81.43% | 77.03% | 79.17% | 89.08% |
| Random Forest | 83.11% | 96.23% | 68.92% | 80.32% | 96.57% |
| Gradient Boosting | 83.78% | 90.32% | 75.68% | 82.35% | 92.26% |
| Hybrid Ensemble | 91.89% | 95.59% | 87.84% | 91.55% | 95.36% |

Among the standalone models, Logistic Regression, Random Forest, and Gradient Boosting showed competitive performance with accuracy scores of 84.46%, 83.11%, and 83.78%, respectively. These models also maintained a strong balance between precision and recall, making them reliable for real-world classification tasks. On the contrary, SVM and AdaBoost underperformed, especially in recall—64.86% and 77.03%, respectively—which reflects their difficulty in identifying scam messages effectively, possibly due to sensitivity to high-dimensional features and noise in the dataset.

**Journal of Information Systems Engineering and Management**
2025, 10(55s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

**Research Article**

Figure 2 further illustrates this, showing that SVM and AdaBoost misclassified more scam messages compared to other models, whereas the Hybrid Ensemble exhibited the most balanced and accurate confusion matrix, misclassifying only a few instances. The enhanced performance of stacking and voting ensembles individually also validated the benefits of combining diverse classifiers, but it was the Hybrid Ensemble that capitalized on both methods' strengths, yielding the most stable and robust outcomes.

The findings in this study align with previous research on SMS scam detection. Sjarif et al. [10] found that Random Forest achieved an accuracy of 97.50%, which is consistent with its strong performance in this study. Similarly, Luo et al. [11] demonstrated that Logistic Regression performed well for spam detection, achieving 99% accuracy, further supporting its effectiveness as observed in this study. Additionally, Airlangga [15] showed that ensemble learning models outperformed individual classifiers, reinforcing the superior performance of stacking and voting ensembles in this study. Furthermore, Gomaa [21] identified Gradient Boosting as one of the best standalone models, which aligns with this study's findings, where GB achieved a high accuracy of 83.78%.

However, some differences exist. Navaney et al. [13] reported that SVM achieved the highest accuracy of 97.40%, whereas in this study, SVM exhibited lower performance with 80.41% accuracy and 64.86% recall. This discrepancy may stem from differences in dataset size, the complexity of Arabic SMS messages, or variations in feature extraction methods. Additionally, Baaqeel and Zagrouba [14] demonstrated that hybrid models combining unsupervised and supervised learning significantly enhanced spam detection accuracy—a finding further validated in this study, as the Hybrid Ensemble Model achieved the highest overall performance.

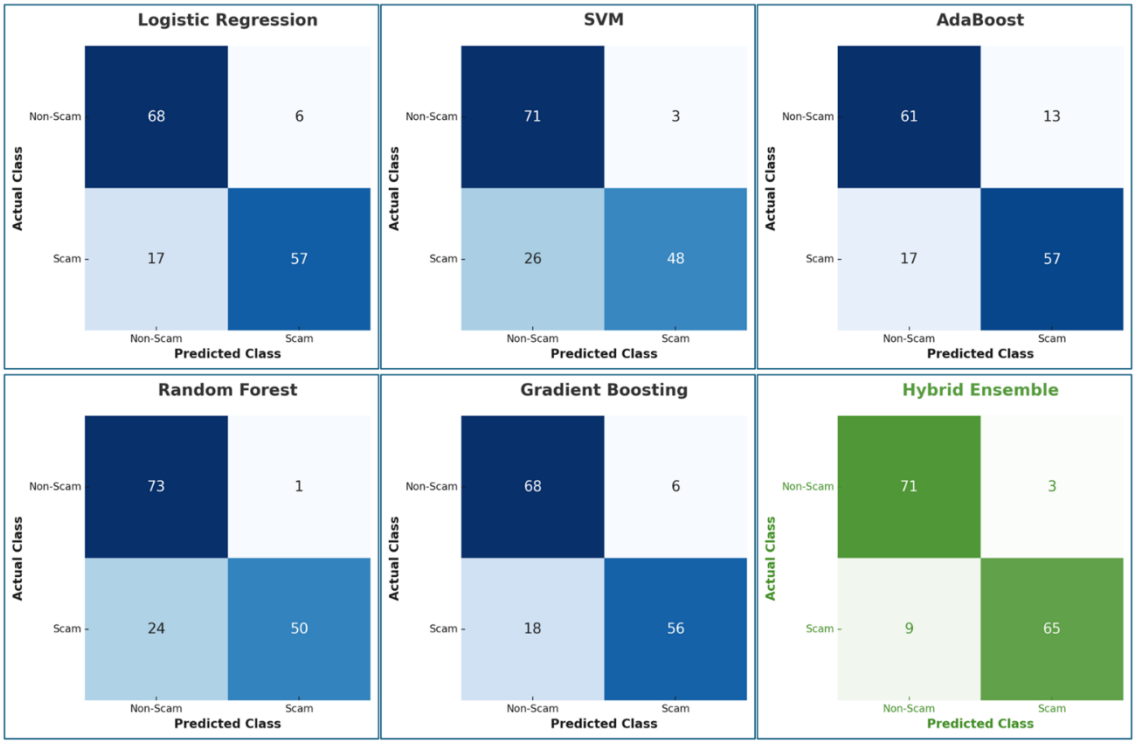


*Figure 2 Confusion Matrices of All Models*

## 7. CONCLUSION

This study explored Arabic SMS scam detection using both traditional machine learning and ensemble learning techniques. Through extensive experimentation with Logistic Regression, Support Vector Machine (SVM), AdaBoost, Random Forest, and Gradient Boosting, the findings revealed that ensemble models consistently outperformed standalone classifiers. Among the individual models, Logistic Regression, Random Forest, and Gradient Boosting achieved strong results, while SVM and AdaBoost showed lower generalization performance. The proposed Hybrid Ensemble Model, combining stacking and soft voting, delivered the best results—achieving an accuracy of 91.89%—demonstrating the effectiveness of integrating multiple classifiers for enhanced scam detection.

A key challenge encountered was the scarcity of publicly available Arabic SMS scam datasets. To overcome this, real-world data was manually collected with the help of community participation. The resulting dataset marks a significant contribution to Arabic NLP and cybersecurity research, offering a valuable resource for future work in this domain.

Looking ahead, future research could expand the dataset to include a broader variety of scam patterns, improving model generalizability. Additional features—such as sender metadata, timestamps, and URL analysis—could be integrated to provide a richer context for classification. Moreover, deploying the trained model within real-world applications, such as SMS filtering systems embedded in mobile devices or telecom infrastructures, would bring practical value to end-users. This study lays a solid foundation for Arabic SMS scam detection using hybrid ensemble learning, and future enhancements can further improve its accuracy, scalability, and impact on secure mobile communication.

## REFRENCES

[1] Drips. (2024). *SMS marketing and usage statistics*. Retrieved August 15, 2024, from https://www.drips.com/resources/sms-marketing-and-usage-statistics

[2] Sociocs. (2024). *The importance of SMS in customer communication*. Retrieved August 15, 2024, from https://www.sociocs.com/post/the-importance-of-sms-in-customer-communication

[3] Federal Trade Commission. (2022). *How to recognize and report spam text messages*. Retrieved August 15, 2024, from https://consumer.ftc.gov/articles/how-recognize-and-report-spam-text-messages

[4] HORISEN. (2021). *10 reasons to use SMS to communicate with customers*. Retrieved August 15, 2024, from https://www.horisen.com/10-reasons-to-use-sms-to-communicate-with-customers

[5] Bouilhac, B. (2024). *Smishing (SMS phishing): How to identify attacks and protect yourself*. Retrieved August 15, 2024, from https://www.vaadata.com/blog/smishing-sms-phishing-how-to-identify-attacks-and-protect-yourself/

[6] Salman, M., Ikram, M., & Kaafar, D. (2022). *An empirical analysis of SMS scam detection systems*.

[7] Abayomi-Alli, O. O., Onashoga, S. A., Sodiya, A. S., & Ojo, D. A. (2015). A critical analysis of existing SMS spam filtering approaches. In *Proceedings of the 1st International Conference on Applied Information Technology* (pp. 211–213).

[8] Purani, D., Bhavsar, N., & Shrimali, H. (2023). Advancements in spam SMS detection techniques - a review. *Journal of Emerging Technologies and Innovative Research (JETIR)*, 10(6), c19–c22.

[9] Salman, M., Ikram, M., & Kaafar, M. A. (2024). Investigating evasive techniques in SMS spam filtering: A comparative analysis of machine learning models. *IEEE Access*, 12, 24306–24312.

[10] Sjarif, N. N. A., Azmi, N. F. M., Chuprat, S., Sarkan, H. M., Yahya, Y., & Sam, S. M. (2019). SMS spam message detection using term frequency-inverse document frequency and random forest algorithm. *Procedia Computer Science*, 161, 509–515.

[11] Luo, G., Nazir, S., Khan, H., & Haq, A. (2020). Spam detection approach for secure mobile message communication using machine learning algorithms. *Security and Communication Networks*, 1–6.

[12] Gadde, S., Lakshmanarao, A., & Satyanarayana, S. (2021). SMS spam detection using machine learning and deep learning techniques. In *Proceedings of the 7th International Conference on Advanced Computing and Communication Systems (ICACCS)* (pp. 358–362).

[13] Navaney, P., Dubey, G., & Rana, A. (2018). SMS spam filtering using supervised machine learning algorithms. In *Proceedings of the 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 43–48).

[14] Baaqeel, H., & Zagrouba, R. (2020). Hybrid SMS spam filtering system using machine learning techniques. In *Proceedings of the 21st International Arab Conference on Information Technology (ACIT)* (pp. 1–8).

[15] 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.

[16] Shirani-Mehr, H. (2012). SMS spam detection using machine learning approach. *Proceedings of the Stanford University Research Project*, 1–10.

[17] Ghourabi, A., Mahmood, M. A., & Alzubi, Q. M. (2020). A hybrid CNN-LSTM model for SMS spam detection in Arabic and English messages. *Future Internet*, 12(156), 1–16.

[18] Oyeyemi, D. A., & Ojo, A. K. (2023). SMS spam detection and classification to combat abuse in telephone networks using natural language processing. *Journal of Advances in Mathematics and Computer Science*, 38(10), 144–156.

[19] Uddin, M. A., Islam, M. N., Maglaras, L., Janicke, H., & Sarker, I. H. (2024). ExplainableDetector: Exploring transformer-based language modeling approach for SMS spam detection with explainability analysis. *Preprint submitted to Elsevier*.

[20] De Luna, R. G., Enriquez, K. L., Española, A. M., Ramos, M., Magnaye, V. C., & Astorga, D. (2023). A machine learning approach for efficient spam detection in short messaging system (SMS). In *Proceedings of TENCON*.

[21] Gomaa, W. H. (2020). The impact of deep learning techniques on SMS spam filtering. *International Journal of Advanced Computer Science and Applications*, 11(1), 544–552.

[22] Smith, R. (2007). An overview of the Tesseract OCR engine. In *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR)* (pp. 629–633).

[23] Almutairi, T., Saifuddin, S., Alotaibi, R., Sarhan, S., & Nassif, S. (2024). Preprocessing techniques for clustering Arabic text: Challenges and future directions. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 15(8), 1–15.

[24] LearnDataSci. (2024). *TF-IDF (Term Frequency-Inverse Document Frequency)*. Retrieved February 19, 2025, from https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/

[25] Hosmer, D. W., & Lemeshow, S. (2000). *Applied logistic regression*. Wiley. https://doi.org/10.1002/0471722146

[26] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20, 273–297. https://doi.org/10.1007/BF00994018

[27] Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5–32. https://doi.org/10.1023/A:1010933404324

[28] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

[29] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

[30] Kalule, R., Abderrahmane, H. A., Alameri, W., & Sassi, M. (2023). Stacked ensemble machine learning for porosity and absolute permeability prediction of carbonate rock plugs. *Scientific Reports*, 13, Article 9855. https://doi.org/10.1038/s41598-023-36096-2

[31] Brownlee, J. (2021). *How to develop voting ensembles with Python*. Retrieved from https://machinelearningmastery.com/voting-ensembles-with-python/

[32] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13, 281–305.