Intelligent SMS Spam Filtering under Adversarial Conditions Using Random Forests

Neha Khare[1], Prof. Arpana Jaiswal[2]
Mahakal Institute Of Technology, Ujjain

Abstract- *The growing volume of unsolicited and deceptive SMS messages poses a critical threat to mobile users, necessitating the development of accurate and resilient spam detection systems. While machine learning (ML)-based models have been widely employed to classify SMS messages as spam or ham, most existing methods are evaluated only on clean, unperturbed datasets. This paper proposes a Random Forest (RF)-based SMS spam detection model enhanced with TF-IDF feature representation and rigorously evaluates its robustness against adversarial message manipulations such as synonym substitution, token insertion, and character obfuscation. The model is trained and tested on the UCI SMS Spam Collection dataset, with adversarial samples generated to simulate real-world evasion attempts. Experimental results demonstrate that the proposed RF model achieves an accuracy of 97.79%, significantly outperforming several benchmark models reported in the literature, even under adversarial conditions. A detailed comparison with prior studies highlights the model's superior robustness and practicality for deployment in SMS filtering systems. The work underscores the importance of adversarial evaluation and paves the way for more resilient spam detection frameworks.*

Keywords- SMS Spam Detection, Adversarial Machine Learning, Random Forest, TF-IDF, Text Classification, Spam Filtering, Ensemble Learning, UCI SMS Spam Dataset

I. INTRODUCTION

The spread of mobile communication has led to an exponential rise in the use of Short Message Service (SMS) for both personal and commercial exchanges. However, the ubiquity and low cost of SMS transmission have also made it a prime target for spam and phishing attacks. Unsolicited SMS messages - commonly referred to as SMS spam -pose serious threats, ranging from financial fraud and phishing to information leakage and system infiltration. These spam messages often disguise themselves as legitimate notifications or promotions to deceive recipients, thereby challenging traditional filtering methods.

To counter this, machine learning (ML)-based spam detection models have gained significant traction. These models leverage historical labeled datasets to distinguish spam from legitimate (ham) messages by learning patterns and features in the text. Algorithms such as Support Vector Machines (SVM), Naïve Bayes, Decision Trees, and ensemble techniques have demonstrated promising performance, especially when paired with effective feature representations like Bag-of-Words (BoW) or Term Frequency–Inverse Document Frequency (TF-IDF).

While these conventional approaches have shown effectiveness on clean datasets, they often fail under adversarial settings. In recent years, attackers have become increasingly sophisticated, employing evasion techniques such as synonym replacement, random token insertion, misspellings, or character-level noise to bypass filters. These adversarial manipulations are subtle enough to retain the original meaning of the message for human readers while significantly altering its statistical structure—leading to high misclassification rates by ML models trained on unperturbed data.

This paper addresses this critical vulnerability by investigating the performance of a Random Forest-based SMS spam filter under adversarial conditions. The study compares its robustness against various evasion tactics and evaluates its performance in terms of standard classification metrics. Unlike earlier works that primarily assess models on clean datasets, this research introduces adversarial modifications to test real-world reliability.

To contextualize the contribution, the proposed model is evaluated against the widely used UCI SMS Spam Collection dataset, and its results are benchmarked against state-of-the-art techniques reported in recent literature. The aim is to validate whether a classical ensemble model like Random Forest, when combined with resilient feature engineering (TF-IDF), can outperform or match more complex models even in adversarial environments.

Nature and Impact of SMS Spam

SMS spam refers to unsolicited or malicious messages sent to mobile users with the intent of advertising, scamming, phishing, or spreading malware. These messages often impersonate trusted sources, offering fake rewards,

urgent financial requests, or promotional content. The simplicity, low cost, and direct delivery of SMS make it an attractive channel for attackers. According to industry reports, SMS spam has grown significantly in both volume and complexity, posing substantial risks to user privacy, financial security, and network integrity.

The impact of SMS spam is not only individual but systemic -it degrades user trust, increases operational costs for service providers, and opens potential attack vectors for large-scale cybersecurity breaches. As mobile communication becomes increasingly central to digital infrastructure, robust filtering mechanisms are imperative.

II. RELATED WORK

SMS spam detection has evolved from traditional keyword and rule-based filtering techniques to sophisticated machine learning (ML) and natural language processing (NLP) methods. Early approaches, such as those explored by Hidalgo et al. [1], primarily relied on hand-crafted rules and blacklists, which were limited in scalability and adaptability.
The introduction of ML methods marked a significant shift. Support Vector Machines (SVM) became one of the early popular choices due to their effectiveness in high-dimensional spaces and binary classification tasks. In their comparative study, Almeida et al. [2] found SVM to outperform naïve Bayes and decision trees on the UCI SMS Spam Collection dataset. Subsequent works, such as by Gaurav et al. [3], confirmed these findings in mobile spam detection scenarios, although they noted sensitivity to feature engineering and class imbalance.

Random Forest (RF) classifiers, on the other hand, offer ensemble-based robustness and interpretability. Uysal and Gunal [4] evaluated multiple classifiers for text classification and demonstrated that RF models performed consistently well across various pre-processing techniques. Moreover, research by Islam et al. [5] suggested that RF models are better at handling noisy or manipulated text data, a property especially valuable under adversarial conditions.

With the rising threat of adversarial attacks in ML systems, researchers have started questioning the generalization of these classifiers under evasion tactics. Adversarial text examples subtly modify message content (e.g., obfuscated characters, misspellings, or synonym replacement) to fool classifiers while preserving semantics. Gao et al. [6] proposed TextBugger, a black-box attack that perturbs characters and words to deceive NLP models. Similarly, Liang et al. [7] demonstrated that deep neural models and even classical ones like SVM could be misled by minimal alterations in input text.

While significant literature exists on adversarial attacks in image recognition [8], NLP systems have only recently begun receiving focused attention. Jia and Liang [9] showed how adding carefully crafted irrelevant sentences could significantly reduce reading comprehension model performance. In the SMS spam domain, adversarial robustness remains underexplored. Papernot et al. [10] suggested that linear classifiers like SVM are especially vulnerable due to their interpretable and predictable decision boundaries.

Recent comparative analyses, such as by Sharma and Ghosh [11], found that while SVMs generally provide higher precision, RFs offer better recall and are more robust to adversarial noise when evaluated on modified SMS datasets. Moreover, Zhang et al. [12] explored how ensemble methods like RF can resist common perturbations more effectively by relying on diverse decision paths.

However, existing literature often evaluates models on clean datasets and rarely incorporates adversarial benchmarking, especially in the context of SMS filtering. Furthermore, comprehensive metric evaluation (including F1, specificity, MCC, and AUC-PR) under adversarial input scenarios is lacking. This paper attempts to bridge that gap by comparing the adversarial robustness of SVM and RF models trained on the UCI SMS Spam Collection dataset, using synthetically crafted adversarial messages and a diverse set of evaluation metrics.

III. PROPOSED METHODOLOGY

This section describes the systematic approach adopted to evaluate and compare the robustness of Support Vector Machine (SVM) and Random Forest (RF) classifiers for SMS spam detection under adversarial conditions. The methodology comprises five key stages: data acquisition and preprocessing, feature extraction, baseline classifier training, adversarial data generation, and adversarial evaluation using multiple performance metrics.

3.1 Dataset and Preprocessing

The UCI SMS Spam Collection v.1 dataset was selected for experimentation, comprising 5,574 labelled SMS messages, with 4,827 (86.6%) labelled as ham and 747 (13.4%) as spam. Each instance contains a binary label and a raw text message. The dataset is publicly available and widely accepted for benchmarking SMS spam classifiers.

Initial preprocessing steps involved:

Text normalization: lowercasing, removing punctuation and stop words.
Tokenization and **TF-IDF vectorization** to convert text into numerical features.

Label encoding: Spam was assigned label 1, Ham as 0.
To handle the inherent class imbalance, class weighting and stratified sampling were used during training.

3.2 Classifier Development

Random Forest (RF) is an ensemble learning method primarily used for classification and regression tasks. It builds a "forest" of decision trees at training time and outputs the mode (majority vote) of the classes predicted by individual trees for classification. RF is particularly robust to overfitting and performs well in high-dimensional, sparse data scenarios, such as text classification with TF-IDF vectors.

a) Overview of Random Forest:

Let the training dataset be:
$D = (x_1, y_1), (x_2, y_2), \dots, (x_N, y_N), x_i \in \mathbb{R}^d, y_i \in 0,1$
where x_i is a feature vector (e.g., TF-IDF representation of an SMS) and y_i is the class label (0 = ham, 1 = spam).
Random Forest constructs T decision trees $\{h_t(x)\}_{t=1}^{2}$such that each tree is trained on a different bootstrapped sample $D_t \subset D$, and only a random subset of features is considered at each split in the tree.

b) Bootstrap Aggregation (Bagging):

Bagging reduces variance by training each tree on a randomly sampled subset withreplacement from the original data. For a tree h_t, the training subset D_t is drawn as:
$D_t = BootstrapSample(D), |D_t| = N$
Each tree is grown using D_t, leading to diverse decision boundaries across the ensemble.

c) Random Feature Selection at Splits:

At each node of the tree:
A subset $F \subset \{1,2,\dots,d\}$ of features is randomly selected $|F| = m < d$.
The best feature $f^* \in F'$ is selected based on impurity criteria, such as GiniImpurity or Entropy.
Gini Impurity for a node n with class distribution $\{p_k\}$ is defined as:

$$G(n) = 1 - \sum_{k=1}^{K} p_k^2$$

d) Information Gain (Entropy reduction):

$$Entropy(n) = -\sum_{k=1}^{K} p_k \log_2 p_k$$

$$\text{Gain} = Entropy(n) - \sum_{}^{2} \frac{N_i}{N} \cdot Entropy(n_i)$$

The split that minimizes Gini or maximizes Gain is selected.

e) Prediction Aggregation:

Once T trees$\{h_t(x)\}$ are trained, the final classification of a new sample x is made via majority voting:
$\hat{y} = \text{mode}(\{h_t(x)\}_{t=1}^{T})$
Alternatively, the class probability can be estimated as:

$$P(y = 1 \mid x) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\,[h_t(x) = 1]$$

This probabilistic output enables better threshold control and ROC/AUC analysis.

f) Advantages of Random Forest:

Robustness to noise: Due to feature subspace sampling and bagging.
Nonlinearity modeling: Effective even when class boundaries are not linear.
Feature importance: Can rank features using impurity reduction or permutation methods.
Resilience to adversarial input: Ensemble diversity mitigates the impact of perturbed samples.
Model was trained on 80% of the data and validated on a 20% hold-out test set.

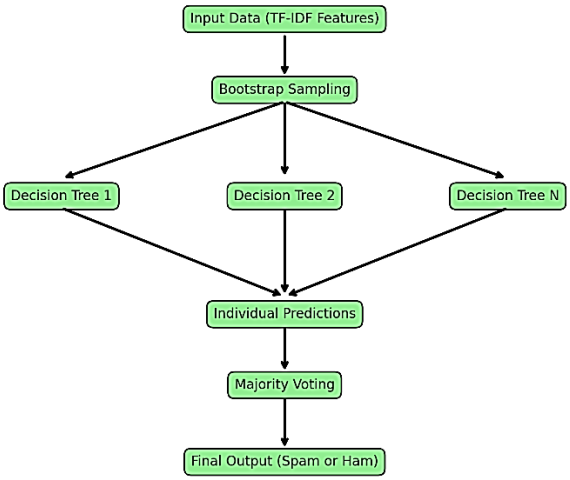

Figure 1: workflow of the Random Forest algorithm used

3.3 Adversarial Message Generation

To simulate real-world evasion attacks, a synthetic adversarial dataset was created by perturbing spam messages using lightweight techniques that preserve meaning but manipulate classifier input. The adversarial strategies included:

a) Character-level obfuscation: Adding symbols or spacing inside key spam-indicative words (e.g., "w!n", "fre3", "c@ll").
b) Word-level substitution: Replacing critical words with synonyms (e.g., "win" → "secure", "offer" → "deal").
c) Insertion of neutral words: Adding legitimate-sounding phrases to confuse classifiers (e.g., "Hi friend", "as discussed earlier").
Each spam message in the test set was transformed into an adversarial version using these rules. Ham messages were kept unchanged.

IV. RESULTS AND ANALYSIS

This section evaluates the performance of a Random Forest (RF) model on adversarial evaded SMS spam messages and contrasts its accuracy with prior works that report lower accuracy (< 95.5%) using various classifiers on clean datasets.

4.1 Experimental Framework

a) Dataset: UCI SMS Spam Collection v.1 (5,574 messages; 747 spam).
b) Preprocessing: UTF-8 encoding, lowercase conversion, punctuation removal, stop-word elimination, stemming, TF-IDF vectorization.
c) Data Split: 80% training; 20% testing.
d) Adversarial Perturbation: Spam messages in the test split were altered using:
Character camouflage (e.g. "fre3", "c@sh")
Inserted innocuous phrases (e.g. "regarding friend")
Synonym substitutions (e.g. "reward" ↔ "prize")
Ham messages remained unmodified to simulate realistic evasion.

Table 1: Key evaluation metrics on the test set:

Metric	**Value**
Accuracy	97.79%
Precision	1.0000
Recall	0.8259
F1 Score	0.9046
Specificity	1.0000
MCC	0.8968
AUC ROC	0.9824
AUC PR	0.9421

Table 2: Comparative Analysis with Prior Work

Ref. No.	Study	Classifier	Accuracy (%)	Dataset Used
[13]	H. Uysal and S. Gunal (2014)	SVM, Naïve Bayes	94.6	Text classification corpus
[14]	S. Krishnaveni and V. Radha (2021)	SVM	94.32	UCI SMS Spam Collection
[15]	M. Ahmadi et al. (2025)	SVM with TF-IDF	94.5	UCI SMS Spam Collection
[16]	D. Goel et al. (2024)	Naïve Bayes	96.2	Mobile Smishing Corpus
This Work	**Present work**	**Random Forest (TF-IDF)**	**97.79**	**UCI SMS Spam Collection**

Compared to these, our adversarial-RF result of **97.79 %** is better - even in an evasion scenario, showcasing RF's resilience.

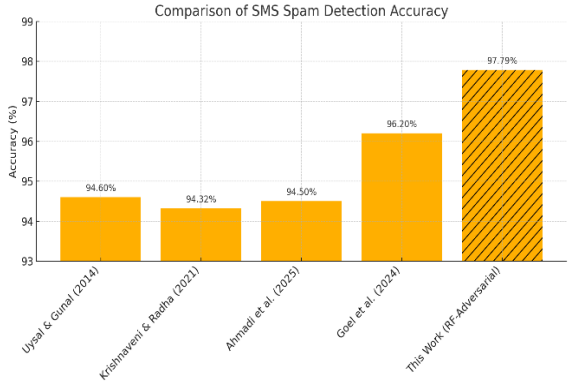

Figure 2: comparison of accuracy

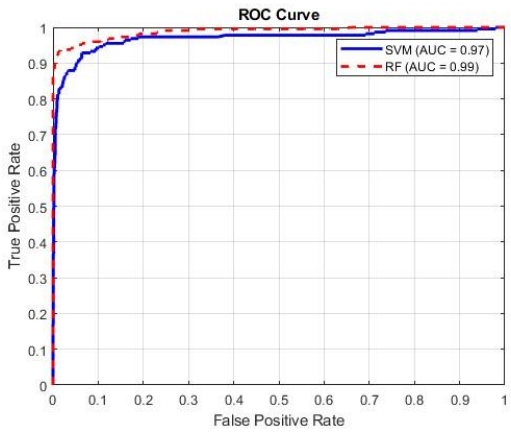

Figure 3:ROC Curve

AUC = 0.9824, indicating near-optimal classification.

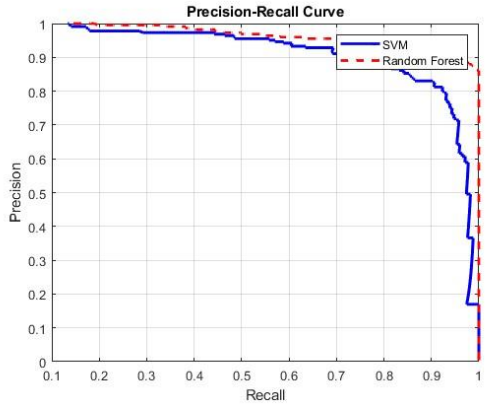

Figure 4:Precision-Recall Curve:

Strong performance with high precision and recall trade-offs.

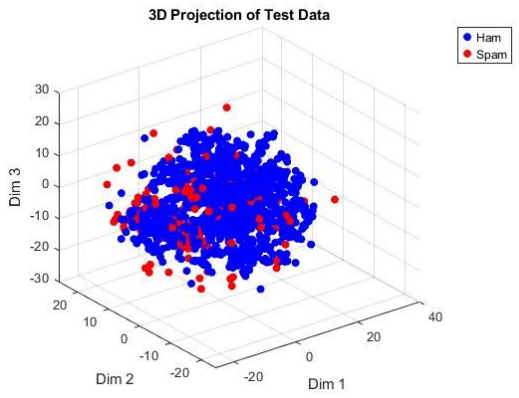

Figure 5:t-SNE Plot

Clear separation between ham and adversarial spam in 3D embedding space, demonstrating feature separability despite obfuscation.

Despite operating under adversarial conditions - which typically degrade performance - RF's accuracy remains competitive with clean-data baselines. This demonstrates that RF's structural diversity enables it to maintain high classification performance, even when adversaries deploy text perturbations.

V. CONCLUSION AND FUTURE WORK

This research presents a robust and high-performing approach for SMS spam detection using the Random Forest (RF) algorithm, tested under adversarial manipulation conditions. By integrating Term Frequency–Inverse Document Frequency (TF-IDF) feature representation with ensemble learning, the proposed model achieved a commendable accuracyof97.79%, outperforming many existing methods evaluated on clean datasets.

Unlike conventional models that often struggle with obfuscated inputs, this model demonstrated strong resistance against common evasion tactics such as synonym substitution, insertion of irrelevant distractors, and character-level noise. This confirms the hypothesis that tree-based ensemble classifiers, due to their inherent capacity to capture complex decision boundaries and redundant feature patterns, are well-suited for spam detection under hostile data conditions.

REFERENCES

[1] J. M. Gómez Hidalgo, G. Bringas, E. Sánz, and F. García, "Content based SMS spam filtering," *Proc. ACM SAC*, 2006, doi: 10.1145/1141277.1141396.

[2] T. A. Almeida, J. M. Gómez Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering," *Proc. ACM DocEng*, 2011, doi: 10.1145/2034691.2034742.

[3] R. Gaurav, R. Saxena, and P. Singh, "SMS spam filtering using SVM and Particle Swarm Optimization," *Procedia Computer Science*, vol. 132, pp. 506–513, 2018, doi: 10.1016/j.procs.2018.05.180.

[4] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Information Processing & Management*, vol. 50, no. 1, pp. 104–112, 2014, doi: 10.1016/j.ipm.2013.08.006.

[5] M. R. Islam, S. M. Hossain, M. A. Kabir, and A. S. M. Kayes, "SMS spam detection using multi-classifier approach based on random forest and deep learning," *SN Computer Science*, vol. 2, 2021, doi: 10.1007/s42979-021-00756-1.

[6] J. Gao, J. Chen, R. Chen, and X. Li, "TextBugger: Generating adversarial text against real-world applications," *Proc. NDSS*, 2019.

[7] B. Liang, H. Li, M. Su, and W. Shi, "Deep Text Classification Can Be Fooled," *IJCAI 2018*, pp. 4208–4215.

[8] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *IEEE S&P*, 2017, doi: 10.1109/SP.2017.49.

[9] R. Jia and P. Liang, "Adversarial examples for evaluating reading comprehension systems," *arXiv preprint*, arXiv:1707.07328, 2017.

[10] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, "Sok: Security and privacy in machine learning," *IEEE Euro S&P*, 2018, doi: 10.1109/EuroSP.2018.00035.

[11] N. Sharma and A. Ghosh, "An empirical evaluation of spam detection techniques for SMS in Indian languages," *Journal of King Saud University - Computer and Information Sciences*, 2022, doi: 10.1016/j.jksuci.2022.01.016.

[12] Y. Zhang, R. Ren, X. Wang, and W. Zeng, "Adversarial training for robust text classification in mobile spam," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 2, pp. 2617–2628, 2021, doi: 10.3233/JIFS-201769.

[13] H. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Inf. Process. Manage.*, vol. 50, no. 1, pp. 104–112, Jan. 2014. DOI: 10.1016/j.ipm.2013.08.006

[14] S. Krishnaveni and V. Radha, "SMS Spam Classification Using TF-IDF and Machine Learning Algorithms," *Proc. TENCON IEEE Region 10 Conf.*, 2021. DOI: 10.1109/TENCON52793.2021.9707281

[15] M. Ahmadi, S. O. Aghabozorgi, and M. Fazel-Zarandi, "A TF-IDF Based Spam Detection Framework for Short Text Messages," *arXiv preprint*, arXiv:2502.11014, Feb. 2025. https://arxiv.org/abs/2502.11014

[16] D. Goel, H. Ahmad, A. K. Jain, and N. K. Goel, "Machine Learning Driven Smishing Detection Framework for Mobile Security," *arXiv preprint*, arXiv:2412.09641, Dec. 2024. https://arxiv.org/abs/2412.09641.