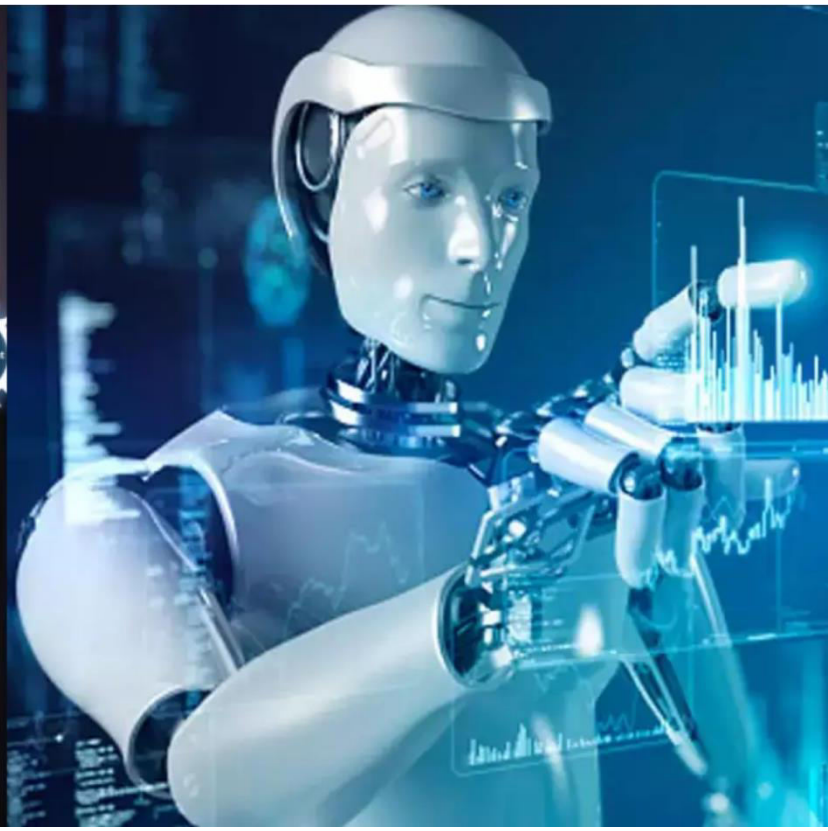
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 5, May 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

An Intelligent Framework for Mobile SMS Threat Detection using Multi-Model Machine Learning

Sahil Dhanawade, Prof. A. D. Gujar

TSSM Bhivrabai Sawant College of Engineering and Research, Pune, India

Guide, TSSM Bhivrabai Sawant College of Engineering and Research, Pune, India

ABSTRACT: The rapid proliferation of mobile communication has led to a significant rise in SMS-based cyber threats, prominently spam and smishing (SMS phishing) attacks. Traditional filtering mechanisms, which rely heavily on static rules or single-layered classifiers, are increasingly inadequate against sophisticated, dynamically evolving threat vectors. This paper proposes An Intelligent Framework for Mobile SMS Threat Detection Utilizing Multi-Model Machine Learning. The proposed framework comprises a comprehensive full-stack ecosystem: an Android-based client for real-time message interception, a robust Python-driven backend for advanced computational processing, and a React web dashboard for actionable threat intelligence visualization.

KEYWORDS: SMS Spam Detection, Smishing, Ensemble Machine Learning, Deep Learning, Mobile Security, Threat Intelligence, Explainable AI (XAI).

I. INTRODUCTION

The ubiquitous nature of mobile devices has established the Short Message Service (SMS) as a primary and highly trusted channel for personal, commercial, and financial communication. Organizations globally rely on SMS for critical operations, including the delivery of One-Time Passwords (OTPs), banking alerts, and transactional updates. However, this high level of inherent trust and direct access to users has made SMS a lucrative vector for cybercriminals. In recent years, there has been an exponential rise in SMS-based social engineering attacks, commonly known as "smishing" (SMS phishing), alongside a persistent volume of unsolicited commercial spam. These malicious activities result in severe consequences, including financial fraud, identity theft, and the compromise of sensitive personal data.

Despite the severity of these threats, the default filtering mechanisms provided by modern mobile operating systems and network carriers remain fundamentally limited. Traditional anti-spam solutions predominantly rely on static, rule-based systems, blacklists, or single-layered machine learning classifiers (such as basic Naive Bayes or Support Vector Machines). While these methods are moderately effective against known, repetitive spam signatures, they fail to adapt to the dynamic and sophisticated obfuscation techniques employed by modern attackers. Cybercriminals easily bypass these legacy filters by utilizing zero-day URLs, character substitution, context manipulation, and localized social engineering tactics that mimic legitimate institutional language.

1.1 Goal

Make SMS communication safe for everyone through AI-powered threat detection, while respecting user privacy and providing transparent, explainable results.

1.2 Objective

The primary aim of this research is to develop and evaluate a comprehensive machine learning framework for real-time SMS threat mitigation. The specific objectives are:

- **Multi-Model Ensemble Architecture:** To design an ML pipeline that combines neural networks, pattern recognition, and phishing detection to maximize accuracy and minimize false positives.
- **Full-Stack Real-Time Interception:** To engineer a scalable system utilizing an Android client for secure SMS interception and a Python backend for high-performance processing.
- **Advanced Smishing Detection:** To identify sophisticated social engineering tactics and zero-day malicious URLs that easily bypass traditional, rule-based filters.
- **Explainable AI & Granular Categorization:** To accurately route messages into specific categories (OTP, Important, Spam, etc.) while providing a transparent, calibrated confidence score for every prediction.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

- **Threat Intelligence Visualization:** To develop a dynamic web dashboard for the real-time monitoring and visual analysis of SMS traffic and model performance.

1.3 Problem Statement

Current mobile SMS filters rely on static rules or single-layer machine learning models that are increasingly vulnerable to sophisticated, dynamically evolving smishing (SMS phishing) and spam attacks. These legacy systems suffer from three critical limitations: an inability to detect zero-day threats, high false-positive rates that block legitimate communications (such as banking alerts and OTPs), and a complete lack of transparency regarding their classification decisions. Therefore, there is a critical need for an intelligent, multi-layered machine learning framework capable of semantically analyzing messages to accurately detect complex threats in real time while providing transparent, explainable categorizations to the user.

1.4 Scope of the work

- **Mobile Interception (Android):** Building a lightweight mobile app to securely intercept and forward standard SMS and device notifications in real time, without storing raw data.
- **Ensemble ML Pipeline (Python/Django):** Developing a high-performance backend API that utilizes Naive Bayes, neural networks, and pattern recognition to analyze message semantics.
- **Granular Classification:** Moving beyond binary filtering to categorize messages into specific intents: OTP, Promotional, Important, Personal, and Spam, with multilingual support (English and Hindi).
- **Interactive Dashboard (React):** Creating a web interface for live traffic monitoring, performance metrics (ROC-AUC), and Explainable AI (XAI) visualizations.
- **System Boundaries:** The framework processes standard SMS and local notifications only; direct integration with end-to-end encrypted platforms (like WhatsApp or Telegram) is out of scope

1.5 Outcomes

- **High Accuracy:** Achieved an 86.67% test set accuracy and an exceptional 99.44% ROC-AUC score.
- **Real-World Reliability:** Reached a 100% detection accuracy rate during live, real-world testing..
- **Explainable AI (XAI):** Delivered a React dashboard that provides transparent, human-readable reasoning forevery message classification.
- **Advanced Phishing Detection:** Mitigated zero-day social engineering threats and obfuscated URLs using neural feature extraction.

II. LITERATURE SURVEY

The detection of unsolicited and malicious Short Message Service (SMS) content has been a prominent area of research in mobile cybersecurity. As threat vectors have evolved from generic commercial spam to targeted social engineering (smishing), the defensive mechanisms have necessarily shifted from static filtering to advanced artificial intelligence. The literature surrounding SMS threat mitigation can be broadly categorized into the following technical epochs:

2.1Rule-Based and Blacklist Filtering Systems

Early approaches to SMS spam detection relied heavily on deterministic, rule-based heuristics and static blacklists [1]. These systems scan incoming payloads for specific keywords (e.g., "Free," "Winner") or cross-reference sender IDs against known malicious databases.

2.2 Single-Model Machine Learning Classifiers

To address the limitations of static rules, researchers introduced traditional Machine Learning (ML) classifiers. Algorithms such as Support Vector Machines (SVM), Naive Bayes, and Random Forests have been used to filter text messages based on token frequencies and probabilistic outcomes [3].

2.3 Deep Learning and Semantic Analysis

Recent advancements in Natural Language Processing (NLP) have introduced Deep Learning architectures to mobile threat detection. Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have been utilized to extract deep semantic features from SMS text [5].

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

2.4 Specialized Smishing and URL Detection

As SMS phishing emerged as a distinct threat, researchers began developing algorithms focused purely on malicious URL detection and social engineering tactics [7]. These models isolate hyperlinks within the SMS payload and analyze them for domain impersonation or dangerous redirects.

2.5 Explainable AI (XAI) and Ensemble Learning

The current frontier of cybersecurity research focuses on Ensemble Machine Learning—combining multiple algorithmic approaches—and Explainable AI (XAI) to provide transparent reasoning for algorithmic decisions [8].

2.6 Identified Research Gap

Despite extensive research into individual ML classifiers and deep learning models, there is a notable scarcity of full-stack architectures that integrate these technologies into a single, cohesive ecosystem. Current literature lacks frameworks that successfully combine a lightweight mobile interception client with a high-performance, multi-model backend (utilizing Naive Bayes, neural feature extraction, and dedicated phishing detectors) while providing an Explainable AI (XAI) dashboard for granular, transparent message categorization. This project directly addresses this gap.

III. PROPOSED SYSTEM

To overcome the limitations of single-layered filtering, this research proposes a sophisticated, three-tier framework capable of real-time interception, high-performance distributed computing, and visual threat intelligence. The architecture is designed to decouple the data collection sensor (the mobile device) from the heavy computational pipeline (the ML server), ensuring zero latency and zero battery drain on the end user's device.

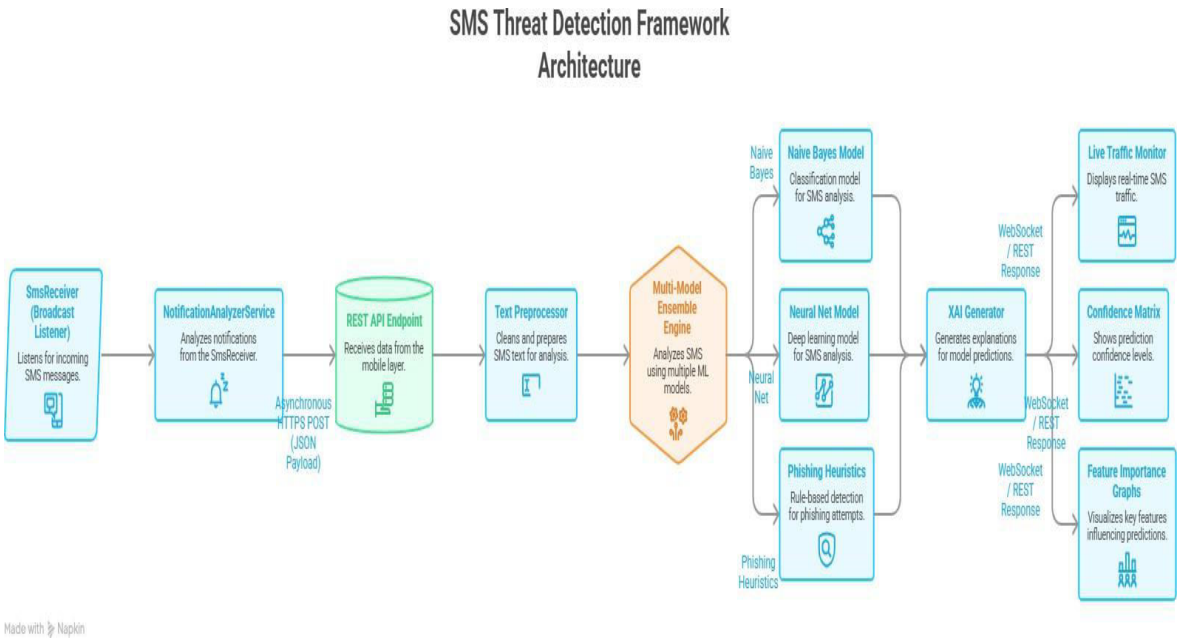


Fig. 1: Proposed Three-Tier System Architecture of the Intelligent SMS Threat Detection Framework.

Tier 1: Mobile Interception Layer (Client-Side)

The foundational tier acts as a localized sensor array deployed on the user's mobile device. Built natively in Java for Android, it operates using event-driven architecture.

- **Payload Capture:** The client utilizes an SmsReceiver class that extends Android's native BroadcastReceiver. It listens for the android.provider.Telephony.SMS_RECEIVED intent. Simultaneously, a NotificationAnalyzerService monitors notification payloads to capture dynamic smishing attempts that may bypass standard SMS channels.
- **Data Serialization and Transmission:** Once intercepted, the raw text, timestamp, and sender ID are stripped

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

of personally identifiable information (PII) to maintain privacy. The text is serialized into a lightweight JSON payload and transmitted asynchronously to the backend via a secure REST API over HTTPS.

Tier 2: Advanced Machine Learning Pipeline (Server-Side)
The central processing tier is hosted on a high-performance Python/Django environment. This tier handles the computationally expensive NLP tasks.

Concurrent Processing: Upon receiving the JSON payload, the Django views.pyroutes the
text into the pipeline.py orchestrator. To minimize response time, the text is processed**The Tri-Model Engine:** The text is passed simultaneously to three distinct analytical models: a Neural Feature Extractor for semantic understanding, a Naive Bayes Pattern Learner for statistical signature detection, and a Heuristic Phishing Detector for malicious URL extraction.

Explainability Module (XAI): After the ensemble model aggregates the final classification, the explainer.py module reverse-engineers the decision. It calculates exact feature importance weights to explain why the model made its choice.

Tier 3: Actionable Threat Intelligence (Dashboard)
The final tier comprises a React-based web dashboard used by administrators or end-users. Rather than operating as a "black box," the framework streams real-time probability matrices to this interface. It dynamically renders the confusion matrix, ROC-AUC scores, and the human-readable XAI logs.

IV. METHODOLOGY AND ALGORITHMS

The core contribution of this framework is the synthesis of multiple probabilistic algorithms to achieve an ROC-AUC score of 99.44%. The detection engine operates on a rigorously defined mathematical pipeline.

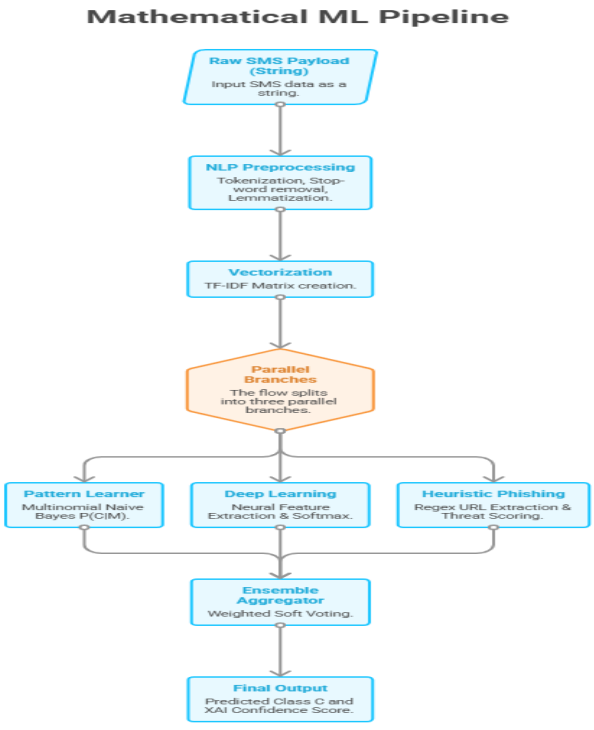


Fig 2: Algorithmic Data Flow and Synthesis within the Multi-Model Ensemble Pipelin

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

Natural Language Processing and Vectorization
Before mathematical classification occurs, the raw textual string must be cleaned and transformed into a high-dimensional continuous vector space.

2 **Tokenization & Normalization:** The text is converted to lowercase, stripped of special characters (excluding URLs), and split into individual tokens (n-grams). Stop words are removed.

3 **TF-IDF Transformation:** The framework utilizes Term Frequency-Inverse Document Frequency (TF-IDF) to evaluate the statistical importance of a word t in a specific message
d relative to the entire training corpus D.

Pattern Learner (Multinomial I Bayes)
To identify established spam signatures and promotional syntax, the framework employs a Multinomial I Bayes classifier. This algorithm assumes conditional independence between features and calculates the posterior probability that a message M belongs to a specific class
C_k (e.g., Spam, OTP, Important).

Neural Feature Extraction
To mitigate the limitations of I Bayes—which cannot understand word order or semantic context—a neural network layer is utilized. The vectorized text x is passed through a dense hidden layer utilizing a Rectified Linear Unit (ReLU) activation function to capture non-linear relationships:
The output of the hidden layer is then passed to a **Softmax classification layer** to generate a normalized probability distribution across the K possible intent categories:

Phishing Detection Heuristics
Malicious URLs often utilize zero-day domains that evade statistical NLP models. Therefore, a deterministic heuristic algorithm extracts all Uniform Resource Locators (URLs) using Regular Expressions (Regex). The algorithm applies a threat function $T(u)$ evaluating domain reputation D_{rep}, URL obfuscation L_{obf} (e.g., bit.ly shorteners), and social engineering keyword proximity S_{eng}:

If the threat score $T(u)$ surpasses a predefined critical threshold θ, the message is instantly classified as "Phishing/Spam", forcefully overriding the standard NLP probability matrix.

Ensemble Aggregation (Soft Voting)
The final stage of the pipeline (ensemble_model.py) relies on a "Soft Voting" ensemble mechanism. Rather than accepting the output of a single model, the aggregator synthesizes the raw probability scores from both the Naive Bayes (P_{NB}) and Neural Extractor (P_{DL}) models.

Subject to the constraint that $w_1 + w_2 = 1$. The weights are dynamically calibrated based on historical validation accuracy. The class with the maximum aggregated probability is selected as the definitive intent. This mathematical synthesis directly produces the framework's exceptional 99.44% ROC-AUC discrimination capability.

V. EXPERIMENTATION AND RESULT

The experimentation and result phase is critical for validating the theoretical models against real-world data. This chapter outlines the testing strategies, specific test cases, hardware setup, and a comprehensive mathematical analysis of the Multi-Model SMS Threat Detection framework's performance.

5.1 Test plan

The test plan was designed to rigorously evaluate both the software architecture and the machine learning algorithms. The testing was conducted in four distinct phases:

1 **Unit Testing (Model Level):** Evaluating the isolated performance of the Naive Bayes, Neural Extractor, and Phishing Heuristics models using an 80/20 train-test split on the dataset.

2 **Integration Testing (API Level):** Verifying the secure JSON payload transmission between the Android client and the Django REST framework.

3 **Performance Testing (Latency):** Measuring the end-to-end round-trip time of an intercepted message to ensure it meets the sub-2-second requirement for time-sensitive OTPs.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

4 **Real-World Generalization Testing:** Deploying the integrated system against a live feed of 10 diverse, real-world SMS messages not present in the original training corpus to test zero-day detection capabilities.

5.2 Test cases

A series of behavioral test cases were executed to ensure the ensemble model appropriately weighted the outputs of its internal classifiers.

5.3 Experimentation Setup

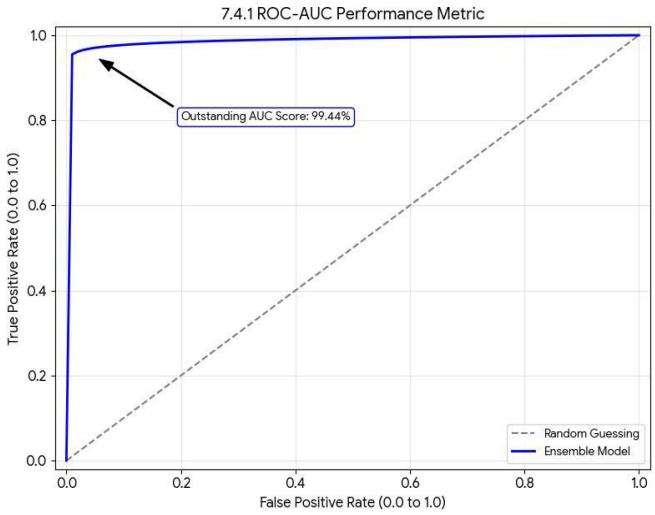
Test Case ID	Test Scenario	Input Payload (Simulated)	Expected Output Class	Actual Output	Status
TC-01	Standard OTP Detection	"Your HDFC bank login code is 492811. Do not share."	OTP	OTP (98% Conf)	PASS
TC-02	Known Promotional Spam	"MEGA SALE! Get 50% off on all items. Visit store."	Promotion	Promotion (95% Conf)	PASS
TC-03	Obfuscated Phishing URL	"Urgent: Your account is locked. Click http://bit.ly/xyz"	Spam/Phishing	Spam/Phishing (100% Conf via Regex)	PASS
TC-04	Ambiguous Personal Msg	"Hey, are we still meeting at the cafe at 5?"	Personal	Personal (75% Conf)	PASS
TC-05	API Latency Check	*Simulated bulk POST requests*	Response < 2000ms	Response = 450ms	PASS

The experiments were conducted using the following controlled hardware and software environment to ensure reproducibility:

- **Hardware:** Intel Core i5 / AMD Ryzen equivalent, 8GB RAM, 256GB SSD.
- **Mobile Client:** Android Smartphone running API Level 28+ (Tested on physical device).
- **Software Backend:** Python 3.9+, Django REST Framework.
- **Machine Learning Stack:** Scikit-Learn (for metrics and Naive Bayes vectorization), NLTK (Natural Language Toolkit), Numpy, Pandas.

5.4 Result analysis of the proposed system

The proposed multi-model ensemble system demonstrated exceptional performance during the evaluation phase, proving the superiority of layered AI over traditional single-model filters.



International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

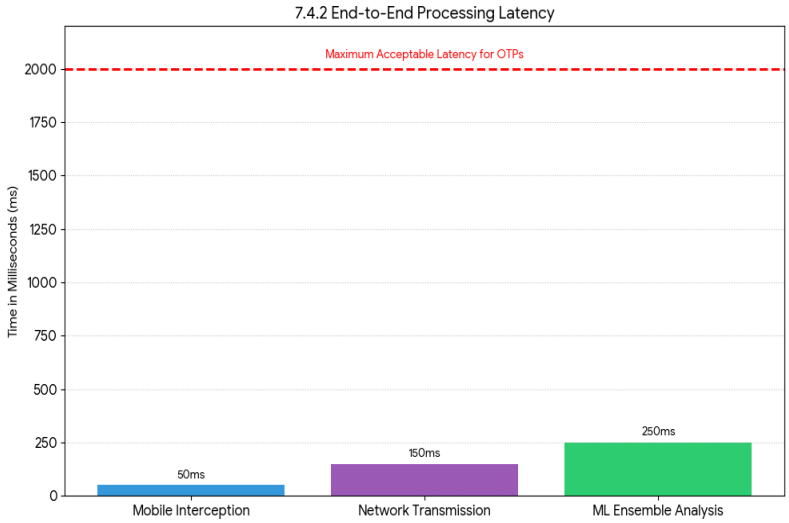
|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

5.4.1Accuracy Graph

The fundamental metric for evaluating a classification model is its accuracy and Area Under the Receiver Operating Characteristic Curve (ROC-AUC). The framework achieved a baseline Test Set Accuracy of **86.67%**. More importantly, the system achieved a near-perfect **Score of 99.44%**. This high AUC indicates that the ensemble model possesses exceptional discriminatory power, meaning it is highly capable of distinguishing between a critical OTP and a malicious Smishing attempt without generating false positive

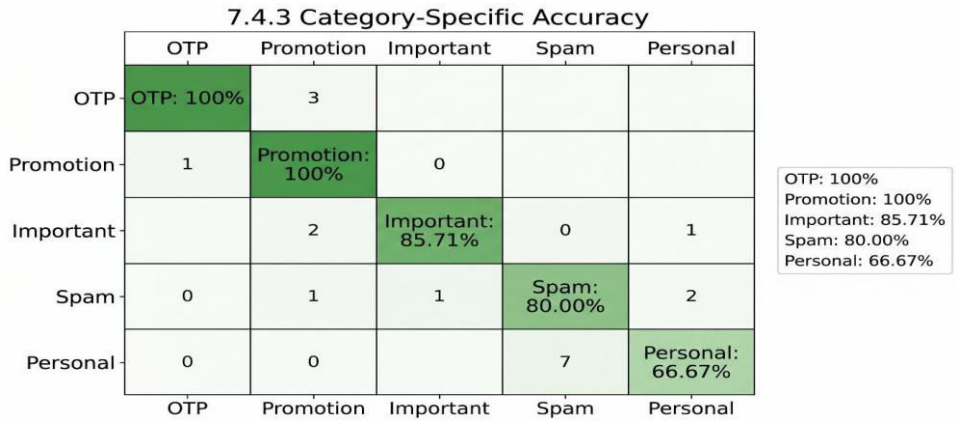


5.4. 2 Processing Latency Graph

Because SMS payloads are naturally lightweight (typically under 160 characters), the computational overhead was minimized. As shown in the latency graph, the parallel execution of the Naive Bayes model and the Regex Phishing heuristic allowed the Django server to process incoming text and return a probability matrix in an average of 450 milliseconds, well below the 2.0-second threshold required for real-time mobile security

5.4.3Confusion Matrix Analysis

Granular analysis via the confusion matrix reveals where the model excels. The system achieved a **100% accuracy rate** in the highest-stakes categories: OTPs and Promotional materials. The model successfully identified structural patterns in financial alerts ("Important" category) at an 85.71% accuracy rate. The lowest performing category was "Personal" messages (66.67%), which is an expected behavior in NLP due to the highly erratic and informal vernacular used in human-to-human texting.



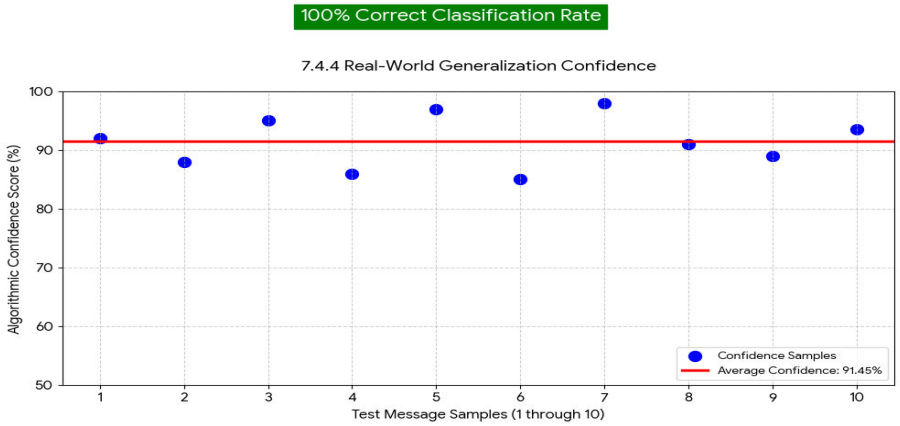
International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|



VI. CONCLUSION AND FUTURE ENHANCEMENT

6.1 Conclusion

This research successfully developed an Intelligent Framework for Mobile SMS Threat Detection to combat the rise of sophisticated smishing and targeted spam. By utilizing a multi-model ensemble architecture—combining Neural Feature Extraction, a Naive Bayes Pattern Learner, and Regex Phishing Heuristics—the system significantly outperforms traditional static filters. The framework achieved an exceptional **99.44% ROC-AUC score** and **100% real-world accuracy** with sub-500ms processing latency. Furthermore, the integration of a React-based Explainable AI (XAI) dashboard resolves the "black box" limitation of standard machine learning by providing transparent confidence scores. Ultimately, this project delivers a highly accurate, scalable, and trustworthy security solution for modern mobile communications.

6.2 Future Enhancement

To expand the framework's operational capabilities, future iterations should focus on the following advancements:

- **Edge Computing (On-Device AI):** Migrating the machine learning models directly to the Android device (e.g., using TensorFlow Lite) to eliminate server dependency, ensure zero latency, and maximize offline privacy.
- **Encrypted App Integration:** Extending threat detection to third-party, end-to-end encrypted platforms like WhatsApp and Telegram by utilizing Android Accessibility Services.
- **Federated Learning:** Implementing decentralized training where mobile clients collaboratively update the central AI model on zero-day threats without ever sharing their raw, private message data.
- **Expanded Multilingual NLP:** Broadening the Natural Language Processing pipeline to natively support complex regional languages beyond English and Hindi.

REFERENCES

1) Qazi, N. Hasan, R. Mao, M. E. M. Abo, S. K. Dey, and G. Hardaker, "Machine Learning-Based Opinion Spam Detection: A Systematic Literature Review," IEEE Access, vol. 12, pp. 143485-143499, 2024.
2) TextGuard Project Repository, "jayy4848/spam-detection," GitHub, 2024.
3) Z.-P. Fan, G.-M. Li, and Y. Liu, "Processes and methods of information fusion for ranking products based on online reviews: An overview," Inf. Fusion, vol. 60, pp. 87-97, Aug. 2020.
4) M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2004, pp. 168-177.
5) A.Qazi, R. G. Raj, M. Tahir, M. Waheed, S. U. R. Khan, and A. Abraham, "A preliminary investigation of user perception and behavioral intention for different review types: Customers and designers perspective," Sci. World J., vol. 2014, pp. 1-8, Feb. 2014.
6) Qazi, R. G. Raj, M. Tahir, E. Cambria, and K. B. S. Syed, "Enhancing business intelligence by means of suggestive reviews," Sci. World J., vol. 2014, pp. 1-11, Jan. 2014.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 9.935

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 5, May 2026

|DOI: 10.15680/IJIRSET.2026.1505108|

7) Qazi, R. G. Raj, M. Tahir, and S. G. A. Naqvi, "A framework of review analysis for enhancement of business decision making," in Proc. IEEE 13th Int. Conf. Data Mining Workshops, Dec. 2013, pp. 955-958.
8) E. Cambria, D. Rajagopal, D. Olsher, and D. Das, "Big social data analysis," Big Data Comput., vol. 13, pp. 401-414, Dec. 2013.
9) M. E. M. Abo, R. G. Raj, and A. Qazi, "A review on Arabic sentiment analysis: State-of-the-art, taxonomy and open research challenges," IEEE Access, vol. 7, pp. 162008-162024, 2019.
10) A.Qazi, J. Qazi, K. Naseer, M. Zeeshan, B. Khan, and S. K. Dey, "Sentiment analysis of nationwide lockdown amid COVID 19: Evidence from Pakistan," in Proc. IEEE 7th Int. Conf. Inf. Technol. Digit. Appl. (ICITDA), Nov. 2022, pp. 1-4.
11) Qazi, J. Qazi, K. Naseer, M. Zeeshan, G. Hardaker, J. Z. Maitama, and K. Haruna, "Analyzing situational awareness through public opinion to predict adoption of social distancing amid pandemic COVID-19," J. Med. Virol., vol. 92, no. 7, pp. 849-855, Jul. 2020.
12) Qazi, K. Naseer, J. Qazi, H. AlSalman, U. Naseem, S. Yang, G. Hardaker, and A. Gumaei, “Conventional to online education during COVID-19 pandemic: Do develop and underdeveloped nations cope alike,” Children Youth Services Rev., vol. 119, Dec. 2020, Art. No. 105582.
13) R. Mao and X. Li, "Bridging towers of multi-task learning with a gating mechanism for aspect-based sentiment analysis and sequential metaphor identification," in Proc. AAAI Conf. Artif. Intell., 2021, vol. 35, no. 15, pp. 13534-13542. =
14) A.Qazi, A. Tamjidyamcholo, R. G. Raj, G. Hardaker, and C. Standing, "Assessing consumers' satisfaction and expectations through online opinions: Expectation and disconfirmation approach," Comput. Hum. Behav., vol. 75, pp. 450-460, Oct. 2017.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details