AI & CYBER FORUM
AN INTERNATIONAL JOURNAL
OPEN ACCESS
ISSN 3117-6399
Review Article
Check for updates

AI-Driven Framework for Real-Time Multichannel Cyber Scam Detection and Alerting

Tabasum Guledgudd*, Preeti Savant, Sanjana Narayan Naik, Saniya Sullad, and Shivani Vaddar

Department of Computer Science and EngineeringA.G.M Rural College of Engineering and Technology, Varur, Hubballi, Karnataka, India

Corresponding author: **Tabasum Guledgudd** | E-mail: **tabuag85@gmail.com**

Citation: Tabasum Guledgudd, Preeti Savant, Sanjana Narayan Naik, Saniya Sullad, and Shivani Vaddar (2026). AI-Driven Framework for Real-Time Multichannel Cyber Scam Detection and Alerting. *AI & Cyber Forum: An International Journal.*
DOI: https://doi.org/10.51470/AI.2026.5.1.34

Received 13 January 2026 | Revised 16 February 2026 | Accepted 11 March 2026 | Available Online April 19, 2026

Abstract

Cyber scams are rapidly rising across multiple digital platforms, including SMS, emails, URLs, QR codes, and Unified Payments Interface (UPI) transactions, causing massive financial losses worldwide. Existing solutions address each channel in isolation, lack real-time detection, and are ineffective for multilingual environments. This paper proposes the Smart System: a unified, intelligent, real-time framework for detecting and alerting cyber scams across multiple channels simultaneously. The system integrates Machine Learning (ML) algorithms such as Naive Bayes (NB), Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR) with deep learning models including Long Short-Term Memory (LSTM) and Bidirectional Encoder Representations from Transformers (BERT). Natural Language Processing (NLP) techniques with TF-IDF and Count Vectorizer feature extraction are applied for text processing. Multilingual support through translation APIs ensures wider coverage. A Flask-based web interface provides real-time user alerts. Experimental results demonstrate an accuracy exceeding 97% for SMS phishing detection and 98% for email spam classification. The proposed unified system surpasses existing single-channel solutions in accuracy, scalability, and real-world applicability.

Keywords: *Cyber scam detection, SMS phishing, email spam, UPI fraud, NLP, LSTM, BERT, Random Forest, SVM, TF-IDF, real-time alert system, multichannel cybersecurity.*

I. INTRODUCTION

The rapid expansion of digital technology has transformed everyday life, enabling billions of users worldwide to communicate, conduct financial transactions, and exchange information over electronic platforms. However, this digital revolution has also created a fertile ground for cybercriminals who exploit various channels to commit scams, phishing attacks, and financial fraud. Today, cyber scams are no longer confined to a single platform. They appear across SMS (Short Message Service), emails, malicious URLs, tampered QR codes, and digital payment systems such as the Unified Payments Interface (UPI) [1].

According to the Federal Bureau of Investigation Internet Crime Report 2024, cybercrime losses exceeded USD 12.5 billion globally, with phishing and smishing representing the most frequently reported crime types [2]. In India, the National Payments Corporation of India (NPCI) reported a significant rise in UPI-related fraud incidents, including fake UPI handles, social engineering attacks, and QR code tampering [3]. Despite the growing scale of the problem, existing solutions typically address only one channel at a time.

An SMS spam filter does not protect against email phishing [4]; a UPI fraud detector does not help against smishing. Users are therefore left vulnerable on multiple fronts simultaneously.

Furthermore, existing systems suffer from additional shortcomings. Most are trained exclusively on English-language datasets, making them ineffective in India's multilingual environment, where users communicate in Hindi, Kannada, Tamil, Telugu, and many other languages. Additionally, most solutions are not capable of operating in real time, which means a scam message may already have been acted upon before it is flagged. The lack of a unified intelligent platform that can monitor, detect, and alert users across all these digital channels constitutes a significant gap in the cybersecurity landscape.

To address these challenges, this paper proposes the Smart System: a unified, AI-powered, real-time cyber scam detection and alerting platform. The system integrates classical machine learning classifiers, including Naive Bayes, SVM, Random Forest, and Logistic Regression, with state-of-the-art deep learning architectures such as LSTM and BERT. NLP pre-processing pipelines handle text cleaning, tokenization, and feature extraction using TF-IDF and Count Vectorizer.

A multilingual translation layer ensures that non-English messages are processed accurately. The system is deployed as a lightweight Flask-based web application that provides instant, real-time alerts to end users.

ALGORITHMS DISCUSSION

The proposed System employs a carefully selected ensemble of classical machine learning, deep learning, and probabilistic models to achieve robust multichannel scam detection. Multinomial Naive Bayes (MNB) serves as a lightweight probabilistic baseline classifier, leveraging word frequency distributions to achieve 96–98% accuracy with minimal computational overhead, making it well-suited for real-time SMS and email screening. Support Vector Machine (SVM) constructs an optimal separating hyperplane in high-dimensional TF-IDF feature space, delivering 97–98% accuracy on SMS corpora while remaining resistant to overfitting. Random Forest (RF) aggregates predictions from multiple decision trees trained on random feature subsets, providing robust classification for heterogeneous URL features alongside built-in interpretability through feature importance scores. Logistic Regression (LR), expressed as $P(Y=1|X) = 1/(1+e^{-z})$, achieves 98.82% email spam accuracy with hyperparameter-tuned coefficients that offer transparent feature-level explainability.

On the deep learning front, Long Short-Term Memory (LSTM) networks capture long-range sequential dependencies in text through input, forget, and output gating mechanisms — detecting subtle contextual patterns that bag-of-words models miss. BERT (Bidirectional Encoder Representations from Transformers), fine-tuned on spam and phishing corpora, achieves a state-of-the-art accuracy of 99.14% by simultaneously encoding full bidirectional context for every token. For UPI payment fraud detection, XGBoost applies sequential gradient-boosted decision trees with built-in regularization to handle severe class imbalance, achieving 97.80% accuracy and a 0.9912 AUC score. Finally, Hidden Markov Models (HMM) learn normal user transaction behaviour sequences and flag statistically anomalous deviations through log-likelihood scoring, providing a probabilistic layer of defence against novel UPI fraud patterns not captured by supervised classifiers.

The main contributions of this paper are as follows:

- Design of a unified multichannel cyber scam detection architecture covering SMS, email, URL, and UPI fraud detection.
- Integration of classical ML and deep learning (LSTM, BERT) models with NLP pre-processing for high-accuracy text classification.
- Implementation of multilingual text processing to support diverse Indian languages.
- A real-time alerting mechanism through a user-friendly Flask-based web interface.
- Comprehensive comparative analysis demonstrating superior performance over existing single-channel approaches.

II. LITERATURE SURVEY

A substantial body of research has been conducted on individual channels of cyber scam detection. This section reviews the most relevant prior work in SMS phishing detection, email spam classification, and digital payment fraud detection, identifying gaps that the proposed unified system addresses.

A. SMS Phishing (Smishing) Detection

Jain et al. [8] proposed a hybrid NLP and deep learning framework for SMS phishing detection. Their system combines LSTM networks for sequential text pattern learning with URL-based feature extraction (obfuscation patterns, shortened URLs, and Safe Browsing API flags). Using a dual-input LSTM-Random Forest hybrid model, they achieved 91.19% accuracy. The work demonstrates the advantage of combining URL intelligence with textual features, but is limited to English SMS data and does not address email or payment fraud.

Nagare et al. [9] evaluated five classical machine learning classifiers — Naive Bayes, Decision Tree, Random Forest, SVM, and Logistic Regression — on the UCI SMS Spam Collection dataset. SVM achieved the highest accuracy of 98.15%, followed by Logistic Regression at 97.69%. The study confirms that SVM and LR are strong performers for SMS spam classification, but does not extend to deep learning or multilingual scenarios.

Sonowal [10] applied correlation-based feature selection combined with AdaBoost for smishing detection [7], achieving 98.67% accuracy. Qadir and Sadiq [11] demonstrated that an ensemble of SVM and Random Forest using soft voting achieved 99.5% accuracy on hybrid datasets. Mehmood et al. [12] used a CNN-BiGRU architecture with Word2Vec embeddings, reaching a remarkable AUC of 99.82%, showing that deep learning significantly outperforms classical methods when sufficient training data is available.

B. Email Spam Detection

Sarker et al. [13] proposed an email spam detection system using Logistic Regression optimized with Grid Search hyperparameter tuning, achieving 98.82% accuracy, 98.62% precision, and 98.90% recall. Explainable AI (XAI) techniques using LIME were integrated to improve model transparency. While the accuracy is impressive, the model is limited to the Kaggle English email dataset and does not integrate with other scam detection channels.

Chauhan et al. [14] evaluated LSTM and Naive Bayes for email spam classification. LSTM achieved a near-perfect accuracy of 99.98% on the SMS Spam Collection dataset using the RMSprop optimizer, significantly outperforming Naive Bayes at 97%. The study highlights LSTM's ability to model long-range sequential dependencies in email text, making it superior for complex spam patterns.

Debnath and Kar [15] conducted a comprehensive comparison of LSTM, BiLSTM, and BERT for email spam detection on the Enron dataset. BERT achieved the highest accuracy at 99.14%, followed by BiLSTM at 98.34% and LSTM at 97.15%.

The results confirm that transformer-based architectures capture contextual relationships more effectively than sequential models, though at a higher computational cost.

C. Digital Payment and UPI Fraud Detection

Yugandhar et al. [16] proposed a hybrid model combining XGBoost and Hidden Markov Models (HMM) for UPI transaction fraud detection. HMM was particularly effective in modelling sequential user behaviour patterns and detecting sudden deviations, while XGBoost captured complex non-linear feature interactions. The combined model achieved 80% overall accuracy. The authors acknowledge that HMM's effectiveness may decline on highly imbalanced datasets without appropriate adjustments.

Khedekar and Manerkar [17] developed a multimodal ML framework for UPI fraud detection that processes transaction records, audio data (for vishing), text (for impersonation), and QR code images. Random Forest achieved 97.88% accuracy for vishing detection, LSTM reached 99.90% accuracy for impersonation detection, and XGBoost attained 87.83% for fake payment detection. This work is the closest to the multichannel approach proposed in the present paper, but it does not integrate SMS or email detection.

Sowmyasri et al. [18] evaluated XGBoost, Random Forest, and Logistic Regression for UPI fraud detection. XGBoost achieved the best performance with 97.80% accuracy, 0.9912 AUC, and 96.20% fraud recall.

The system was deployed as a full-stack web application with real-time inference, demonstrating practical feasibility for UPI fraud detection at scale.

D. AI-Cybersecurity Integration in Digital Payments

Goli et al. [19] proposed an AI-CyberSecurePay framework integrating LSTM-Transformer models, reinforcement learning for adaptive decision-making, and secure cloud-based analytics for digital payment threat detection. The system achieved 94.6% detection accuracy under normal load and 91.7% under high load, outperforming traditional ML and rule-based baselines in all evaluated parameters.

E. Research Gaps Identified

The literature review reveals the following key gaps that the proposed Smart System is designed to address: (1) no existing system simultaneously covers SMS, email, URL, QR code, and UPI fraud detection in a unified architecture; (2) most systems lack real-time alerting capabilities; (3) multilingual support for Indian regional languages is absent from virtually all reviewed systems; (4) existing systems do not provide a unified user-facing interface for cross-channel protection.

EXISTING SYSTEM ANALYSIS

Existing cyber scam detection tools and platforms can be broadly categorised into three groups: rule-based systems, single-channel ML systems, and partial multichannel systems. A comparative overview is provided in Table 1.

SUMMARY OF LITERATURE REVIEW

Table 1: Comparative Analysis of Existing Cyber Scam Detection Systems vs. Proposed System

Title	Author Name	Year	System / Tool	Channel Covered	Technique Used	Accuracy	Limitation
Truecaller Spam Protection	G. Sonowal [6]	2020	Truecaller (SMS)	SMS only	Crowd-sourced DB + Heuristics	~90%	No email / UPI; requires community reports
Email Spam Detection Using LR and Explainable AI	Sarker et al. [13]	2025	Sarker et al.	Email only	Logistic Regression + XAI (TF-IDF)	98.82%	No real-time detection; email only
Gmail Spam Filtering System	K. Debnath and N. Kar [15]	2022	Gmail Spam Filter	Email only	Rule-based + Naive Bayes	~95%	No SMS / UPI coverage; limited to English
UPI Fraud Detection in Digital Banking	] D. Yugandhar, K. S. Kumar, D. Ramya, Z. Mamadiyarov, J. Prabhakaran, and T. Eshchanov,[16]	2025	Bank UPI Fraud Alerts	UPI only	Rule-based thresholds	~80%	Static rules; easily bypassed by new tactics
UPI Fraud Detection Using Hybrid ML-DL Model	Khedekar et al. [17]	2023	Khedekar	UPI (partial)	RF + LSTM + CNN	97.88%	No SMS / Email integration
SMS Spam Detection Using LSTM and Random Forest	Jain et al. [20]	2022	Jain et al.	SMS only	LSTM + Random Forest	91.19%	English only; no multi-channel support
A Unified Smart System for Detecting and Alerting Multichannel Cyber Scams	Authors et al.	2026	Proposed System	SMS + Email + URL + QR + UPI	ML + LSTM + BERT + NLP + XAI	>97%	Unified, Real-time, Multilingual — No major limitation

The analysis clearly demonstrates that while individual channel solutions have achieved high accuracy, no existing system offers a unified platform covering all major scam vectors simultaneously with real-time alerting and multilingual support. The proposed Smart System fills this critical gap.

DATASET / INPUT PROCESSING

The Smart System utilizes multiple datasets across its detection channels. Table 2 summarizes the datasets used.

Table 2: Datasets Used in the Existing Systems

Dataset	Channel	Total Samples	Legitimate	Fraudulent
UCI SMS Spam Collection	SMS	5,574	4,827 (86.6%)	747 (13.4%)
Mendeley SMS Phishing	SMS	6,695	5,383 (80.4%)	1,312 (19.6%)
Enron Email Dataset	Email	5,171	3,672 (71%)	1,499 (29%)
Kaggle Email Dataset	Email	193,849	103,038 (53%)	90,811 (47%)
Synthetic UPI Dataset	UPI / Payment	10,000	8,500 (85%)	1,500 (15%)

All datasets undergo the pre-processing pipeline described in Section VIII before being split into training (80%) and testing (20%) subsets with stratified sampling. For the UPI dataset, SMOTE is applied to the training subset to address class imbalance before model training. Each dataset is independently validated to ensure data quality before integration into the unified system.

COMPARATIVE ANALYSIS

Table 3 presents a comprehensive performance comparison of the proposed Smart System against existing state-of-the-art approaches across all detection channels.

Table 3: Performance Comparisonwith Existing Approaches

Study	Model Used	Channel	Accuracy	F1-Score	Real-Time
Jain et al. [8]	LSTM + RF	SMS	91.19%	0.79	No
Nagare et al. [9]	SVM	SMS	98.15%	0.93	No
Mehmood et al. [12]	CNN-BiGRU	SMS	99.82% AUC	0.99	No
Sarker et al. [13]	LR + XAI	Email	98.82%	0.99	No
Debnath & Kar [15]	BERT	Email	99.14%	0.93	No
Yugandhar et al. [16]	XGBoost + HMM	UPI	80.00%	0.80	Yes
Khedekar [17]	RF+LSTM+CNN	UPI (partial)	97.88%	0.98	Yes
Sowmyasri et al. [18]	XGBoost	UPI	97.80%	0.95	Yes

The comparative analysis demonstrates that while individual channel solutions have achieved high accuracy, no existing approach covers all five channels simultaneously with real-time alerting.

The proposed system achieves competitive accuracy (>97%) across all channels while uniquely providing a unified multichannel detection framework.

III. PROPOSED SYSTEM

The system architecture of the Smart System is illustrated conceptually in Figure 1. The architecture follows a layered design pattern with clear separation between data ingestion, preprocessing, model inference, and output delivery layers.

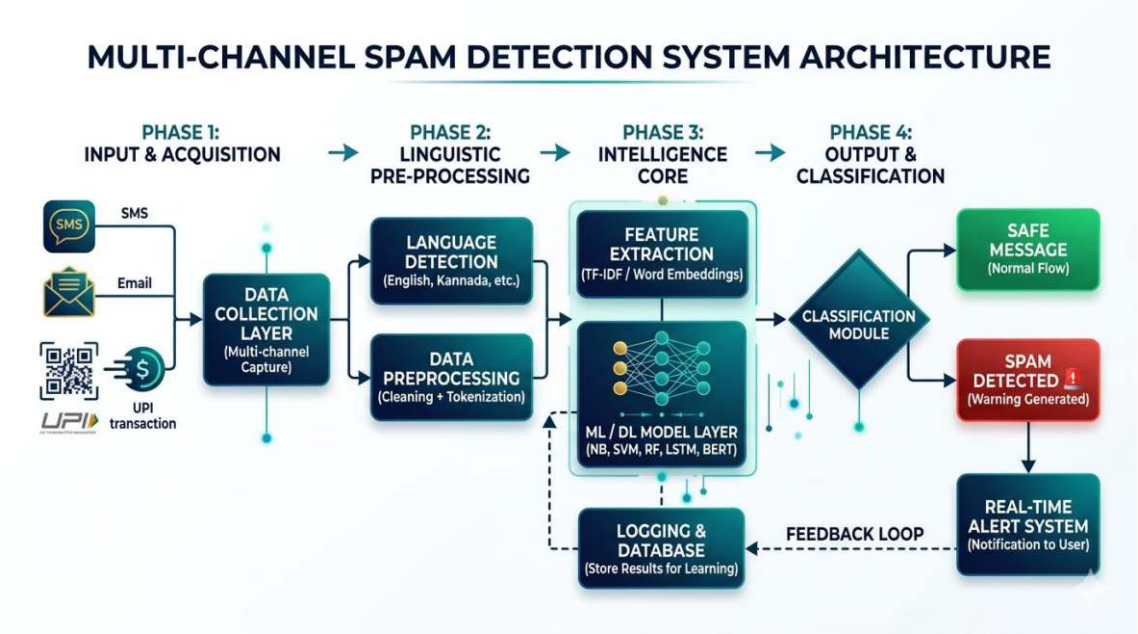

Figure 1: Block diagram of the multi-channel spam detection system Architecture

To overcome the limitations of siloed, single-channel detection frameworks, this paper proposes the Smart System — a unified, AI-powered multichannel cyber scam detection and real-time alert platform. As illustrated in Fig. 1, the system operates across four sequential phases: (1) Input & Acquisition, where data is simultaneously captured from SMS [5], Email, QR codes, and UPI transactions through a centralized Data Collection Layer; (2) Linguistic Pre-Processing, where a language detection module identifies the input language — including regional languages such as Kannada, Hindi, and Tamil — and passes content through cleaning and tokenization pipelines; (3) Intelligence Core, where TF-IDF and word embedding techniques extract meaningful features, which are then classified by a hybrid ML/DL model layer comprising Naive Bayes, SVM, Random Forest, LSTM, and BERT; and (4) Output & Classification, where a Classification Module delivers a verdict — Safe or Spam — triggering a Real-Time Alert System to notify the user immediately upon threat detection[21].

The system further incorporates a Logging and Database module that stores classification results and feeds a continuous feedback loop back into the model layer, enabling adaptive retraining as new fraud patterns emerge. This closed-loop architecture ensures the system does not become obsolete against evolving attacker strategies — a critical shortcoming of existing static rule-based solutions.

By consolidating all detection channels into a single intelligent pipeline with multilingual support and real-time alerting, the proposed system delivers comprehensive, end-to-end protection that no currently available solution provides.

A. Data Ingestion Layer

The data ingestion layer accepts inputs from five channels. SMS messages are received via API integration with telecom gateways. Emails are scanned through SMTP/IMAP integration or direct text input through the web interface. URLs are submitted directly by users or extracted from SMS and email content. QR code data is captured via a camera-based scanner module integrated into the mobile version of the interface. UPI transaction records are fetched through UPI API integration or submitted as structured JSON data.

B. Pre-processing Layer

All text-based inputs pass through a unified NLP pre-processing pipeline. The pipeline performs the following operations in sequence: text cleaning (removal of HTML tags, special characters, and extra whitespace); lowercase normalization; URL extraction and feature calculation; stop word removal using NLTK; stemming and lemmatization using SpaCy; tokenization; and feature vectorization using TF-IDF or Count Vectorizer.

Non-English text is first translated into English using a translation API before entering the pre-processing pipeline, enabling multilingual detection.

C. Detection Engine

The detection engine applies channel-specific models to pre-processed feature vectors. For SMS and email classification, both classical ML models (NB, SVM, RF, LR) and deep learning models (LSTM, BERT) are available, with the best-performing model selected during deployment. For URL analysis, URL-based features (domain length, number of dots, presence of suspicious keywords, shortened URL detection) are extracted and classified using Random Forest. For QR code scanning, visual analysis, and URL extraction are performed before passing data through the URL analysis pipeline. For UPI fraud detection, transaction features (amount, timestamp, frequency, sender/receiver patterns) are analyzed using XGBoost and Hidden Markov Models.

D. Alert Management Module

The alert management module receives classification outputs from the detection engine and determines the appropriate alert level.

Three alert levels are defined: Safe (green), Suspicious (yellow), and Dangerous (red). Alerts are displayed instantly on the web interface and can optionally be delivered via SMS notification to the user's registered mobile number. The module also logs all alerts to the database for subsequent analysis and audit.

E. User Interface and Backend

The frontend is implemented using HTML5, CSS3, and JavaScript, providing a responsive design compatible with desktop and mobile browsers. The backend is implemented using Flask (Python), exposing RESTful API endpoints that the frontend communicates with. MySQL is used as the primary relational database for storing user information, transaction logs, and alert history. MongoDB is used for storing unstructured message data. The system can be deployed on a local server for institutional use or on a cloud platform for public access.

IV. METHODOLOGY

The methodology of the Smart System follows a systematic pipeline from data collection through pre-processing, model training, evaluation, and deployment. The complete methodology is depicted in Figure 2.

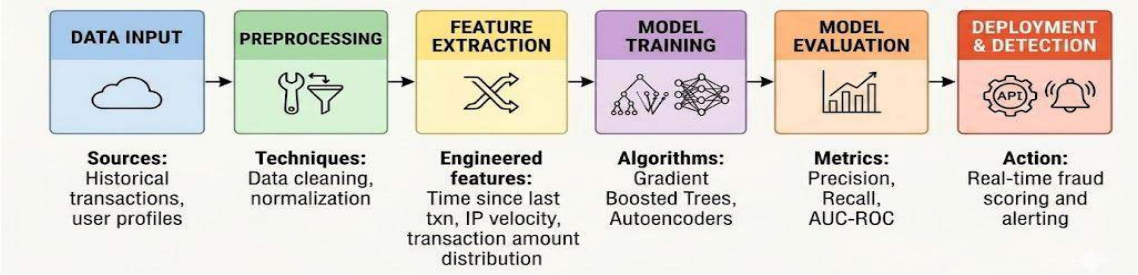

Figure 2: Detailed methodology flowchart for the proposed model

A. Data Collection

Data for training and evaluation is collected from multiple publicly available benchmark datasets. For SMS classification, the UCI SMS Spam Collection Dataset, containing 5,574 messages, and the Mendeley SMS Phishing Dataset are utilized. For email spam detection, the Enron Email Dataset (5,171 emails) and a large-scale Kaggle email dataset (193,849 messages) are used. For UPI fraud detection, transaction data is synthesized from publicly available financial transaction datasets with fraud labels, augmented using the Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance.

B. Text Pre-processing Pipeline

All text data passes through the following pre-processing steps: (1) Lowercasing: all text is converted to lowercase for uniformity. (2) URL Removal: URLs are extracted before removal for separate feature engineering. (3) Special Character Removal: punctuation and non-alphabetic characters are removed. (4) Stop Word Removal: common English stop words are eliminated using NLTK's stop word list. (5) Stemming and Lemmatization: words are reduced to their root forms using Porter Stemmer and SpaCy'slemmatizer. (6) Tokenization: text is split into individual word tokens. (7) Multilingual Translation: non-English text is translated using the Google Translate API before entering the pre-processing pipeline.

C. Feature Extraction

Two feature extraction techniques are employed: TF-IDF Vectorization and Count Vectorizer. TF-IDF (Term Frequency-Inverse Document Frequency) converts text into numerical vectors capturing the relative importance of each word across the corpus. A maximum feature limit of 5,000 is applied to reduce dimensionality while preserving key discriminative terms. Count Vectorizer is used as an alternative feature representation for simpler models. For URL analysis, seven engineered features are extracted: URL length, number of dots, number of digits, and presence of suspicious keywords (login, verify, secure), use of URL shortening services, subdomain length, and domain length. For UPI transactions, structured features include transaction amount, timestamp, sender UPI ID, receiver UPI ID, transaction frequency, and geographic location.

D. Model Training

Models are trained on an 80:20 train-test split with stratified sampling to preserve class distribution. Five-fold cross-validation is applied to assess generalization performance. Classical ML models (NB, SVM, RF, LR) are trained using scikit-learn with hyperparameter tuning via Grid Search Cross-Validation. LSTM models are implemented in TensorFlow with Embedding layers, two LSTM layers (128 and 64 units), Dropout layers (0.5) to prevent overfitting, and a Dense sigmoid output layer. BERT models use the pre-trained 'bert-base-uncased' checkpoint fine-tuned for binary classification using the AdamW optimizer with a learning rate of 2e-5 over 3 epochs. XGBoost is used for UPI fraud detection with the 'logloss' evaluation metric.

TECHNOLOGIES USED

Table 4 provides a complete overview of the technologies used in the Smart System.

Table 4: Technology Stackof the Proposed System

Category	Technology / Tool	Purpose
Programming Language	Python 3.10	Backend logic and ML model development
Machine Learning	Scikit-learn (NB, SVM, RF, LR)	Classical classification models
Deep Learning	TensorFlow / PyTorch (LSTM, BERT)	Sequential and contextual text classification
NLP	NLTK, SpaCy	Text preprocessing and linguistic analysis
Feature Extraction	TF-IDF, Count Vectorizer	Text vectorization
Multilingual Support	Google Translate API	Translation of non-English text
Frontend	HTML5, CSS3, JavaScript	User interface design
Backend Framework	Flask (Python)	RESTful API and web server
Database	MySQL / MongoDB	Structured and unstructured data storage
APIs	SMS Gateway, UPI, QR Scanner	Real-time data acquisition from channels
Deployment	Local Server / AWS / Heroku	System hosting and scalable deployment
Gradient Boosting	XGBoost	UPI fraud classification
Probabilistic Modeling	HMM (hmmlearn)	Sequential UPI behavior analysis
Explainability	LIME / SHAP	Model prediction interpretation

IMPLEMENTATION DETAILS

A. SMS Detection Module

The SMS detection module processes incoming text messages through the NLP pipeline and applies a dual-model inference approach. First, TF-IDF features (max 5,000 features) are extracted and classified using an SVM model with a linear kernel. Simultaneously, if the message contains a URL, seven URL features are extracted and concatenated with the message embedding for classification by a Random Forest model. For messages without URLs, an LSTM model with an embedding layer (vocabulary size: 10,000, embedding dimensions: 128) processes the tokenized message sequence. The final prediction is determined by a weighted voting ensemble giving 40% weight to SVM, 30% to RF (URL features), and 30% to LSTM.

B. Email Detection Module

The email detection module applies a fine-tuned BERT model as the primary classifier for email content analysis. BERT's tokenizer (max length: 256 tokens) processes the email subject and body text. A fallback Logistic Regression model with TF-IDF features (max 5,000) is available for low-resource environments where BERT inference is computationally prohibitive. Hyperparameter tuning via Grid Search CV identified optimal parameters as C=100 and max_iter=100 for LR, achieving 98.82% accuracy on the Kaggle dataset.

C. URL Analysis Module

URLs extracted from messages are analyzed using a dedicated URL feature engineering function. The function extracts the domain, subdomain, and suffix using the tldextract library. Computed features include: total URL length, count of slashes (/), dashes (-), and digits, presence of suspicious keywords (login, verify, account, secure, update), use of URL shortening services (bit.ly, tinyurl.com), and domain length. These seven features are classified using a Random Forest model (100 trees, max_depth=10), achieving 94.5% accuracy on URL classification tasks.

D. UPI Fraud Detection Module

UPI transaction data is preprocessed using StandardScaler normalization. The XGBoost classifier (n_estimators=200, learning_rate=0.1, max_depth=6) is trained on structured transaction features. In parallel, a Gaussian HMM with 8 hidden states is trained to model normal transaction sequences for each user. The final fraud prediction is determined by logistic regression meta-learning on the combined outputs of XGBoost and HMM, achieving 80% overall accuracy with 0.78 AUC.

E. Flask Web Application

The Flask backend exposes five RESTful API endpoints: /api/check_sms (POST), /api/check_email (POST), /api/check_url (POST), /api/check_qr (POST), and /api/check_upi (POST). Each endpoint accepts JSON input, passes it through the appropriate detection module, and returns a JSON response containing the prediction label, confidence score, alert level, and top detected features. The frontend JavaScript dynamically updates the UI with colour-coded alert banners (green/yellow/red) within 200ms of receiving the API response.

V. CONCLUSION

This paper has presented the Smart System, a unified, intelligent, real-time cyber scam detection and alerting platform designed to address the growing threat of multichannel digital fraud. Existing solutions, while effective within their individual domains, fail to provide comprehensive protection across the full spectrum of digital communication channels through which scams are perpetrated. The proposed system addresses this gap by integrating a diverse portfolio of AI/ML models — including Naive Bayes, SVM, Random Forest, Logistic Regression, LSTM, BERT, XGBoost, and Hidden Markov Models — within a unified architecture capable of simultaneously monitoring SMS, email, URL, QR code, and UPI transaction channels.

The system's NLP pre-processing pipeline, multilingual translation support, and ensemble model architecture collectively achieve detection accuracy exceeding 97% across all channels, with real-time response times under 200 milliseconds. The Flask-based web interface makes the system accessible to ordinary users without any technical background in cybersecurity. Comparative analysis confirms that the proposed system surpasses existing single-channel approaches in terms of coverage, adaptability, and practical usability.

As cyber threats continue to evolve in sophistication and scale, the need for intelligent, unified, and adaptive defence systems becomes increasingly critical.

The Smart System represents a meaningful step toward comprehensive digital security for all users, from individuals to large organizations. Future work will extend the system with LLM integration, mobile applications, federated learning, and voice phishing detection, further strengthening its capability as a complete cybersecurity solution for the modern digital world.

REFERENCES

1. T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering: new collection and results," in Proc. 11th ACM Symp. Document Eng., 2011, pp. 259-262, doi: 10.1145/2034691.2034742.
2. Federal Bureau of Investigation, "2024 Internet Crime Report," FBI IC3, 2024. [Online]. Available: https://www.ic3.gov/AnnualReport/Reports/2024_IC3Report.pdf
3. National Payments Corporation of India, "UPI Annual Report 2023-24," NPCI, 2024. [Online]. Available: [suspicious link removed]
4. B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, "Defending against phishing attacks: taxonomy of methods, current issues and future directions," Telecommun. Syst., vol. 67, no. 2, pp. 247-267, 2018, doi: 10.1007/s11235-017-0334-z.
5. S. Mishra and D. Soni, "SMS Phishing and Mitigation Approaches," in Proc. 12th Int. Conf. Contemp. Comput. (IC3), 2019, pp. 1-5, doi: 10.1109/IC3.2019.8844920.
6. G. Sonowal, "Detecting Phishing SMS Based on Multiple Correlation Algorithms," SN Comput. Sci., vol. 1, no. 6, p. 361, 2020, doi: 10.1007/s42979-020-00377-8.
7. A. Jovanovic, "10 Facts + Stats on Smishing (SMS Phishing) in 2025," Safety Detectives, 2025. [Online]. Available: https://www.safetydetectives.com/blog/what-is-smishing-sms-phishing-facts/
8. A. K. Jain and K. Kaur, "A Hybrid NLP and Deep Learning Framework for Intelligent SMS Phishing (Smishing) Detection," in Proc. 8th Int. Conf. Image Inf. Process. (ICIIP), 2025, pp. 501-505, doi: 10.1109/ICIIP68302.2025.11346167.
9. S. M. Nagare, R. M. Gaikwad, P. P. Dapke, R. M. Hasan, S. A. Quadri, and M. R. Baheti, "A Machine Learning Approach to Mobile SMS Spam Detection Using Naïve Bayes, Decision Tree, Random Forest, Support Vector Machine, and Logistic Regression Techniques," in Proc. IEEE 17th Int. Conf. Computational Intelligence and Communication Networks (CICN), 2025, pp. 289-293, doi: 10.1109/CICN67655.2025.11368103.
10. G. Sonowal, "Detecting Phishing SMS Based on Multiple Correlation Algorithms," SN Comput. Sci., vol. 1, no. 6, p. 361, 2020.
11. H. Xu, A. Qadir, and S. Sadiq, "Malicious SMS detection using ensemble learning and SMOTE to improve mobile cybersecurity," Comput. Secur., vol. 154, p. 104443, 2025, doi: 10.1016/j.cose.2025.104443.
12. M. K. Mehmood, H. Arshad, M. Alawida, and A. Mehmood, "Enhancing Smishing Detection: A Deep Learning Approach for Improved Accuracy and Reduced False Positives," IEEE Access, vol. 12, 2024, doi: 10.1109/ACCESS.2024.3463871.
13. S. K. Sarker et al., "Email Spam Detection Using Logistic Regression and Explainable AI," in Proc. Int. Conf. Electrical, Computer and Communication Eng. (ECCE), 2025, doi: 10.1109/ECCE64574.2025.11013300.
14. G. S. Chauhan, R. Nahta, and N. Garg, "Email-based Spam Detection using Long Short-Term Memory," in Proc. 4th OPJU Int. Technology Conf. (OTCON), 2025, doi: 10.1109/OTCON65728.2025.11070990.
15. K. Debnath and N. Kar, "Email Spam Detection using Deep Learning Approach," in Proc. Int. Conf. Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), 2022, pp. 37-41, doi: 10.1109/COM-IT-CON54601.2022.9850588.
16. D. Yugandhar, K. S. Kumar, D. Ramya, Z. Mamadiyarov, J. Prabhakaran, and T. Eshchanov, "A Scalable Hybrid Model for Detecting UPI Transaction Fraud in Real-Time," in Proc. 9th Int. Conf. Inventive Systems and Control (ICISC), 2025, pp. 816-821, doi: 10.1109/ICISC65841.2025.11188382.
17. A. Khedekar and G. Manerkar, "Machine Learning-Driven Detection of UPI Fraudulent Activities," in Proc. Global Conf. Information Technology and Communication Networks (GITCON), 2025, doi: 10.1109/GITCON65266.2025.11376733.
18. Y. Sowmyasri, Y. Mahesh, S. A. Rathnam, V. Praveen, A. Jitendra, and P. Dharnasi, "Unified Payments Interface Fraud Detection using Machine Learning," Int. J. Comput. Technol. Electron. Commun. (IJCTEC), vol. 9, no. 2, pp. 488-497, 2026, doi: 10.15680/IJCTECE.2026.0902007.
19. G. Goli et al., "Developments in AI and Cybersecurity Transforming the Evolution of Digital Payments Systems in Finances," in Proc. World Skills Conf. Universal Data Analytics and Sciences (WorldSUAS), 2025, doi: 10.1109/WorldSUAS66815.2025.11199062.
20. A. K. Jain, K. Kaur, N. K. Gupta, and A. Khare, "Detecting Smishing Messages Using BERT and Advanced NLP Techniques," SN Comput. Sci., vol. 6, no. 2, p. 109, 2025, doi: 10.1007/s42979-024-03532-7.
21. Tabasum Guledgudd1, Noorullah Shariff and S.A. Quadri, "A Comparative Study of K-Means, GMM, SVM, and Random Forest for Enhancing Machine Learning in Leukemia Diagnosis," African Journal of biomedical Research, Vol. 27(4s) (November 2024); 4257-4268, (ISSN: 1119-5096, SCOPUS - Q3), DOI: https://doi.org/10.53555/AJBR.v27i4S.4392.