INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING & APPLIED SCIENCES
ISSN: 2322-0821(O)
ISSN: 2394-9910(P)
VOLUME 13 ISSUE 4
Oct 2025 - Dec 2025
www.irjeas.org

Review Article

The Evolution of SMS Phishing (Smishing) Detection: A Comprehensive Review of Heuristic, Machine Learning and Natural Language Processing Techniques

Yogita Rajput[1]*, Kalpana Mishra[2]

[1] Research Scholar, Dept. of Computer Science Engineering, JNCT, Bhopal, India
yogitarajput65645@gmail.com
[2] Asst. Professor, Dept. of Computer Science Engineering, JNCT, Bhopal, India
kalpana.cse@jnctbhopal.ac.in

**Corresponding Author*: yogitarajput65645@gmail.com

DOI–10.55083/irjeas.2025.v13i04005

***Abstract*:** Smartphones are a primary target for cybercriminals, with "smishing" (SMS phishing) being one of the most pervasive threats. The immediacy and implicit trust of SMS messaging make these attacks highly effective, leading to significant financial and personal damage. This necessitates the development of robust, adaptive defenses. This paper provides a critical review of the evolution of smishing detection techniques. We begin by examining foundational static methods, such as signature-based and rule-based filters, and detail their inherent limitations against dynamic threats. The core of the review then analyzes the machine learning (ML) paradigm, breaking down the complete pipeline from Natural Language Processing (NLP) for text normalization to feature engineering and model classification.

However, this review argues that the field has entered a new, more challenging era. The foundational ML models that defined the 2010s are now being outpaced by a 2023-2025 surge in sophisticated, AI-generated campaigns and large-scale

Phishing-as-a-Service (PhaaS) operations. We analyze this modern threat landscape, including the defenses deployed by commercial leaders like Google and Truecaller, which leverage on-device large language models (LLMs). Critically, we introduce an analysis of non-technical constraints, detailing how data privacy and legal mandates, such as the GDPR and CCPA, are a central design driver pushing the industry toward privacy-preserving, on-device architectures. This analysis indicates that the future of smishing detection lies not in a single "best" classifier, but in a multi-layered, on-device framework that can counter AI-generated content while remaining compliant with global privacy laws.

Key words: Smishing, SMS Phishing, Machine Learning, Natural Language Processing (NLP), Text Normalization, Mobile Security, Threat Detection.

I. INTRODUCTION

The smartphone has evolved from a communication tool into a central hub for personal and financial life [1]. This transition, while convenient, has created a concentrated and valuable target for cybercrime [13]. With billions of global smartphone users sending trillions of SMS messages annually [11], the trusted, private nature of the SMS inbox has been weaponized.

Phishing, a form of social engineering where attackers impersonate legitimate entities to steal credentials or data [2, 32], is dubbed "smishing" when conducted via SMS [10]. The threat's potency lies in its immediacy; SMS messages have a reported open rate as high as 98%, with most read within minutes [35]. This bypasses the cautious skepticism users might apply to email, leading to direct financial losses, identity theft, and an erosion of trust in mobile communications [10, 12]. Attackers continuously evolve, using tactics like ransomware to amplify their impact [36].

A primary challenge in smishing defense is the nature of SMS language. It is informal, concise, and filled with non-standard slang, abbreviations, and misspellings (e.g., "ur acc" for "your account"). Attackers intentionally leverage this "linguistic noise" to bypass simple keyword filters. This has historically created a challenge for defenders, requiring intelligent systems that can understand the underlying malicious *intent* rather than just matching superficial patterns [4].

Today, this challenge is amplified. The same "noisy" and evasive language can now be generated at scale by Large Language Models (LLMs), enabling sophisticated, grammatically perfect, and highly personalized attacks. This paper provides a critical review of the detection methods developed to combat this evolving threat. We survey the evolution from static filters to foundational machine learning models. Crucially, we then analyze the modern 2023-2025 landscape, which is defined by a new arms race: commercial, on-device AI defenses against AI-driven Phishing-as-a-Service (PhaaS) operations.

To address common gaps in prior surveys, this paper explicitly introduces:

1. **Visual Frameworks:** A detection pipeline diagram (Fig. 1) and a

taxonomy of methods (Fig. 2) to visually structure the field.

2. **Performance Context:** A comparative table of foundational classifiers (Table 1) to assess performance.
3. **Real-World Analysis:** A discussion of commercial-grade defenses (e.g., Google, Truecaller) and the non-technical, legal constraints (e.g., GDPR) that shape modern system design.

II. FOUNDATIONAL DETECTION METHODS

The first line of defense against smishing was adapted from email spam filtering. These foundational methods were largely static and relied on manually defined patterns to identify malicious content.

A. Systems Based on Signatures

Signature-based detection is the most basic approach. A "signature"—such as a sender's phone number, a known malicious URL, or a hash of the message content—is stored in a database of known threats. When a message arrives, its features are compared to this database. A match results in the message being flagged.

The advantage of this method is speed and a very low false-positive rate for known threats. However, its primary flaw is its complete inability to detect new, or "zero-day," attacks [32]. An attacker can trivially bypass this system by slightly altering the message text, using a new URL shortener, or sending from a new number.

B. Heuristic and Rule-Based Frameworks

Heuristic, or rule-based, systems use a set of manually crafted rules to assign a "smishing score." These rules are based on patterns observed by human experts. For example, a system might use logic such as: IF (message contains 'bank' OR 'account') AND (message contains a URL) THEN (increase smishing_score).

Jain and Gupta [19] proposed a rule-based system with nine content-based rules. Such systems are highly transparent, as the reason for a flag is easily auditable. However, like signature systems, they are brittle. Attackers quickly learn the rules and design messages to evade them (e.g., using "b@nk" or "Your account has an urgent update, click here: [url]"). Their static nature makes them unsuitable for the dynamic strategies used by modern attackers. The clear limitations of these static methods necessitated a paradigm shift toward automated, learning-based systems.

III. THE MACHINE LEARNING PIPELINE FOR DETECTION

To overcome the brittle nature of static rules, researchers turned to machine learning (ML). An ML model can automatically learn the distinguishing characteristics of smishing from a labeled dataset. The process of applying ML to unstructured SMS text is a multi-stage pipeline, visualized in Figure 1.

Figure 1: A conceptual diagram of the machine learning pipeline for smishing detection. The quality of the final classification is highly dependent on the preprocessing and feature engineering stages.

A. NLP Preprocessing & Normalization

Before classification, raw SMS text must be cleaned and structured. This is arguably the most critical stage due.

- **The "Noisy" Text Problem:** SMS language is characterized by slang ("lol"), abbreviations ("idk"), intentional misspellings ("gr8"), and relaxed grammar. This "noise" creates feature space fragmentation; a model sees "ur," "your," and "ure" as three distinct features, diluting the statistical signal that "your" (as in "your account") is linked to smishing.
- **Preprocessing Steps:** Common steps include:
    1. **Tokenization:** Splitting the message into individual words (tokens).
    2. **Lowercasing:** Converting all tokens to lowercase.
    3. **Stop Word Removal:** Removing common but low-meaning words ("a," "the," "is").
    4. **Stemming/Lemmatization:** Reducing words to their root form ("claiming" -> "claim").
- **Text Normalization:** For SMS, a crucial, non-optional step is **text normalization**—the process of converting non-standard tokens back to their canonical English form (e.g., "u" -> "you"). Almeida et al. [23] demonstrated that text normalization significantly improves classifier performance. This is often achieved using manually curated dictionaries of slang [25]. The importance of this step was validated in their work, which serves as one of the few ablation studies in the literature, proving that classifier performance drops without this step [23].

B. Feature Engineering

Once the text is clean, it is converted into a numerical vector representation (features) that a model can understand.

- **Bag-of-Words (BoW) and TF-IDF:** The most common approach is BoW, where each message is represented by a vector showing the count of each word in the vocabulary. A refinement, **Term Frequency-Inverse Document Frequency (TF-IDF)**, gives more weight to words that are common in *one* message but rare across *all* messages, making them more discriminative.
- **Content-based Semantic Features:** To capture intent, Karami and Zhou [20] used features from the LIWC lexicon, which categorizes words into psychological groups (e.g., "financial terms," "anxiety").
- **Metadata and URL-based Features:** Features can be extracted from sources beyond the text. Mishra and Soni [21] and Joo et al. [3] focused on analyzing the properties of embedded URLs (e.g., use of shortening services, number of subdomains) as powerful indicators of malicious intent.

C. Classification Models

With features extracted, a classifier model is trained to distinguish "ham" (legitimate) from "smishing."

- **Probabilistic Classifiers (Naive Bayes):** Naive Bayes (NB) is one of the most common baseline models due to its speed, simplicity, and strong performance [3, 14, 23]. It uses Bayes' Theorem to calculate the probability a message is smishing given the features (words) it contains. Its efficiency makes it ideal for resource-constrained environments, such as on-device applications.
- **Discriminative Classifiers (SVM):** A Support Vector Machine (SVM) works by finding the optimal hyperplane that best separates the data points of the two classes (ham vs. smishing) in a high-dimensional feature space. Yadav et al. [14] used SVMs in their SMSAssassin framework, demonstrating its high efficacy.
- **Deep Learning Models:** While classic ML models are effective, they often fail to capture word order and context. Deep learning models, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, were adopted to understand the *sequence* of words, theoretically leading to a more nuanced understanding of intent [39, 5].

To provide a clear comparison of these foundational academic models, Table 1 synthesizes the performance reported in several key studies.

Table 1: Performance Comparison of Foundational Smishing Classification Models

Study	Classifier(s) Used	Dataset	Key Metrics Reported	Key Finding
Almeida et al. [23]	Naive Bayes, SVM	SMS Spam Collection v.1	96.6% Accuracy (SVM)	**Text normalization is critical.** Performance dropped significantly without it.
Yadav et al. [14]	Naive Bayes, SVM	Self-Collected (India)	98.7% Recall (SVM)	Crowdsourced-based filtering (SMSAssassin) was highly effective.

Jain & Gupta [19]	RIPPER (Rule-Based)	Self-Collected	98.6% Accuracy	A carefully crafted rule-set could outperform early ML in zero-hour attacks.
Mishra & Soni [21]	Naive Bayes, J48	SMS Spam Collection + PhishTank URLs	98.1% Accuracy	**Hybrid features are superior.** Combining text content + URL behavior analysis yielded the best results.

As reported by the original authors. Metrics are not directly comparable across studies due to different datasets.

IV. A TAXONOMY OF DETECTION PARADIGMS

To synthesize the methods discussed, Figure 2 presents a taxonomy that organizes the evolution of smishing detection. This taxonomy plots the progression from simple, static techniques to the complex, AI-driven systems that define the modern landscape.

Figure 2: A taxonomy classifying smishing detection methods from foundational static analysis to modern, dynamic AI-driven paradigms.

V. THE MODERN (2023-2025) THREAT LANDSCAPE

The methods discussed in Section III, while foundational, are increasingly being tested by a new wave of attacks that began surging in 2023. This modern landscape is defined by the industrialization of phishing through AI and Phishing-as-a-Service (PhaaS).

A. The Rise of AI-Generated Smishing & PhaaS

The "linguistic noise" that was once a hurdle for defenders is now being weaponized *by* attackers. The wide availability of generative AI has led to a documented **1,265% surge in AI-driven phishing attacks** since 2023 [42]. As of March 2025, AI agents have been shown to **outperform elite human red teams** in creating successful phishing campaigns [43].

This threat is amplified by **Phishing-as-a-Service (PhaaS)** platforms, which sell pre-packaged phishing kits and infrastructure. A prominent 2024-2025 example is "**Smishing Triad,**" a PhaaS group linked to over **194,000 malicious domains** [45]. This group, primarily impersonating toll services and package delivery, exemplifies the new industrial scale of smishing.

The key tactic of these groups is **rapid domain churn**. Analysis shows 71.3% of their malicious domains are active for less than one week [46]. This tactic renders traditional signature-based (Section II.A) and URL-reputation features

(Section III.B) almost completely ineffective, as a domain is taken down before it can be blacklisted. This threat is confirmed by Microsoft's 2025 Digital Defense Report, which identifies nation-state actors and criminal groups alike using generative AI to scale social engineering and evade controls [47, 48].

B. Commercial & On-Device Defenses

In response to this AI-driven threat, the most advanced defenses are no longer academic models but are being deployed by commercial, platform-level providers. These systems address the real-world challenge of detecting AI-generated content at scale.

- **Google (Android):** Google Messages has integrated a sophisticated spam and smishing filter that runs entirely on-device. In 2024, Google confirmed this filter is now powered by its **Gemini Nano** LLM. This on-device model can predict scamming sites and messages, even for "zero-day" threats, without the user's message content ever leaving their phone [49].
- **Truecaller:** In March 2024, Truecaller rolled out its "Max" protection, an AI-based feature for detecting new spam numbers. Furthermore, its **AI Call Scanner** can detect AI-synthesized voices [50], a direct counter to the 442% surge in voice phishing (vishing) [44], a threat that text-only models cannot address.

This industry shift to on-device AI is not just a performance choice; it is a critical response to the practical and legal challenges of real-world deployment.

VI. DEPLOYMENT, PRIVACY, AND OPEN CHALLENGES

A model's "accuracy" in a lab is only one part of its value. Real-world deployment involves challenges of data, user trust, and legal compliance.

A. Datasets and Performance Metrics

A persistent barrier in smishing research is the lack of large, public, and modern datasets. Much of the foundational research used the "SMS Spam Collection v.1" [23], which is now over a decade old and does not reflect AI-generated threats.

Furthermore, simple "accuracy" is a poor metric. In a dataset where 99% of messages are "ham," a model that flags nothing is 99% accurate. For security, we must use a confusion matrix to balance:

- **True Positive Rate (Recall):** The percentage of smishing that is correctly caught.
- **False Positive Rate (FPR):** The percentage of legitimate "ham" messages incorrectly flagged as smishing. A high FPR destroys user trust and renders the app unusable.

B. Deployment Models: On-Device vs. Cloud

There are two primary architectures for deploying a detection model:

1. **Cloud-Based:** The SMS content is sent to a remote server for analysis. This allows for massive, complex models but introduces network latency and severe privacy concerns [17].

2. **On-Device:** The entire model runs on the user's smartphone. This is preferred for privacy and real-time detection [6]. However, it requires models to be extremely efficient to avoid draining the CPU, memory, and battery.

C. Critical Challenge: Privacy and Legal Compliance (GDPR/CCPA)

The choice between cloud and on-device deployment is not purely technical. Scanning the content of a user's private SMS messages is a profound privacy intrusion.

Cloud-based models that transmit and analyze personal messages on a server fall under the jurisdiction of strict data privacy laws, most notably the **EU's General Data Protection Regulation (GDPR)** and the **California Consumer Privacy Act (CCPA)** [52].

- **Legal Risk:** Non-compliance carries severe penalties, with GDPR fines reaching up to **4% of a company's global annual revenue** [53].
- **Consent Mandate:** To operate legally, a cloud-based service must obtain explicit, granular, and unambiguous "opt-in" consent from the user *before* any data is scanned [53].

This significant legal and financial risk is the primary business driver for the industry's shift to **on-device** models. By processing data directly on the user's phone, as Google does with Gemini Nano [49], the data never leaves the device. This "data minimization" approach is inherently more compliant with global privacy laws, as it avoids the collection and processing of sensitive personal communications.

D. Open Challenges and Future Research Directions

The field is far from solved. The shift to AI-driven attacks opens several new, specific avenues for research:

1. **Countering Phishing-as-a-Service (PhaaS):** Foundational models rely heavily on URL and sender reputation. With PhaaS groups like Smishing Triad using 194,000+ domains with a one-week churn rate [45, 46], this is no longer viable. Future research must focus on *content-intrinsic* features that are independent of a rapidly changing URL.
2. **Detection of AI-Generated Content:** The new arms race is detecting AI-generated text and voice. This requires models that can identify the subtle statistical signatures of LLMs and voice synthesis, moving beyond simple keywords to analyze text consistency, style, and semantic coherence.
3. **Resource-Efficient On-Device Models:** To be viable, the powerful LLMs needed to detect AI-generated text must run on a phone. This requires significant research into model quantization, pruning, and knowledge distillation to create models (like Gemini Nano [49]) that are both powerful and resource-light.
4. **Multimodal Threat Detection:** Smishing is a text-based vector, but it is often paired with vishing (voice) and quishing (QR code) attacks. Future systems must fuse signals from multiple modalities—analyzing text, URLs, sender behavior, and even synthesized

voice [50]—into a single, unified risk score.

VII. CONCLUSION

This review has charted the evolution of smishing detection, from brittle static rules to the foundational machine learning models that dominated the field for a decade. Our analysis of the foundational ML pipeline confirms that a robust NLP preprocessing stage, particularly text normalization, is a critical prerequisite for effective classification [23].

However, the central finding of this review is that this foundational paradigm, while still relevant, has been rendered insufficient by the modern threat landscape. The fight against smishing is no longer an academic exercise in classifier optimization but an active, industrial-scale arms race. The modern attacker is not a lone actor but an AI-powered **Phishing-as-a-Service (PhaaS)** operation, like "Smishing Triad" [45], that leverages generative AI and rapid domain churn to achieve unprecedented scale and evasiveness [46].

In response, the *de facto* defense, led by commercial giants like Google and Truecaller [49, 50], has shifted to **on-device artificial intelligence**. This review provides the critical context that this shift is driven not only by a quest for performance but fundamentally by the non-negotiable legal and privacy mandates of **GDPR and CCPA** [53]. The massive liability of processing personal SMS messages in the cloud has made privacy-preserving, on-device models a design necessity.

The future of smishing defense, therefore, lies not in finding a slightly more accurate cloud-based classifier, but in solving the complex challenge of developing resource-efficient, on-device models that can detect AI-generated text and voice in real-time, all while protecting user privacy.

REFERENCES:

[1] S. F. Verkijika, "Understanding smartphone security behaviours: An extension of the protection motivation theory with anticipated regret," *Computers & Security*, 2018.

[2] B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, "Fighting against phishing attacks: state of the art and future challenges," *Neural Computing and Applications*, vol. 28, no. 12, pp. 3629–3654, 2017.

[3] J. W. Joo, S. Y. Moon, S. Singh, and J. H. Park, "S-Detector: an enhanced security model for detecting Smishing attacks for mobile computing," *Telecommunication Systems*, pp. 1–10, 2017.

[4] S. Mishra and D. Soni, "A content-based approach for detecting smishing in a mobile environment," in *Proc. Int. Conf. Sustainable Computing in Science, Technology and Management (SUSCOM)*, Amity University Rajasthan, Jaipur, India, 2019.

[5] J. W. Seo, J. S. Lee, H. Kim, J. Lee, S. Han, J. Cho, and C. H. Lee, "On-Device Smishing Classifier Resistant to Text Evasion Attack," *IEEE Access*, 2024.

[6] O. N. Akande, O. Ghenle, C. C. Abikoye, R. G. Jimoh, H. B. Akande, A. O. Balogun, and A. Fatokun, "SMSPROTECT: An automatic smishing detection mobile application," *ICT Express*, vol. 9, no. 2, pp. 168–176, 2023.

[7] D. Goel and A. K. Jain, "Mobile phishing attacks and defence mechanisms: state of the art and open research challenges," *Computers & Security*, 2017.

[8] M. Abdulstatar, H. Ahmad, D. Goel, and F. Ullah, "Towards Deep Learning Enabled Cybersecurity Risk Assessment for Microservice Architectures," *arXiv preprint arXiv:2403.15105*, 2024.

[9] R. K. Jayalah, H. Ahmad, D. Goel, M. S. Syed, and F. Ullah, "Microservice Vulnerability Analysis: A Literature Review with Empirical Insights," *IEEE Access*, 2024.

[10] Infosec Institute, "Smishing," 2018.

[11] "Number of smartphone users worldwide from 2014 to 2020," *Statista*, 2018.

[12] "State of the Phish," *Wombat Security*, 2013.

[13] G. Deepak and B. S. Pradeep, "Challenging issues and limitations of mobile computing," *Int. J. Computer Technology & Applications*, vol. 3, no. 1, pp. 177–181, 2012.

[14] K. Yadav, P. Kumaraguru, A. Goyal, A. Gupta, and V. Naik, "SMSAssassin: Crowdsourcing-driven mobile-based system for SMS spam filtering," in *Proc. 12th Workshop on Mobile Computing Systems and Applications*, pp. 1–6, ACM, 2011.

[15] E. S. M. Alfy and A. A. AlHasan, "Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm," *Future Gener. Comput. Syst.*, vol. 64, pp. 98–107, 2016.

[16] Hauri Inc., "Smishing Defender," 2017.

[17] A. Lee, K. Kim, H. Lee, and M. Jun, "A Study on Realtime Detecting Smishing on Cloud Computing Environments," in *Advanced Multimedia and Ubiquitous Engineering*, pp. 495–501, Springer, Berlin, Heidelberg, 2016.

[18] H. Ahmad, C. Treude, M. Wagner, and C. Szabo, "Towards Resource-Efficient Reactive and Proactive Auto-Scaling for Microservice Architectures," *SSRN*, 2024.

[19] A. K. Jain and B. B. Gupta, "Rule-Based Framework for Detection of Smishing Messages in Mobile Environment," *Procedia Comput. Sci.*, vol. 125, pp. 617–623, 2018.

[20] A. Karami and L. Zhou, "Improving static SMS spam detection by using new content-based features," in *Proc. IEEE 15th Int. Conf. Information Reuse and Integration (IRI)*, Redwood City, CA, USA, 2014.

[21] S. Mishra and D. Soni, "Smishing Detector: A security model to detect smishing through SMS content analysis and URL behaviour analysis," *Future Gener. Comput. Syst.*, vol. 108, pp. 803–815, 2020.

[22] R. M. Silva, T. A. Almeida, and A. Yamakami, "MDLText: An efficient and lightweight text classifier," *Knowledge-Based Syst.*, vol. 118, pp. 152–164, 2017.

[23] T. A. Almeida, T. P. Silva, I. Santos, and J. M. G. Hidalgo, "Text normalization and semantic indexing to enhance Instant Messaging and SMS spam filtering," *Knowledge-Based Syst.*, vol. 108, pp. 25–32, 2016.

[24] H. Kaur and E. J. S. Mann, "Text Normalization using Statistical Machine Approach," *Unpublished manuscript*, 2016.

[25] "NoSlang. Internet Slang Dictionary & Translator," *NoSlang*, 2017.

[26] "Smishing message images," *Pinterest*, 2017.

[27] G. Somani, M. S. Gaur, D. Sanghi, and M. Conti, "DDoS attacks in cloud computing: collateral damage to non-targets," *Computer Networks*, vol. 109, pp. 157–171, 2016.

[28] H. Ahmad, C. Treude, M. Wagner, and C. Szabo, "Smart HPA: A Resource-Efficient Horizontal Pod Auto-scaler for Microservice Architectures," *arXiv preprint arXiv:2403.07909*, 2024.

[29] S. Chopra, H. Ahmad, D. Goel, and C. Szabo, "ChainWD: Advancing Cybersecurity Vulnerability Assessment with Large Language Models," *arXiv preprint arXiv:2412.04756*, 2024.

[30] H. Ahmad, I. Dharmadasa, F. Ullah, and M. A. Babar, "A review on c3i systems' security: Vulnerabilities, attacks, and countermeasures," *ACM Comput. Surv.*, vol. 55, no. 9, pp. 1–38, 2023.

[31] F. Mouton, L. Leenen, and H. S. Venter, "Social engineering attack examples, templates and scenarios," *Computers & Security*, vol. 59, pp. 186–209, 2016.

[32] A. K. Jain and B. B. Gupta, "Phishing Detection: Analysis of Visual Similarity Based Approaches," *Security and Communication Networks*, 2017.

[33] G. S. Awumez, J. O. Agyemang, S. S. Boakye, and D. Bennpong, "SmisIsShield: A Machine Learning-Based Smishing Detection System," in *Int. Conf. Wireless Intelligent and Distributed Environment for Communication*, pp. 205–221, Cham: Springer Nature Switzerland, 2023.

[34] R. Kohilan, H. E. Warakagoda, T. T. Kittigoda, N. Skandhakumar, and N. Kuruwitatarachchi, "A Machine Learning-based Approach for Detecting Smishing Attacks at End-user Level," in *2023 IEEE Int. Conf. e-Business Engineering (ICEBE)*, pp. 149–154, IEEE, 2023.

[35] Sender, "SMS Open Rates: Everything You Need to Know," 2024.

[36] R. Brewer, "Ransomware attacks: detection, prevention and cure," *Network Security*, vol. 2016, no. 9, pp. 5–9, 2016.

[37] A. K. Jain, A. Panday, and D. Goel, "S-Defender: A Smishing Detection Approach," in *Cyber Warfare, Security and Space Computing: 2nd Int. Conf., SpacSec 2024*, Jaipur, India, Feb. 22–23, 2024, p. 68, Springer Nature, 2024.

[38] D. Goel and A. K. Jain, "Smishing-classifier: A novel framework for detection of smishing attack in mobile environment," in *Smart and Innovative Trends in Next Generation Computing Technologies: 3rd Int. Conf., NGCT 2017*, Dehradun, India, Oct. 30–31, 2017, pp. 502–512, Springer, 2018.

[39] A. K. Jain, D. Goel, S. Agarwal, Y. Singh, and G. Bajaj, "Predicting spam messages using back propagation neural network," *Wireless Personal Commun.*, vol. 110, pp. 403–422, 2020.

[40] D. Goel and A. K. Jain, "Overview of smartphone security: Attack and defense techniques," in *Computer and Cyber Security*, pp. 249–279, Auerbach Publications, 2018.

[41] D. Goel, "Enhancing network resilience through machine learning-powered graph combinatorial optimization: Applications in cyber defense and information diffusion," *arXiv preprint arXiv:2310.10667*, 2023.

[42] Brightside AI Blog, "AI-Generated Phishing vs Human Attacks: 2025 Risk Analysis," Oct 24, 2025.

[43] Hoxhunt, "AI-Powered Phishing Outperforms Elite Red Teams in 2025," Mar 2025.

[44] DeepStrike, "Vishing Statistics 2025: AI Deepfakes Drive $40B in Losses," Oct 20, 2025.

[45] Thailand Computer Emergency Response Team (ThaiCERT), "“Smishing Triad” – Chinese PhaaS Group Linked to Over 194,000 Malicious Domains in Global Smishing Campaign," Oct 27, 2025.

[46] R. Lakshmanan, "Smishing Triad Linked to 194,000 Malicious Domains in Global Phishing Operation," *The Hacker News*, Oct 24, 2025.

[47] Industrial Cyber, "Microsoft 2025 digital defense report flags rising AI-driven threats," Oct 21, 2025.

[48] Help Net Security, "What Microsoft's 2025 report reveals about the new rules of engagement in cyberdefense," Oct 24, 2025.

[49] Google Safety Center, "Protection from Online Scams & Fraud," retrieved Oct 2025.

[50] Truecaller, "Truecaller: Spam Call Blocker - Apps on Google Play," retrieved Nov 2025.
[51] Deccan Herald, "Truecaller gets new AI feature to block spam calls," Mar 20, 2024.
[52] Microsoft Learn, "Consent management overview - Dynamics 365 Customer Insights," retrieved Oct 2025.
[53] Annotations Micro Systems, "How to Comply with the GDPR and Other Regulations for Email and SMS Marketing," *Medium*, Feb 26, 2024.

Conflict of Interest Statement: *The authors declare that there is no conflict of interest regarding the publication of this paper.*

Generative AI Statement: *The author confirms that no Generative AI tools were used in the preparation or writing of this article.*

Publishers Note: *All statements made in this article are the sole responsibility of the author(s) and do not necessarily reflect the views of their affiliated institutions, the publisher, editors, or reviewers. Any products mentioned or claims made by manufacturers are not guaranteed or endorsed by the publisher.*

Cite this Article

Yogita Rajput, Kalpana Mishra. The Evolution of SMS Phishing (Smishing) Detection: A Comprehensive Review of Heuristic, Machine Learning and Natural Language Processing Techniques. International Research Journal of Engineering & Applied Sciences (IRJEAS). 13(4), pp. 48-59, 2025. https://doi.org/10.55083/irjeas.2025.v13i04005