Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.078940

REVIEW

Survey of AI-Based Threat Detection for Illicit Web Ecosystems: Models, Modalities, and Emerging Trends

Jaeho Hwang[1] **and Moohong Min**[2,*]

[1]Department of Immersive Media Engineering, Sungkyunkwan University, Seoul, Republic of Korea
[2]Department of Computer Education/Social Innovation Convergence Program, Sungkyunkwan University, Seoul, Republic of Korea

*Corresponding Author: Moohong Min. Email: iceo@skku.edu

Received: 11 January 2026; Accepted: 02 March 2026; Published: 30 March 2026

ABSTRACT: Illicit web ecosystems, encompassing phishing, illegal online gambling, scam platforms, and malicious advertising, have rapidly expanded in scale and complexity, creating severe social, financial, and cybersecurity risks. Traditional rule-based and blacklist-driven detection approaches struggle to cope with polymorphic, multilingual, and adversarially manipulated threats, resulting in increasing demand for Artificial Intelligence (AI)-based solutions. This review provides a comprehensive synthesis of research on AI-driven threat detection for illicit web environments. It surveys detection models across multiple modalities, including text-based analysis of Uniform Resource Locator (URL) and HyperText Markup Language (HTML), vision-based recognition of webpage layouts and logos, graph-based modeling of domain and infrastructure relationships, and sequence modeling using transformer architectures. In addition, the paper examines system architectures, data collection and labeling pipelines, real-time detection frameworks, and widely used benchmark datasets, while also discussing their inherent limitations related to imbalance, representativeness, and reproducibility. The review highlights critical challenges such as evasion strategies, cross-lingual detection barriers, deployment latency, and explainability gaps. Furthermore, it identifies emerging research directions, including the use of Generative Adversarial Network (GAN) for threat simulation, few-shot and self-supervised learning for data-scarce environments, Explainable Artificial Intelligence (XAI) for transparency, and predictive AI for proactive threat forecasting. By integrating technical, legal, and societal perspectives, this survey offers a structured foundation for researchers and practitioners to design resilient, adaptive, and trustworthy AI-based defense systems against illicit web threats.

KEYWORDS: Artificial intelligence (AI); cybersecurity; threat detection; machine learning (ML); deep learning (DL); phishing; illegal online gambling; illicit websites; resilient detection; explainable artificial intelligence (XAI)

1 Introduction

In recent years, the quantitative and qualitative growth of web-based cybercrimes in cyberspace has become increasingly evident. Among them, phishing and illicit online gambling websites have emerged as representative digital threats that cause substantial social harm. In 2023, more than five million phishing attacks were reported worldwide, and in the first quarter of 2024 alone, 960,000 incidents were detected, underscoring the severity of the threat [1]. According to statistics from the Federal Bureau of Investigation's Internet Crime Complaint Center (FBI IC3), approximately 200,000 phishing-related crimes are reported annually in the United States, while globally an estimated 3.4 billion phishing emails are transmitted every day [2,3].

The emergence of Large Language Model (LLM), such as ChatGPT, has triggered a surge in AI-driven, automated phishing attacks, with increasingly diverse techniques. Reports by cybersecurity companies Hoxhunt and KnowBe4 indicate that phishing attempts have risen by approximately 4151% since the release of ChatGPT in 2022 [4]. Polymorphic phishing campaigns, which modify content to evade detection, are also proliferating rapidly [5].

During the early phase of the COVID-19 pandemic, adversaries exploited widespread social instability to launch large-scale phishing campaigns, which in turn stimulated research on phishing detection [6]. Early foundational studies proposed taxonomies of phishing attacks by categorizing attack motivations, techniques, and defense mechanisms. Since then, numerous studies have advanced Machine Learning (ML)-based phishing detection, demonstrating the effectiveness of integrating URL-based, visual similarity-based, and behavioral approaches [7,8].

The online gambling industry has also expanded rapidly since the pandemic, leading to a parallel increase in illicit activities and cybersecurity threats. Min and Lee analyzed more than 11,000 illegal gambling websites and demonstrated that ML models integrating landing page features, domain information, WHOIS records, and images achieved high detection performance [9]. Musa et al., in their study of the Croatian lottery system, highlighted that vulnerabilities in online gambling platforms are distributed across networks, web applications, and user interactions [10].

Ultimately, phishing and illicit gambling threats are evolving within highly organized web ecosystems, and the limitations of traditional rule-based detection methods are becoming increasingly evident. As a result, there is a growing demand for the systematic adoption of AI-based detection techniques, coupled with a macro-level analysis of the structural characteristics and domain interconnections of the web ecosystem.

Fig. 1 is an overview of traditional and AI-based phishing detection. Most conventional detection systems have relied on blacklists, signatures, or static rule-based methods. While these methods are relatively simple to design and operate, they suffer from fundamental shortcomings in real-world environments. Chief among these is the inability to respond effectively to rapidly mutating threats. For example, when attackers slightly alter and re-register phishing domains or subtly modify content, static methods alone cannot identify them [11–13]. Moreover, blacklist-based approaches are hindered by detection delays and higher false positive/false negative rates. Their reliance on manual updates not only slows response times but also weakens adaptability to evolving attack scenarios [14]. Similarly, heuristic methods that rely on predefined keywords or patterns are easily disrupted by adversarial obfuscation strategies, and their performance deteriorates significantly under shifts in user behavior [8].

To address these limitations, ML and Deep Learning (DL)-based detection methods have been actively explored. ML models are capable of generalizing classification rules from diverse input features, enabling them to adapt flexibly to changing attack patterns [15]. DL methods, such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM) networks, can process multiple modalities—including URLs, HTML structures, images, and screenshots—achieving higher accuracy and lower error rates compared to rule-based approaches [16,17].

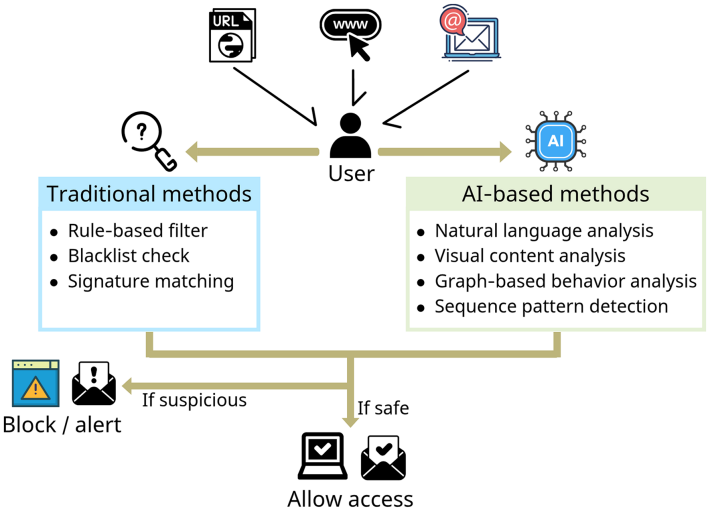

Figure 1: Overview of traditional vs. AI-based phishing detection.

More recently, LLM-based detection techniques have garnered significant attention. By leveraging advanced contextual understanding, LLMs go beyond keyword matching to evaluate the semantic coherence of sentences. While phishing sentences generated by ChatGPT, Gemini, or Claude frequently bypass traditional ML detectors, LLM-based models have shown superior resilience [18]. In fact, rephrased phishing content generated by LLMs often degrades the accuracy of conventional detectors, reinforcing the importance of integrating LLM-based approaches as a complement to existing systems [19].

In addition, Reinforcement Learning (RL)-based adaptive detection systems have proven valuable in adversarial environments where evasion and detection strategies co-evolve. RL models learn reward/penalty structures in security contexts and can design proactive detection strategies that evolve autonomously [20].

This paper does not limit itself to analyzing individual attack techniques. Rather, it offers an integrated survey of illicit web ecosystems while comprehensively reviewing AI-based detection techniques and recent research trends. By extending beyond the scope of prior studies, which often focused narrowly on specific attacks or datasets, this work provides a multi-layered perspective encompassing models, modalities, datasets, system architectures, legal responses, and future directions.

As outlined in Fig. 2, the remainder of this paper is organized as follows: Section 2 presents the review methodology. Section 3 categorizes illicit web ecosystems, presenting their technical and social characteristics along with legal responses. Section 4 reviews AI-based detection methods by modality, including text, vision, graph, and sequence models, and summarizes recent findings. Section 5 introduces system architectures and detection pipelines, covering data collection, labeling, real-time processing, and response mechanisms. Section 6 surveys public benchmarks and datasets, examining issues of data quality, representativeness, and reproducibility. Section 7 discusses current limitations and open challenges, including evasion strategies, multilingual issues, and practical deployment barriers. Section 8 outlines future research directions, such as GAN-based threat simulation, few-shot and self-supervised learning, XAI, and predictive AI. Finally, Section 9 concludes the paper by summarizing the overall discussion.

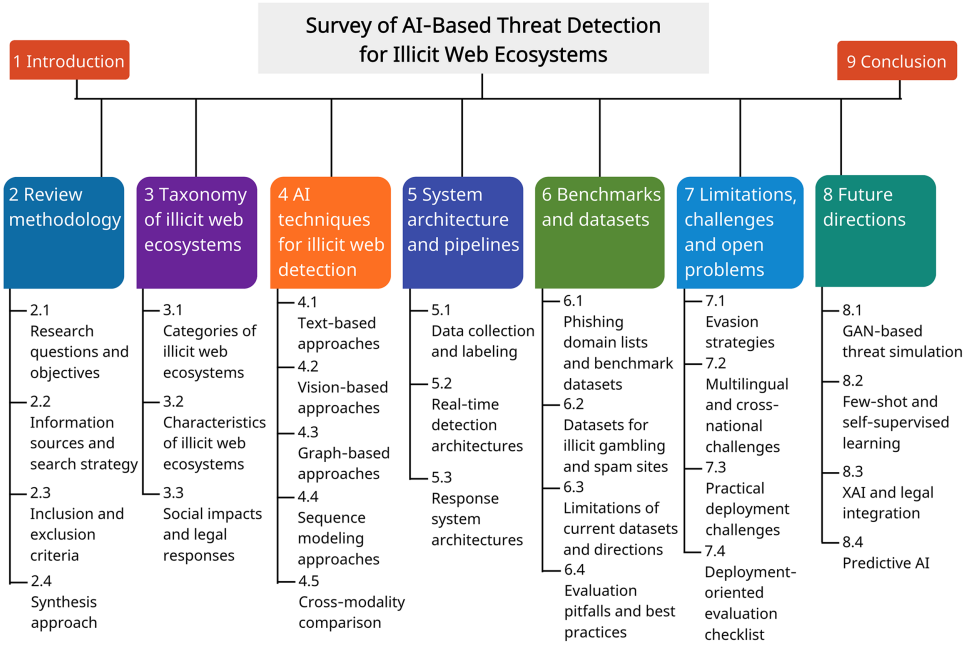

Figure 2: Overview of the content structure.

Through this structured approach, the paper aims to provide both macro-level and micro-level perspectives on AI-driven threat detection for illicit web ecosystems, offering researchers and practitioners a comprehensive reference for future studies and practical implementations.

2 Review Methodology

This survey was conducted using a structured but iterative literature exploration process. Rather than following a strictly pre-registered systematic review protocol, the study employed an evolving search-and-refinement strategy to capture the rapidly developing research landscape of AI-based detection for illicit web ecosystems.

2.1 Research Questions and Objectives

To systematically synthesize the rapidly evolving body of research on AI-driven threat detection for illicit web ecosystems, this survey is guided by the following Research Questions (RQ):

- **RQ1:** How can AI-based detection techniques be systematically categorized for illicit web ecosystems, and what distinct threat assumptions does each modality address?
- **RQ2:** What factors in dataset construction and evaluation protocols critically influence reported detection performance and generalization to unseen or zero-day threats?
- **RQ3:** What architectural patterns and system-level designs enable scalable, real-time, and deployment-aware detection in practical environments?
- **RQ4:** What fundamental limitations constrain the robustness and adaptability of current AI-based detection systems in evolving illicit web ecosystems?

By addressing these questions, this survey aims not only to organize existing approaches across modalities and system designs, but also to critically examine their empirical validity, deployment feasibility, and resilience against zero-day and adversarial evolution.

In this survey, we use the term *zero-day threats* to refer to attack instances or infrastructures that have not been previously observed or labeled in the training data. The term *unseen data* denotes evaluation samples drawn from distributions not represented in the training split, while *novel attacks* may involve new Tactics, Techniques, or Procedures (TTPs) that differ structurally from known patterns. Although these terms are sometimes used interchangeably in prior work, we distinguish them conceptually for clarity.

2.2 Information Sources and Search Strategy

Literature was primarily retrieved from IEEE Xplore, ACM Digital Library, ScienceDirect, Scopus, Web of Science, and Google Scholar. Initial searches were performed using combinations of ecosystem-related and AI-related keywords (e.g., illicit, illegal, malicious, taxonomy, website detection, threat detection, machine learning, deep learning, AI).

During the review process, backward and forward snowballing techniques were also employed. Additional keyword searches (e.g., phishing, gambling, scam, malvertising, text, vision, graph, sequence, transformer, ensemble, dataset) were repeated to expand coverage. Among the search results, the review was conducted with a priority on relatively high-cited or recently published.

2.3 Inclusion and Exclusion Criteria

Studies were included if they met the following criteria:

- Proposed or evaluated AI-based detection mechanisms targeting illicit web content, infrastructure, or distribution channels.
- Provided empirical evaluation using publicly available or proprietary datasets.
- Clearly described model architecture or feature engineering techniques.
- Based on peer-reviewed journals or conference publications. Exceptionally, a limited number of preprint studies (e.g., large language model technical reports) were included due to their relevance to emerging trends.

Studies were excluded if they:

- Focused solely on traditional signature-based detection without AI components.
- Lacked experimental validation.
- Were non-English publications (excluding reports to verify local statistical data).
- Were duplicate reports of the same methodology.

2.4 Synthesis Approach

A narrative synthesis approach was employed to identify cross-paper patterns, trade-offs, and recurring evaluation pitfalls. Given the interdisciplinary and fast-evolving nature of the topic, the review emphasizes conceptual synthesis and modal analysis rather than strict quantitative aggregation of performance metrics.

Particular attention was given to overlapping domains such as phishing, gambling, and scam. When studies addressed multiple illicit domains, they were classified based on their primary threat model and evaluation focus.

3 Taxonomy of Illicit Web Ecosystems

3.1 Major Categories of Illicit Web Ecosystems

The structure of the web-based cybercrime ecosystem has become increasingly complex, with blurred boundaries among different types of criminal activities. Accordingly, to effectively detect and counter illicit

websites, it is essential to establish a clear taxonomy of categories and to systematically understand the characteristics and threat mechanisms of each. This section summarizes the representative types of illicit web ecosystems—namely phishing, illegal gambling, malicious advertising, and scam—focusing on their technical and socio-structural classification criteria.

3.1.1 Phishing Ecosystems

Phishing is a representative social-engineering attack designed to steal sensitive user information or to install malware. Chiew et al. analyzed phishing attacks by categorizing their types, vectors, and technical implementations, thereby offering a systematic understanding of phishing ecosystems [13]. Fraşzczak and Fraşzczak systematized phishing detection techniques according to the types of detection features, such as URL-based, content-based, and visual similarity-based approaches [21].

Apea and Yin categorized phishing detection into list-based, HTML content similarity-based, and hybrid approaches, and comparatively analyzed their applicability and technical limitations [22]. From a practical perspective, phishing types are also distinguished by attack channels and media, including spear-phishing, Short Message Service (SMS) phishing (smishing), voice-based phishing (vishing), and pharming (phishing without a lure).

3.1.2 Illegal Gambling Ecosystems

Illegal gambling websites not only cause direct financial loss to users but also involve broader social issues such as money laundering, links to organized crime, and personal data breaches. Banks and Waugh classified gambling-related crimes into categories such as organized-crime related, financial-crime related, and user addiction-inducing, while elaborating on the proliferation structure through online platforms [23]. Gainsbury et al. proposed a game-content-based taxonomy of gambling, classifying gambling games according to platform, payment method, and the level of skill required [24].

In addition, the boundary between legal and illegal online gambling platforms is often ambiguous, with the same website potentially being legal in one jurisdiction while illegal in another. Therefore, a comprehensive taxonomy must consider not only technical analyses of platforms but also operational entities, payment flows, and distributed server infrastructures [23,24].

3.1.3 Malvertising and Scam Ecosystems

Scam sites and malicious advertising (malvertising) resemble phishing in that they lure users but primarily aim at fraudulent payments, click-through revenue, or malware distribution. Verma et al. proposed a domain-independent deception taxonomy that encompasses various fraudulent activities, including phishing, scams, fake news, and job fraud. Their framework introduces a multi-dimensional classification of deception, covering agents, stratagems, goals, exposure mechanisms, and communication modalities, thereby extending analysis beyond phishing alone [25].

Phillips and Wilder conducted clustering-based classification of advance-fee scams and cryptocurrency scam sites, revealing that such sites are often linked at the infrastructure level to phishing or illegal gambling sites [26]. Consequently, the taxonomy of malvertising and scam sites should adopt a multidimensional approach that considers not only content but also link structures, hosting environments, and backend domain connections.

3.2 Key Characteristics of Illicit Web Ecosystems

The illicit website ecosystem goes beyond a mere list of URLs; it comprises a multi-layered infrastructure that encompasses the dynamics of domain creation, alteration, and expiration, Internet Protocol (IP) distribution strategies, and the utilization of global hosting infrastructures. This section describes the principal features of web-based cyber threats in three aspects: domain lifetimes, IP distribution structures, and the concentration of multinational operations and hosting providers.

3.2.1 Short Domain Lifetimes

The average lifetime of malicious domains is generally very short, and in many cases they are abandoned or re-registered before being detected. Lee et al. analyzed 286,237 phishing URLs collected at 30-min intervals and reported an average lifetime of approximately 54 h, with a median lifetime of 5.46 h. Notably, Google Safe Browsing's average detection time was 4.5 days, while 83.9% of phishing domains had already expired before detection [27].

Research by Foremski and Vixie and Affinito et al. further revealed that 9.3% of newly registered domains were deleted within seven days, most of them as part of automated phishing campaigns. Their reported median lifetime was only four hours and 16 min [28,29]. An analysis by the DNS Research Federation indicated that about 63% of newly registered domains were blocked within four days of registration, with an average lifetime of about 21 days [30].

3.2.2 IP Distribution and Fast-Flux Structures

Malicious domains often employ the fast-flux technique to evade detection by rapidly rotating IP addresses for a single domain name. Fast-flux typically involves associating multiple IPs with a single domain and configuring a short Time-to-Live (TTL) so that each Domain Name System (DNS) query returns a different IP. Each IP corresponds to a pool of compromised zombie hosts (proxy bots), which relay requests to the actual control server (mothership, or Command and Control (C2) server), thereby concealing its location. As illustrated in Fig. 3, this process enables rapid IP replacement through short TTLs, while directing traffic through proxy bots to the hidden C2 server. Such structures are widely used to support load balancing, peer-to-peer C2 server operations, and domain evasion strategies, thereby undermining IP-based blocking policies [31,32].

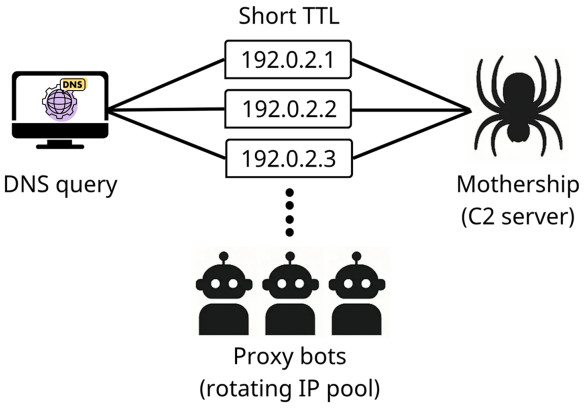

Figure 3: Fast-flux IP rotation.

This IP distribution strategy complicates detection at the DNS level, as it combines rapid IP changes, geographic dispersion, short TTLs, and short-term registrations identifiable via WHOIS records, all of which favor detection evasion [33,34].

3.2.3 TLDs and Multinational Hosting Infrastructures

According to Interisle Consulting Group's 2024 annual report, approximately 79% of domains used in phishing were maliciously registered, and many became operational within three days of registration. A substantial share of phishing domains were registered through the top five generic Top-Level Domain (TLD) registrars, and 42% of all phishing domains reported during the study period were concentrated in the top ten new generic TLDs. Moreover, half of all phishing attacks originated from IP address ranges belonging to the top ten hosting providers. Among them, Cloudflare alone accounted for 23% of cases (437,108 incidents), underscoring the severe concentration of registration and hosting infrastructures [35].

At the same time, infrastructures supporting malicious domains and phishing emails are becoming increasingly geographically distributed. For example, numerous malicious websites have been observed embedding U.S. city names into their domain names to target local users [36]. Furthermore, about 40% of phishing emails were routed through more than three countries before reaching recipients, with many originating from Eastern Europe, Central America, Middle East, and Africa [37]. Such routing practices hinder traceability and illustrate how attackers leverage globally dispersed infrastructures and multinational email routing paths to avoid detection, while tailoring attacks with geographic and linguistic features to target specific user groups.

3.3 Social Impacts and Legal Responses

Illegal web–driven cybercrimes extend beyond mere technical threats, leading to tangible societal harms and imposing significant burdens on legal and policy responses. This section analyzes the social damages and legal response frameworks with a focus on phishing, illegal online gambling, and scam/malicious advertising platforms.

3.3.1 Social Impacts and Legal Responses to Phishing

Phishing attacks cause far-reaching consequences, not only individual damages but also corporate financial losses, brand reputation degradation, and data breaches. According to a UK government survey, 79% of companies reported experiencing phishing within the past 12 months, with 56% responding that phishing was the most disruptive type of attack [38]. A Proofpoint report similarly revealed that 91% of companies experienced at least one email-based phishing attempt, and 26% suffered direct financial losses [39]. A 2025 Keevee research review indicated that social engineering–based cyberattacks account for 98% of all cyber incidents, of which phishing comprises 70%. On average, companies incur USD 17.7 million in annual financial losses from such attacks, with approximately 60% suffering reputational damage and an average of 200 days required for incident recovery [40].

Legal response systems have been progressively institutionalized. In the United States, the Anti-Phishing Act of 2005 was introduced [41]. In the United Kingdom, the Fraud Act 2006 established a general offense of fraud, thereby closing loopholes within preexisting statutory and common law frameworks [42]. National cybercrime legislations across multiple jurisdictions are evolving not only to impose criminal penalties on phishing perpetrators but also to incorporate victim protection, corporate accountability, and compensation frameworks [43–45].

3.3.2 Social Impacts and Legal Responses to Illegal Online Gambling

The determination of illegality in online gambling varies across nations and cultural contexts. In Asian cultural spheres, where social sensitivity toward gambling is particularly high, illegality criteria are applied more strictly. For example, research on illegal online gambling in South Korea indicates that, as shown in Fig. 4, the scale of the illegal market has grown annually and has already surpassed the legal gambling market [46]. According to a 2024 meta-analysis by Tran et al., gambling has become a globally prevalent activity, with approximately 46% of adults worldwide having engaged in gambling. Alongside the expansion of the online gambling sector, the prevalence of problematic gambling has increased. An estimated 8.7% of adults engage in risky gambling, and 1.41% meet the criteria for problematic gambling, which is closely associated with household financial deterioration, mental health decline, and the breakdown of interpersonal relationships [47]. Illegal online gambling is particularly harmful due to its structural characteristics, such as the availability of high-stakes betting, absence of age or time restrictions, and lack of minimal safeguards. Moreover, fraudulent operations, such as manipulated winning odds or misappropriation of funds, further exacerbate the risk of users falling into problematic gambling, amplifying socioeconomic harms [48].

Figure 4: Market growth trends of legal vs. illegal online gambling in South Korea [46].

Representative legal responses include the U.S. Unlawful Internet Gambling Enforcement Act of 2006 (UIGEA), which restricts payment flows related to illegal internet gambling. The act prohibits the acceptance of funds from unlawful gambling through credit cards, electronic transfers, or checks, and obligates payment networks and financial intermediaries to block such transactions [49]. Similarly, countries such as South Korea, Japan, Philippines, Singapore, and Indonesia apply national cybercrime statutes and gambling regulations to restrict online gambling through a combination of platform blocking, Internet Service Provider (ISP) controls, and criminal prosecutions [48,50].

3.3.3 Social Impacts and Legal Responses to Scam Platforms

Scam platforms combine phishing, malicious advertising, and ransomware, thereby causing multilayered damages: stealing personal information, encrypting systems for ransom, or harvesting identity data for secondary crimes [51–53]. According to Tanti's 2023 study, Business Email Compromise (BEC) in the United States alone resulted in USD 2.9 billion in damages, with average losses per company ranging from USD

4.65 million to USD 5.01 million. Among surveyed companies, 60% reported experiencing data loss, 52% reported credential or account compromise, and 47% had encountered ransomware infections; 18% reported direct financial losses [54].

A notable case is LabHost, a global scam platform originating in Canada in 2021, which positioned itself as a "one-stop-shop" for fraud. LabHost is estimated to have caused more than 94,000 victims and USD 28 million in damages in Australia alone. Over 10,000 cybercriminals are reported to have used the platform to replicate legitimate websites of banks, government agencies, and major organizations, thereby producing large-scale victimization. The Australian Federal Police (AFP) continues to pursue the related infrastructure under the international investigation Operation Nebulae, highlighting the highly organized nature of the scam ecosystem [55].

Consequently, international cooperation has become both essential and increasingly active. A prime example is the takedown of the Avalanche infrastructure, one of the largest botnet dismantling operations in history. This four-year collaboration involved law enforcement agencies from 180 countries, resulting in the shutdown of approximately 800,000 malicious domains, the arrest of key operators, and the successful dismantling of the infrastructure [56].

4 AI Techniques for Illicit Site Detection

4.1 Text-Based Approaches

One of the most widely adopted approaches in illicit website detection is the analysis of text-based features, including URL strings, HTML content, email bodies, and headers. As shown in Table 1, this section provides a structured overview of detection techniques centered on text modalities, and compares representative model architectures.

Table 1: Text-based AI techniques for illicit web threat detection.

Ref.	Approach	Input Features	Model/Architecture	Key Results/Contributions
Asif et al. [57]	URL-based ML/DL	URL lexical, domain, content features	ML/DL models	Surveyed 26 studies; summarized reported performance of ML/DL methods; highlighted trade-offs between efficiency and accuracy
Bhat et al. [58]	URL-based ML/DL	24 handcrafted features	RF, SVM, DT vs. CNN, LSTM	Reported CNN/LSTM up to 98.7% F1-score; ML models around 92.8% F1; showed DL superiority for complex URL patterns

(Continued)

Table 1 (continued)

Ref.	Approach	Input Features	Model/Architecture	Key Results/Contributions
Opara et al. [59]	HTML+URL -based DL	Raw HTML + URL embeddings	CNN end-to-end	Achieved ≈ 98.1% accuracy; reduced manual feature engineering; robust across HTML/URL inputs
Opara et al. [60]	HTML-based DL	HTML text embeddings	CNN	Reported ≈ 93% accuracy with ~2% FPR; demonstrated language-independent phishing detection
Tian et al. [61]	Multimodal	URL, HTML, JavaScript, visual features	Transformer, GNN, LLM-based architectures	Proposed a modality-based taxonomy; analyzed use of Transformers/GNNs/LLMs; emphasized multimodal integration
Asiri et al. [62]	Hybrid DL	Combined URL, HTML, DOM	CNN, LSTM, hybrid (Web2Vec, OFS-NN)	Reviewed hybrid DL approaches; noted challenges in real-time deployment; highlighted importance of preprocessing and feature representation

4.1.1 Text-Based AI Techniques

Although a URL is short in length, it inherently contains various syntactic and structural characteristics—such as regular expression patterns, length, domain form, and the presence of special characters—which make it highly useful for malicious URL detection. Asif et al. conducted a comprehensive review of recent ML–based phishing detection algorithms focusing on structural URL features (e.g., token counts, number of subdomains, and suspicious TLDs), emphasizing the balance between accuracy and efficiency [57]. Bhat et al. extracted 24 structural URL features such as length, number of special characters, and the presence of IP addresses, and reported that traditional classifiers (Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT)) achieved an average F1-score of approximately 92.8%. In comparison, DL models such as CNN and LSTM achieved up to 98.7% F1-score, demonstrating superior detection performance on complex URL patterns [58].

At the HTML code level, multiple features can also be leveraged to detect phishing. Opara et al. designed a CNN-based DL model (WebPhish) that takes entire HTML documents as raw input, achieving a high accuracy of about 98.1%. This end-to-end model integrates both URL and HTML embeddings during training [59]. The same researchers also proposed HTMLPhish, which embeds HTML content at both character and word levels and uses a CNN for classification. This model incorporates all elements of HTML—including tags, attributes, links, and text—highlighting its ability to perform language-independent detection. In experiments with over 50,000 multilingual HTML documents, HTMLPhish achieved approximately 93% accuracy on test datasets [60]. The model demonstrated strong generalization to new HTML patterns without manual feature engineering, while maintaining a low False Positive Rate (FPR) (~2%), thereby showing practical applicability.

More recently, multimodal detection models that integrate textual and non-textual inputs—such as URL strings, HTML structures, JavaScript execution, and webpage visual features—have gained attention. Tian et al. categorized these inputs into key modalities, systematically reviewed state-of-the-art detection techniques, and analyzed the architectures, advantages, and limitations of each. They also discussed the applicability of advanced DL techniques—including Transformers, Graph Neural Network (GNN), and LLMs—to malicious URL detection, and suggested directions for future research [61]. Notably, their study shifted the taxonomy from an algorithm-centric perspective to a modality-centric perspective, thereby offering a multidimensional view of multimodal information utilization in URL security research.

Asiri et al. provided a detailed analysis of hybrid DL–based phishing detection models (e.g., Web2Vec, Optimal Feature Selection and Neural Network (OFS-NN)). Among them, the Web2Vec model, illustrated in Fig. 5, integrates multiple inputs such as URLs, HTML content, and Document Object Model (DOM) structures, and employs CNN, Bidirectional LSTM (BiLSTM), and attention mechanisms to extract multi-layered features before performing final classification. While these models achieved high accuracy, the complexity of processing heterogeneous inputs introduces latency in real-time scenarios. As such, the trade-off between detection performance and processing time was emphasized as a major issue. Accordingly, they proposed model design strategies that combine lightweight CNN architectures with sequence-learning LSTM components to ensure both real-time performance and detection accuracy [62].

4.1.2 Synthesis and Implications of Text-Based Approaches

Text-based detection approaches—leveraging URL lexical features, HTML content, email headers, and textual embeddings—remain the most computationally efficient and widely deployable modality for illicit web threat detection. Their strengths lie in scalability, low-latency processing, and ease of integration into browser extensions, email gateways, and Application Programming Interface (API)-based filtering systems.

However, under zero-day or unseen attack conditions, purely supervised text-based models exhibit notable limitations. Because many models rely on statistical regularities within known lexical patterns, adversarial modifications—such as homoglyph substitutions, token obfuscation, multilingual drift, or LLM-generated rephrasing—can significantly degrade performance. While transformer-based and LLM-driven models demonstrate improved contextual generalization, their robustness remains dependent on training data diversity and evaluation protocols.

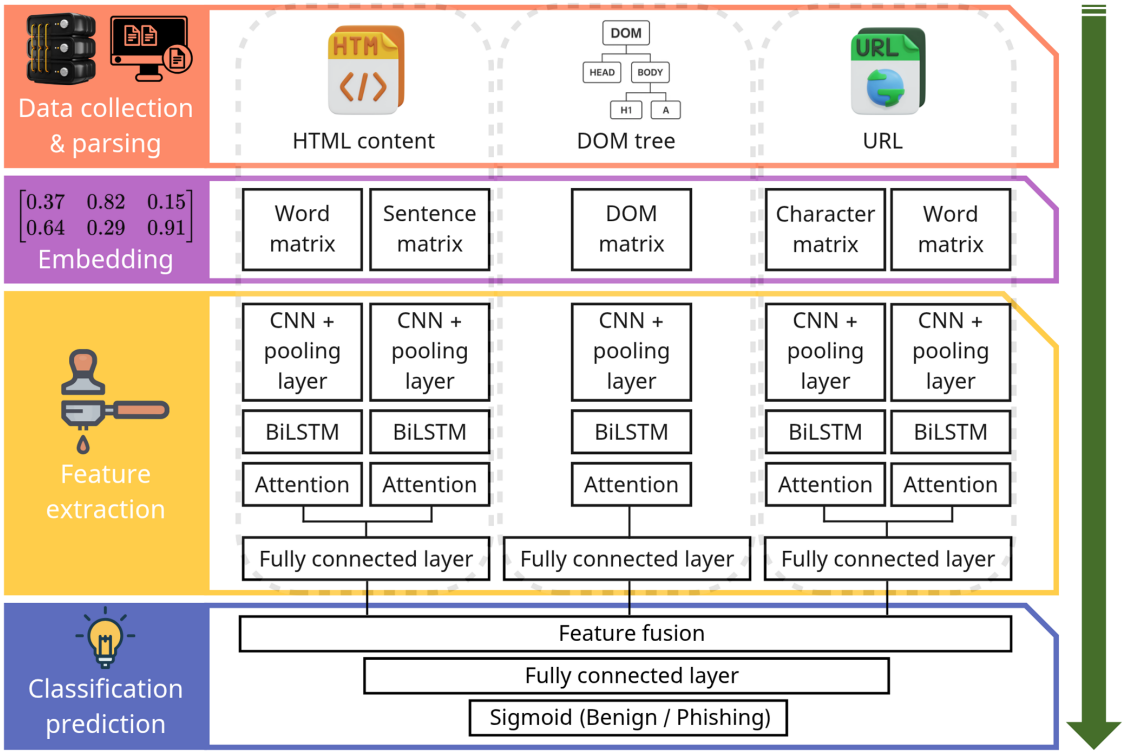

Figure 5: Architecture of the Web2Vec hybrid phishing detection model [62].

From a deployment perspective, text-based approaches offer low computational overhead and minimal crawling requirements, making them suitable for real-time detection scenarios. Nevertheless, their vulnerability to semantic obfuscation and evolving content manipulation suggests that text-only detection may be insufficient for resilient zero-day defense. These findings address RQ1 by clarifying the threat assumptions under which text modalities perform optimally, and highlight the need for multimodal or anomaly-aware integration to enhance robustness.

4.2 Vision-Based Approaches

Text-based detection models suffer from inherent limitations, as they are highly sensitive to variations or crafted patterns in the strings used by attackers. To overcome this, recent studies have actively explored detection methods that leverage non-textual visual elements of websites, such as layout, logos, colors, and buttons. Such vision-based analysis is particularly effective for detecting illicit sites designed on the principle of "brand impersonation" from the user's perspective. In this section, we provide a structured overview of vision modality–oriented detection techniques, as summarized in Table 2. By comparing the visual differences between legitimate and illicit websites, one can intuitively identify features such as layout, color schemes, and advertisement patterns that can be exploited by detection models. For instance, Fig. 6 presents screenshots of legitimate vs. illicit online gambling sites (a, b) and legitimate vs. illicit webtoon platforms (c, d), demonstrating visual cases that vision-based approaches can analyze.

Table 2: Vision-based AI techniques for illicit web threat detection.

Ref.	Approach	Input Data	Model/Architecture	Key Results/Contributions
Liu and Lee [63]	SIA-CNN: security indicator area classification	Cropped browser security indicator area screenshots	CNN classifier	Achieved F1-score 0.962 on imbalanced real-world dataset; demonstrated language-independent phishing detection; effective against spoofed visual cues but limited to SIA region only
Abdelnabi et al. [64]	VisualPhishNet: visual similarity detection for zero-day phishing	Full-page screenshots	Triplet CNN embedding	Constructed large-scale VisualPhish dataset; achieved 81.03% Top-1 match, ROC AUC 0.9879; robust to layout variations and zero-day attacks; generalizes beyond page-to-page matching
Ji et al. [65]	Robustness evaluation of vision-based models	451k real-world phishing screenshots	Comparative benchmarking of 7 models	First large-scale robustness study; revealed up to 18% accuracy drop under adversarial manipulations; highlighted evasion strategies and design flaws in vision-based detectors
Van Dooremaal et al. [66]	Hybrid text–visual fusion	DOM text + webpage screenshots	Combined text classifier + image similarity	Achieved 99.2% accuracy on target brand identification; hybrid fusion reduced misclassification vs. text-only; showed value of combining DOM and visual cues, though with higher computational cost
Dalgic et al. [67]	Phish-IRIS: compact descriptor–based phishing detection	Full-page screenshots	MPEG-7 descriptors + SVM/RF	Introduced Phish-IRIS dataset; achieved F1-score 90.5%, TPR 90.6%, FPR 8.5%; provided lightweight and fast solution; suitable for browser plugins and low-resource deployment

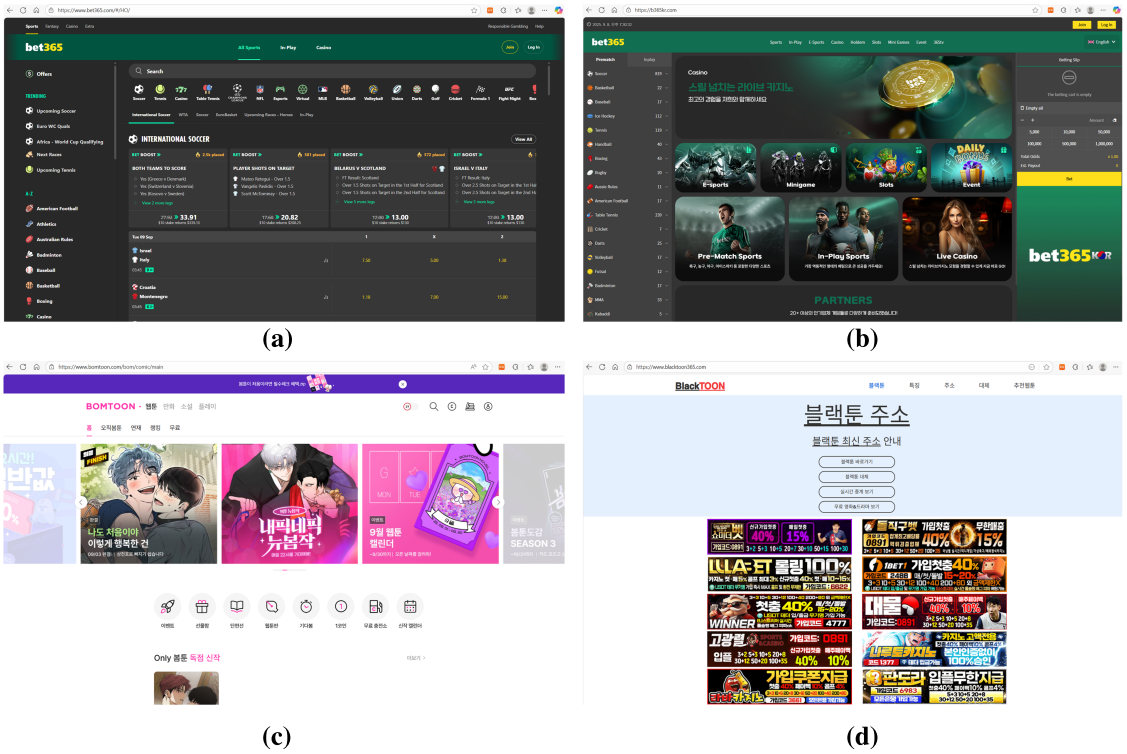

Figure 6: Sample screenshots for vision-based detection. (**a**) Legitimate online gambling site; (**b**) Illicit online gambling site; (**c**) Legitimate webtoon platform; (**d**) Illicit webtoon platform. The visual differences across these categories (e.g., layout consistency, advertisement patterns, and design quality) illustrate how vision-based approaches can leverage screenshot-level cues for distinguishing between benign and malicious websites. **Takeaway:** Screenshot-level cues can expose impersonation patterns beyond lexical analysis.

4.2.1 Vision-Based AI Techniques

The Security Indicator Area (SIA) refers to the visual elements located in the upper portion of a browser, including the URL address bar, HyperText Transfer Protocol Secure (HTTPS) padlock, tab icons, and brand logos, which users intuitively rely on to assess the trustworthiness of a website. Liu and Lee proposed an approach that automatically captures this SIA region, converts it into images, and classifies them using a CNN model. By doing so, the method achieves phishing detection in a manner that is independent of language or textual content. Even on an imbalanced real-world dataset (3843 legitimate samples and 1593 phishing samples), the model achieved an F1-score of 0.962, demonstrating its ability to overcome the language dependency and spoofing weaknesses of conventional URL/HTML-based detectors [63].

Abdelnabi et al. proposed VisualPhishNet, a model that leverages a triplet CNN to learn visual similarities between webpages belonging to the same brand, and then classify phishing attempts based on this learned similarity space. Unlike simple image matching, the model generalizes visual characteristics across webpages with diverse layouts and components, thereby achieving strong robustness against zero-day attacks and user interface obfuscations. In experiments, the model recorded a Top-1 accuracy of 81.03% and an Receiver Operating Characteristic (ROC) Area Under the Curve (AUC) of 0.9879. The evaluation was conducted on the VisualPhish dataset, which included 155 legitimate websites and 1195 phishing pages targeting those websites [64].

Ji et al. systematically evaluated the effectiveness and robustness of seven vision-based phishing detection models using a large-scale dataset comprising over 450,000 real phishing screenshots. Their study analyzed various evasion strategies, including logo removal, font perturbation, background color imitation, and logo repositioning. The results revealed that some logo-based models exhibited accuracy drops of over 18% under certain manipulations, and that these models were particularly vulnerable even to simple techniques such as logo replacement or removal [65]. This work emphasizes the need for designing vision-based detectors that incorporate adversarial evasion strategies from the attacker's perspective in order to enhance robustness.

To overcome the limitations of single-modality detection, hybrid models have recently been proposed that integrate textual information extracted from the DOM (e.g., title elements) with visual features obtained from webpage screenshots. Van Dooremaal et al. suggested a model that combines these two feature types to identify the legitimate websites being impersonated, thereby enabling phishing detection. Evaluated on a dataset of 2000 real phishing and benign websites, this model achieved a target identification accuracy of 99.2% [66]. Furthermore, the approach was designed to be deployable as a browser plugin, demonstrating its feasibility for real-time response in practical user environments.

Dalgic et al. introduced Phish-IRIS, a framework that summarizes webpage screenshots into Moving Picture Experts Group-7 (MPEG-7) and related compact visual descriptors (e.g., Scalable Color Descriptor (SCD), Color Layout Descriptor (CLD), Color and Edge Directivity Descriptor (CEDD)), which are then classified using lightweight ML models such as SVM or RF. Notably, the combination of the SCD with SVM achieved an F1-score of 90.5%, a True Positive Rate (TPR) of 90.6%, and an FPR of 8.5% [67]. With an average processing time of only 0.21 s per image, the approach is highly efficient and can operate without high-end Graphics Processing Unit (GPU), making it suitable for deployment as a browser plugin or phishing detector on mobile devices.

4.2.2 Synthesis and Implications of Vision-Based Approaches

Vision-based approaches address a distinct threat assumption: brand impersonation through visual mimicry. By analyzing webpage layouts, logos, and security indicator areas, these models operate independently of textual obfuscation and multilingual variation. As a result, vision-based systems are particularly effective against phishing campaigns that replicate legitimate brand identities.

However, empirical studies demonstrate that these models remain vulnerable to adversarial visual perturbations, including logo removal, color modification, background imitation, and subtle pixel-level attacks. Furthermore, the requirement for webpage rendering and screenshot acquisition introduces additional latency and infrastructure overhead, limiting deployment feasibility in high-throughput environments.

In zero-day scenarios, vision-based models may outperform lexical models when attacks rely on novel domain names but visually mimic trusted brands. Conversely, purely content-based scams lacking recognizable visual anchors may evade such systems. These observations clarify the operational boundaries of vision modalities in addressing RQ1 and underscore the importance of balancing robustness with deployment constraints (RQ3).

4.3 Graph-Based Approaches

Illicit websites are not confined to a single URL or HTML page but are instead constructed upon a complex infrastructure ecosystem involving inter-domain connections, shared IP structures, and common name servers. Such relational structures are difficult to capture using conventional text- or image-based detection methods; hence, in recent years, GNN-based detection approaches have attracted significant

attention. As shown in Table 3, this section summarizes the principal detection models leveraging GNNs, the corresponding graph construction strategies, and the advantages and limitations of GNNs.

Table 3: Graph-based AI techniques for illicit web threat detection.

Ref.	Approach	Graph Construction Elements	Model/Architecture	Key Results/ Contributions
Luo et al. [68]	Domain relation modeling with attention	Nodes: domain names, client IPs, resolved IPs; edges: client query relation, resolution relation, shared CNAME relation	AGCN (GCN + attention)	Accuracy 94.27%, F1 87.93% using only 10% labeled data; effectively integrates multiple relational graphs with attention weighting
Guo et al. [69]	Multi-entity phishing URL detection	URL strings, IPs, authoritative name servers; heterogeneous edges for URL–IP–NS associations	LBP + message passing	Achieved F1 ≈ 98.77%, accuracy 97.71%; robust with RF priors and cycle-removal strategy
Bilot et al. [70]	PhishGNN: website hyperlink structural analysis	Nodes: root & child URLs; edges: hyperlinks via tags	GNN + RF pre-classification	End-to-end feature learning, removes handcrafted feature dependency; multiple GNN architectures tested; high structural detection accuracy
Sun et al. [71]	GNN-IDS: intrusion detection via attack-flow graphs	Nodes: assets, vulnerabilities, actions; edges: host connections + traffic flows	GCN, GCN-EW, GAT + GNN explainer	Outperformed classical NN IDS; provides explainability and robustness against adversarial noise
Pujol-Perich et al. [72]	Host-connection graph for flow relations	Nodes: hosts and flows; edges: host-flow associations	Custom GNN message-passing IDS	Weighted F1 = 0.99 on CIC-IDS2017; robust under adversarial attacks; generalizes multi-flow attack detection
Unit 42, Palo Alto Networks [73]	Proactive malicious infrastructure discovery	Seed indicators; relations from hosting, certificates, phishing kits	GNN classifier with automated pivoting	Discovered thousands of new phishing & skimmer domains proactively; detection within 1–7 days of domain registration; applied at scale in real threat hunting

4.3.1 Graph-Based AI Techniques

A key factor in GNN-based domain detection lies in the definition of nodes (entities) and edges (relations). The AGCN-Domain model proposed by Luo et al. defines domains, client IPs, and resolved IP addresses as nodes and models three principal relations—client relation, resolution relation, and Canonical Name (CNAME) relation—into graph structures. Here, the client relation connects domains queried by the same client, the resolution relation connects domains mapped to the same IP, and the CNAME relation connects domains sharing identical CNAME records. The proposed model applies a Graph Convolutional Network (GCN) to each relational graph and integrates them by assigning weights to each relation through an attention mechanism, thereby constructing unified feature vectors. Experimental results demonstrate that

even with only 10% of the domains labeled, the model achieved 94.27% accuracy and an F1-score of 87.93%, thereby outperforming existing baselines [68].

Guo et al. proposed a method that constructs a heterogeneous graph integrating URL strings, IP addresses, and authoritative name servers, and applied a detection model based on Loopy Belief Propagation (LBP) to classify phishing URLs. The model dynamically adjusts edge potentials using similarity-based metrics and enhances stability through a combination of RF-based priors and a cycle-removal convergence strategy. On a large-scale dataset comprising 306,354 URLs, the approach recorded an F1-score of 98.77% and an accuracy of 97.71%, demonstrating superior performance and robustness compared with existing baselines [69].

PhishGNN introduces a framework that models the internal hyperlink structures of phishing websites as graphs, learning semantic relations via a GNN. Each node represents either a root URL or child URLs linked through internal hyperlinks, while edges are constructed from links extracted from HTML tags such as `<a>`, `<form>`, and `<iframe>`. PhishGNN incorporates initial labels for root and child nodes using a RF pre-classifier, which are subsequently integrated into the GNN for automatic embedding and message passing. In this way, the model effectively captures the structural characteristics of websites [70]. This framework reduces dependency on handcrafted features while supporting end-to-end learning and maintains extensibility to multiple GNN architectures.

GNN-IDS is an Intrusion Detection System (IDS) that integrates attack graphs with real-time measurements to construct input graphs, upon which models such as GCN, GCN with Edge Weights (GCN-EW), and Graph Attention Network (GAT) are applied. By combining static information within networks (e.g., assets, vulnerabilities) with dynamic behavior information (e.g., traffic, logs), the system is capable of learning complex attack scenarios. Experimental results reveal that the proposed GNN-IDS achieves superior accuracy and F1-scores compared to conventional neural network-based IDS. Furthermore, with the aid of GNNExplainer, the model provides interpretable detection results. It also demonstrates resilience under noisy or altered attack scenarios, indicating strong robustness for practical deployment [71].

GNN can enhance the detection of multi-flow attacks or interaction-based threats by modeling structural relations across multiple network flows. To overcome the limitations of traditional ML-based IDS that learn only individual flow features, Pujol-Perich et al. introduced the Host-Connection Graph (HCG), which represents flows as node–edge structures, and designed a specialized GNN message-passing architecture. Their model achieved an F1-score of 0.99 on the CIC-IDS2017 dataset and maintained detection accuracy even under adversarial attacks that manipulated packet sizes and inter-arrival times [72]. These findings demonstrate the potential of GNNs to capture structural patterns of emerging threats and ambiguous domains, enabling resilient detection. They further suggest that integration with IDS systems can substantially enhance flow-based detection. Beyond supervised classification, GNNs are also considered effective for unsupervised tasks such as graph anomaly detection, link prediction, and similarity assessment, as they can jointly exploit graph structures and node attributes to detect anomalous patterns at the node, edge, subgraph, or entire graph level [74]. This capability provides flexibility in identifying novel threats, ambiguous domain clusters, and anomalous flows arising from dynamic structural changes, thereby extending applicability to graph-based IDS research [75].

The Unit 42 project of Palo Alto Networks proposed a proactive discovery framework, illustrated in Fig. 7, that constructs graphs from a small set of known malicious domains (seed indicators) and applies a GNN-based classifier to identify related but previously undetected infrastructures. The system expands relational graphs using indicators such as IP co-hosting, Transport Layer Security (TLS) certificate similarity, and registrar information. By leveraging GNN learning, the framework reduced detection delays to within 1–7 days after domain registration, effectively identifying phishing and skimmer domains before

they became weaponized [73]. This demonstrates the feasibility of proactive infrastructure hunting strategies that preemptively block malicious assets.

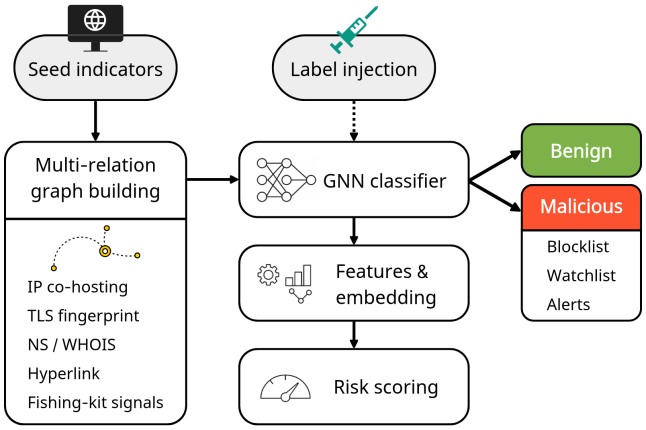

Figure 7: Proactive domain-infrastructure discovery model. Starting from a few seed indicators, a multi-relation graph is built by aggregating. Labels are injected for training only. The graph is processed by a GNN classifier to produce embeddings and risk scores, enabling early identification of related infrastructure and driving operational outputs (benign/malicious decision) [73]. **Takeaway:** Infrastructure-level graph modeling enables early identification of malicious domain clusters, supporting proactive rather than purely reactive threat mitigation.

While GNN-based detection techniques offer advantages in modeling the structural relationships of illicit web ecosystems, persistent challenges remain. These include the limitations of semi-supervised learning under label scarcity, performance degradation due to heterophily and noise within graphs, and class imbalance issues. Furthermore, GNNs often face challenges in interpretability, restricting their adoption in high-trust security analyses, and are vulnerable to privacy attacks (e.g., membership inference, model extraction) in distributed or collaborative detection environments. To address these issues, active research efforts are exploring strategies such as label denoising, structural redesign, imbalance correction, differential privacy, and federated GNN frameworks [76].

4.3.2 Synthesis and Implications of Graph-Based Approaches

Graph-based approaches shift the detection paradigm from single-instance classification to infrastructure-level analysis. By modeling domain-IP relationships, hosting clusters, certificate reuse, and hyperlink structures, GNN-based systems capture latent structural dependencies within illicit web ecosystems.

This modality is particularly advantageous under zero-day conditions, where newly registered domains may evade lexical and visual inspection but remain structurally linked to malicious infrastructure. However, graph construction requires large-scale data aggregation, label availability, and periodic updates, introducing computational and operational complexity.

Performance may degrade under conditions of graph noise, heterophily, or label scarcity. Additionally, interpretability and privacy concerns remain open challenges. These trade-offs highlight the ecosystem-level strengths of graph models while exposing scalability and deployment limitations, directly addressing RQ1 and RQ4.

4.4 Sequence Modeling Approaches

Traditional URL-based rule filters and HTML analysis methods have inherent limitations, as they rely on fixed patterns that are difficult to adapt to complex and dynamic cyber threats. Against this backdrop, recent studies have applied Transformer-based Natural Language Processing (NLP) models, which process sequence data as input, to the detection of illicit websites. As shown in Table 4, this section reviews web content analysis approaches employing sequence models such as Transformer, Bidirectional Encoder Representations from Transformers (BERT), and Generative Pre-trained Transformer (GPT) families.

4.4.1 Sequence Modeling AI Techniques

Although URLs appear short and simple in structure, they actually contain information units at the character and token levels that reflect the attacker's obfuscation strategies. Recent studies have considered this property and proposed Transformer-based models such as URLTran, which directly ingest URL strings as input for phishing classification. These approaches have been reported to achieve higher detection accuracy and robustness than conventional CNN- or RNN-based architectures [77]. In particular, the URLTran_BERT model achieved a TPR of 86.80% at a very low FPR of 0.01%, outperforming URLNet (71.20%) and Texception (52.15%). Moreover, by incorporating character-level and byte-level subword tokenization, URLTran demonstrated superior contextual understanding and generalization for phishing URLs with ambiguous word boundaries or deliberately manipulated structures [78].

Table 4: AI-based sequence modeling approaches for illicit web threat detection.

Ref.	Approach	Input Data	Model/Architecture	Key Results/Contributions
Maneriker et al. [78]	Transformer -based URLTran	URL strings	Transformer (BERT, RoBERTa, custom vocab)	TPR 86.80% at FPR 0.01%; AU-ROC 0.9993; robust against homoglyph & compound attacks
Kumar [79]	Comparative fine-tuned BERT vs. DistilBERT classifiers	URL dataset	BERT, DistilBERT	DistilBERT: accuracy 99.65%, F1 99.64%, faster and smaller than BERT; BERT: accuracy 99.29%, F1 99.00%
Mahendru and Pandit [80]	Comparative DeBERTa V3 vs. GPT-4/Gemini	Mixed text datasets	DeBERTa V3, GPT-4, Gemini 1.5	DeBERTa V3: recall 95.17%, F1 91.75%; GPT-4: recall 91.04%, F1 80.52%; LLMs effective in phishing detection
Kiazolu [81]	Hybrid BERT + LSTM (feature fusion)	URL sequences + webpage text content	BERT (semantic) + LSTM (structural)	Accuracy 95%, F1 0.95, recall 0.96; outperforms single-modal baselines

(Continued)

Table 4 (continued)

Ref.	Approach	Input Data	Model/Architecture	Key Results/Contributions
Meléndez et al. [82]	Comparative ML vs. Transformer	Large phishing email corpora	Traditional ML models, transformer models	DistilBERT: accuracy 98.99%, F1 0.9911; higher than NB (96.44%) and SVM (98.76%)
Koide et al. [83]	ChatPhishDetector: LLM prompt-based phishing detection	Web crawler data	GPT-4V, LLM-based reasoning	GPT-4V: precision 98.7%, recall 99.6%, F1 99.2%; detects brand impersonation & SE tactics across different languages

In recent years, pre-trained language models such as BERT, RoBERTa, and DistilBERT have been widely studied for cybersecurity text analytics, including URL and email content. These models achieve significantly higher precision and recall compared to traditional keyword-based detection methods. Among them, the lightweight DistilBERT maintains more than 95% of BERT's language understanding capability on the GLUE benchmark, while offering a smaller model size and faster inference speed. In Kumar's comparative study on phishing URL detection, DistilBERT achieved 99.65% accuracy and 99.64% F1-score, slightly but consistently outperforming BERT (99.29% accuracy and 99.00% F1-score), thereby demonstrating practical detection performance even under resource-constrained environments [79].

The SecureNet project further compared multiple pre-trained language models for phishing detection. Results indicated that DeBERTa V3 achieved the best performance, with 95.17% recall and an F1-score of 91.75% on the HuggingFace phishing dataset. In contrast, GPT-4 recorded 91.04% recall and an F1-score of 80.52% under the same conditions, showing particular strength in web and HTML-based phishing scenarios [80]. These experiments demonstrate that LLMs can deliver meaningful detection performance beyond text generation, suggesting their potential in integrated detection strategies that span diverse data types such as URLs, HTML, emails, and SMS.

To leverage complementary information from both URL structure and web content, Kiazolu proposed a hybrid model combining LSTM and BERT. The model extracts structural features such as domain length, use of special characters, and subdomain patterns from URL sequences using LSTM, while simultaneously generating deep semantic embeddings of webpage content using BERT. The embeddings are fused through feature fusion, and the final classifier predicts phishing status based on the combined representation. Experimental results showed accuracy of 95%, F1-score of 0.95, and recall of 0.96, surpassing single-modal baselines [81]. These findings indicate that integrating sequential URL structures with semantic content features can enhance both detection accuracy and adaptability.

Meléndez et al. demonstrated that lightweight pre-trained language models such as DistilBERT also achieve excellent results in phishing email detection compared to traditional ML methods. For instance, DistilBERT achieved 98.99% accuracy and an F1-score of 0.9911, slightly but consistently outperforming Naive Bayes (NB) (96.44%) and SVM (98.76%) [82]. This suggests that Transformer-based models can effectively replace conventional approaches not only in terms of accuracy improvement but also in terms of model efficiency and deployability.

The emergence of LLMs has opened the possibility of zero-shot approaches for phishing detection without additional training. As illustrated in Fig. 8, ChatPhishDetector utilizes GPT-4V to determine phishing status based on prompts constructed from website screenshots, HTML, Optical Character Recognition (OCR) text, and URLs. This system was designed to overcome the limitations of conventional detection approaches. Experimental results demonstrated that GPT-4V achieved a precision of 98.7%, recall of 99.6%, and F1-score of 99.2%, outperforming other LLMs and traditional detection systems. Furthermore, it effectively identified brand impersonation and social engineering techniques [83]. Notably, the system demonstrated high practicality and scalability in real-world settings, as it can generalize across multiple languages and domains based solely on prior training.

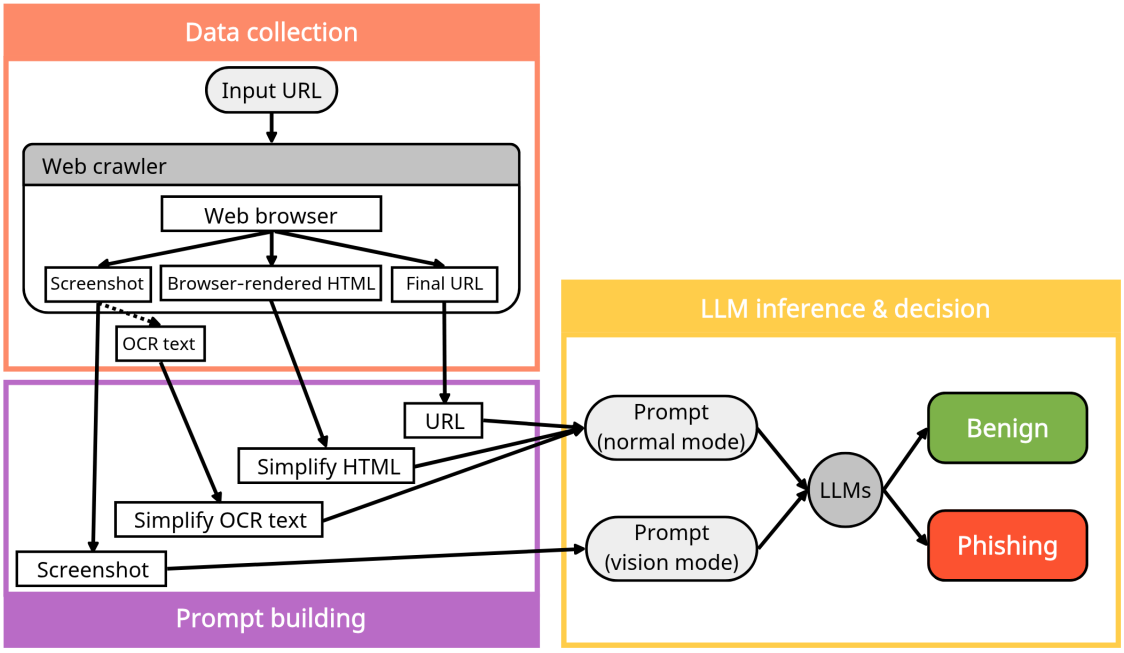

Figure 8: LLM-based phishing detection model (ChatPhishDetector). URL/HTML/screenshot (+OCR) are fused into normal (text) or vision (text+image) prompts, yielding a benign/phishing decision [83]. **Takeaway:** Prompt-driven LLM architectures demonstrate how multimodal contextual reasoning can unify textual and visual signals for adaptive phishing detection.

4.4.2 Synthesis and Implications of Sequence Modeling Approaches

Transformer-based and sequence modeling approaches introduce contextual reasoning capabilities that extend beyond handcrafted features. By learning semantic dependencies within URLs, HTML structures, and phishing messages, these models exhibit improved resilience to superficial lexical manipulation.

In zero-day settings, sequence models demonstrate stronger generalization than traditional classifiers when trained on diverse corpora. However, their performance remains sensitive to domain shifts and adversarial prompt engineering. Moreover, the computational demands of large language models introduce latency and resource constraints, particularly in real-time detection environments.

While LLM-based detection frameworks show promise for adaptive and cross-lingual threat analysis, their reliability depends heavily on calibration, evaluation protocols, and robustness testing. These findings reinforce the need for evaluation rigor (RQ2) and deployment-aware model design (RQ3).

4.5 Cross-Modality Comparison of AI-Based Detection in Illicit Web Ecosystems

The comparative analysis between the modalities of AI-based detection in the illegal web ecosystem is shown in Table 5.

Table 5: Cross-modality comparison of AI-based detection in illicit web ecosystems.

Modality	Typical Inputs	Representative Models	Strengths	Weaknesses	Zero-Day Robustness
Text-based	URL strings, HTML source, email headers, DOM text	RF, SVM, CNN, LSTM, BERT, DistilBERT	Scalable, low latency, easy integration, minimal crawling	Token obfuscation, homoglyph attacks, multilingual drift, LLM-based rephrasing	Moderate (improves with transformer models)
Vision-based	Webpage screenshots, logo regions, security indicator areas	CNN, triplet networks, visual similarity models	Effective for brand impersonation, language-independent	Logo removal, visual perturbations, layout modification, adversarial noise	Strong for brand-based phishing; weak for non-brand scams
Graph-based	Domain-IP relations, DNS records, certificate reuse, hyperlink structures	GCN, GAT, AGCN, heterogeneous GNNs	Captures infrastructure-level abuse, detects malicious clusters	Label scarcity, graph noise, heterophily, privacy constraints	High (resilient to domain-level mutation)
Sequence/ Transformer	Tokenized URLs, HTML sequences, phishing emails, multimodal prompts	Transformer, BERT, RoBERTa, LLM-based reasoning	Contextual understanding, cross-lingual capability, adaptable	Domain shift, prompt manipulation, computational cost	High (better generalization to unseen patterns)
Hybrid	Combined URL, HTML, visual, graph, metadata inputs	Ensemble models, fusion architectures	Improved robustness, complementary detection signals	Increased complexity, synchroniza-tion challenges	Very high (if well-calibrated)

The comparative analysis highlights that no single modality provides comprehensive resilience across all illicit web threat scenarios. Text-based methods offer high scalability but remain vulnerable to semantic obfuscation. Vision-based approaches excel in brand impersonation detection yet incur rendering overhead.

Graph-based models demonstrate strong ecosystem-level robustness but require substantial infrastructure data. Sequence models improve contextual generalization but introduce computational trade-offs. Hybrid architectures appear most promising for zero-day resilience, though their deployment complexity necessitates careful system design. These findings directly address RQ1 and RQ3 by clarifying the interplay between modality strengths, operational constraints, and generalization capacity.

5 System Architecture and Pipelines

5.1 Data Collection and Labeling

The performance of AI-based illicit website detection systems largely depends on the quality of the input data. To train high-performance models, it is essential to secure a sufficient amount of high-quality labeled data, and thus strategies for data collection and labeling must be carefully considered at the early stages of model design. This section reviews several dataset construction cases, focusing on key collection methods, labeling procedures, feature design principles, and quality evaluation techniques.

For data collection, widely used public phishing domain repositories such as PhishTank and OpenPhish are typically employed. In addition, domain lists scraped from Computer Emergency Response Team (CERT), security companies, research institutes, and public data sources can be incorporated. Based on these domain lists, a dedicated environment is configured to automatically collect webpage HTML sources and metadata, with various feature design and labeling methodologies being introduced to ensure data quality.

PhishStorm is a system built for real-time phishing URL detection. It constructed a balanced dataset of 96,018 entries using 48,009 phishing URLs collected from PhishTank and 48,009 legitimate URLs obtained from DMOZ. Each URL was split into its registered domain and the remaining part, and twelve features were extracted based on intra-URL relatedness and URL popularity using query data from Google Trends and Yahoo Clues. These features were computed within a distributed streaming architecture and Bloom filters to achieve real-time performance. Using a RF classifier, PhishStorm achieved high classification performance, with 94.91% accuracy and a 1.44% FPR [84].

Verma et al. emphasized that the performance of AI models in the security domain is heavily influenced by data quality, and to this end, they constructed a public dataset called IWSPA-AP for phishing email detection. The dataset was built by collecting phishing and legitimate emails from diverse sources, including university IT departments, Wikileaks, and the Nazario dataset. The collected emails underwent cleaning processes such as URL replacement, organization name normalization, signature removal, and HTML tag sanitization. However, from the earliest version, classification accuracy consistently exceeded 99%, and further refinements yielded little change. Consequently, they proposed a new quality indicator termed data difficulty [85]. This case highlights the importance not only of accurate labeling but also of identifying and correcting dataset construction issues that may lead to overly optimistic results.

Khan et al. applied and compared various ML algorithms to three publicly available datasets for phishing website detection. The first dataset comprised 10,000 webpages (5000 phishing and 5000 legitimate) collected between 2015 and 2017. The second was the University of California, Irvine (UCI) repository dataset containing 11,055 URLs (4898 phishing and 6157 legitimate). The third was a multi-class dataset of 1353 websites (702 phishing, 548 legitimate, and 103 suspicious). Each dataset included diverse attributes such as URL structure, domain information, Secure Sockets Layer (SSL)/TLS status, and page rank. Principal Component Analysis (PCA) was employed to assess attribute importance and eliminate redundant variables. Experiments were conducted using DT, SVM, RF, NB, K-Nearest Neighbors (KNN), and Artificial Neural Network (ANN), with RF and ANN achieving the best performance, recording over 97% accuracy on the

first dataset [86]. This study underscores the importance of enhancing generalizability by leveraging datasets from multiple sources and conducting attribute analyses.

Aung and Yemana presented a detailed methodology for the construction and refinement of URL datasets for phishing detection. They collected phishing URLs from PhishTank and legitimate domains from IP2Location, subsequently crawling up to 100 webpages per domain and randomly extracting legitimate URLs. This process produced both a balanced dataset (10,000 phishing and 10,000 legitimate) and an imbalanced dataset (5000 phishing and 50,000 legitimate). Preprocessing steps included URL length restrictions and the removal of GET parameters. The proposed URL-Tokenizer, which combined BERT and WordSegment tokenizers, successfully segmented URLs into meaningful units, drastically reducing the number of out-of-vocabulary terms. As a result, the system achieved detection accuracies of 95.7% on the balanced dataset and 97.7% on the imbalanced dataset [87]. This case demonstrates that a high-quality dataset suitable for learning can be constructed solely from refined URLs, beyond mere collection.

Sánchez-Paniagua et al. introduced PILWD-134K, a dataset designed to ensure both timeliness and representativeness in phishing detection research. This dataset, collected between August 2019 and September 2020, contains 134,000 verified samples, including URLs, HTML code, screenshots, web technology analyses, offline replicas, and PhishTank metadata. Noting that 77% of phishing websites contain login forms, they ensured that legitimate sites in the dataset mirrored this proportion by crawling based on Quantcast and the Majestic Million. Phishing URLs were collected in real time from PhishTank, with content and final redirection URLs also stored. Following a three-stage filtering process, 66,964 phishing samples and 129,742 legitimate samples were retained. With 54 proposed features and a LightGBM classifier, the dataset achieved 97.95% accuracy [88]. This work demonstrates a practical and generalizable approach by reflecting login page distribution, employing multi-source data collection, and designing independent features.

Prasad and Chandra proposed the PhiUSIIL framework, which integrates a URL similarity index with incremental learning for phishing URL detection. A distinctive feature of this framework is its large-scale dataset construction, comprising 134,850 legitimate URLs and 100,945 phishing URLs. Legitimate domains were collected from Open PageRank, while phishing domains were obtained from PhishTank, OpenPhish, and MalwareWorld. Automated Java-based scripts were used to archive HTML content and extract dozens of features, including TLD, number of subdomains, HTTPS usage, iframes, and form submission methods. Additional derived features, such as CharContinuationRate, URLTitleMatchScore, and TLDLegitimateProb, were also designed to enhance dataset quality. Combining pre-training with incremental learning, the PhiUSIIL dataset achieved a detection accuracy of 99.79% [89]. Considering the short lifespan of phishing sites, this study is notable for establishing a real-time monitoring and immediate crawling environment, thereby providing a continuously expandable high-quality learning resource.

Hannousse and Yahiouche defined a total of 87 hybrid features by integrating URL-based, content-based, and external service-based attributes, and constructed a reproducible and extensible dataset for phishing detection, as shown in Fig. 9. The collected data included automated HTML content and external metadata. Experimental results demonstrated a detection accuracy of 96.83% using a RF classifier [90]. This study provides empirical evidence for the effectiveness of hybrid dataset-based approaches combining multiple feature categories.

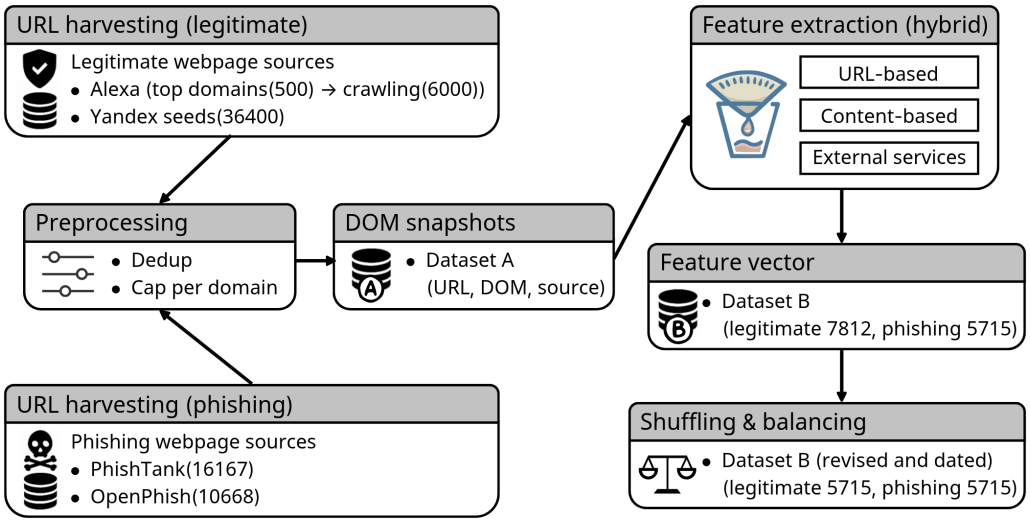

Figure 9: Hybrid dataset construction pipeline for phishing detection. URLs are harvested from legitimate and phishing sources, preprocessed, and stored as DOM snapshots (dataset A). Hybrid features (URL, content, external services) are extracted to produce feature vectors (dataset B), which are then shuffled and balanced; the final dataset is dated for versioning [90]. **Takeaway:** A structured and version-controlled data pipeline improves dataset reproducibility and mitigates biases arising from asynchronous collection and feature inconsistency.

5.2 Real-Time Detection System Architectures

The detection of illicit websites must satisfy not only accuracy requirements but also practical considerations such as latency, adaptability, and deployment environments. Given that many malicious domains are short-lived and abandoned shortly after registration, the real-time capability of a detection system is a critical determinant of its overall detection rate. Consequently, recent years have witnessed the proposal of various real-time detection system architectures. As shown in Table 6, this section provides a comparative analysis of representative systems, focusing on their architectural components.

Table 6: Comparative overview of real-time threat detection systems in illicit web ecosystems.

	PhishIntel [91]	**Hybrid Detection System [92]**	**Real-Time Alert System [93]**	**PhishLang [94]**
Input source	URL submission via API	Browser extension	Browser extension	Local extension
Crawling/acquisition	Cache + blacklist + HTML + screenshot	Screenshot + WHOIS/SSL	HTML scraping + SSL	None (source parsing)
Preprocessing	URL normalization, blacklist check, result cache update	111 lexical features + visual features	URL feature extraction + SSL certificate check + heuristic filtering	Text normalization, tokenization; selective parsing of tags
Core model	RBPD (vision+HTML)	XGBoost + MobileNet	RF + 1D CNN	MobileBERT

(Continued)

Table 6 (continued)

	PhishIntel [91]	Hybrid Detection System [92]	Real-Time Alert System [93]	PhishLang [94]
Execution location	Hybrid (client/server backend)	Client + backend pipeline	Client + backend pipeline	Fully client-side
Response latency	Very low (fast path)	Low	Moderate	Very low
Zero-day capability	Strong (fast-slow architecture + RBPD for novel phishing)	Effective via visual similarity and URL features	Adaptive ML but limited by feature generalization	High adaptability (robust against obfuscation, adversarial attacks)
Explainability	Moderate (brand intention analysis in RBPD)	Limited (feature importance available in XGBoost)	Low (basic ML confidence scores)	Enhanced (GPT-based textual explanations; interpretable MobileBERT outputs)
Unique features	Fast-slow arch.; Outlook plugin	Hybrid lexical + visual; Chrome plugin	Browser/email alerts	Open-source; privacy-preserving

The PhishIntel system is based on a fast–slow task system architecture, in which the detection process is divided into fast and slow tasks (Fig. 10). The fast task quickly returns detection results using caches and blacklists, while URLs that remain unresolved are forwarded to the slow task for webpage crawling and model-based analysis. This design reduces the average response time to 0.016 seconds while maintaining robust zero-day phishing detection capability, thereby achieving a balance between responsiveness and detection performance [91]. The hybrid detection system by Gnanesh et al. and the real-time alert system by Pawar et al. both operate through browser extensions that collect URLs when users visit websites or open emails. On the backend, these URLs undergo further inspection, including HTML content crawling, SSL/TLS certificate validation, and screenshot-based visual similarity analysis. This enables more precise and real-time phishing detection [92,93].

Input URLs are subjected to basic feature extraction, such as length, domain structure, and suspicious keywords, in addition to analyses of HTML structure, SSL/TLS certificate details, redirect logs, and iframe composition. The hybrid system by Gnanesh et al. combines XGBoost-based lexical analysis with MobileNet-driven visual similarity analysis to provide more resilient detection [92]. PhishLang executes all preprocessing on the client device. It parses meaningful elements directly from webpage HTML source code and feeds them into MobileBERT. The model tokenizes and embeds the input text to classify phishing attempts, offering effective resistance against previously unseen URLs and zero-day phishing attacks. With its lightweight architecture, MobileBERT achieves an average inference time of approximately 0.4 s and a memory footprint below 74 MB, ensuring stable real-time detection even in low-resource environments [94].

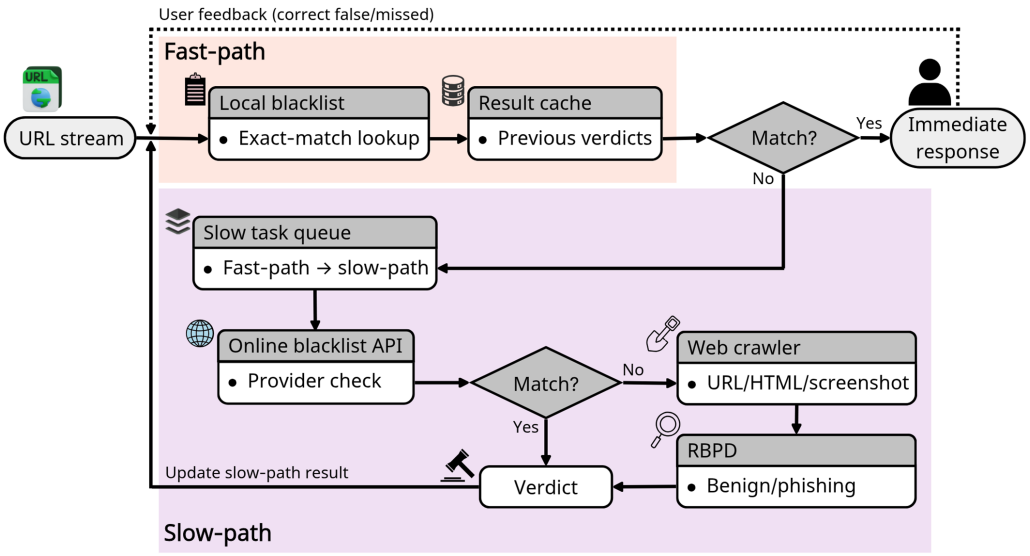

Figure 10: Fast–slow task architecture for phishing detection (PhishIntel). Incoming URLs first take a latency-optimized fast path. Unmatched URLs are enqueued to the slow path for deep analysis using an online blacklist, a crawler, and RBPD. Slow-path verdicts update the cache, achieving sub-second response while retaining robust zero-day detection capability [91]. **Takeaway:** The fast–slow design balances sub-second response latency with deep inspection, illustrating a practical trade-off between real-time usability and zero-day robustness.

PhishIntel employs a Reference-Based Phishing Detection (RBPD) model, which classifies webpages by comparing their similarity with known phishing samples. Other systems adopt traditional ML, DL, or MobileBERT-based approaches, with some additionally incorporating CNN-based visual analysis. The execution environment varies among server-centric, client-centric, and hybrid client–server configurations, which influences latency, scalability, and privacy guarantees. Finally, inference results are communicated to users through browser extensions or backend APIs, and certain systems incorporate user feedback loops to refine models via continual retraining [91–94]. Such architectures enable continuous learning frameworks, thereby enhancing zero-day resilience.

5.3 Response System Architectures

Illicit website detection systems should not merely stop at detection; they must be followed by actual blocking measures and coordinated responses at the network level. As shown in Table 7, this section examines various system architectures for response, including blacklist automation, network filtering, and user-driven blocking frameworks.

Table 7: Comparative overview of threat response systems in illicit web ecosystems.

	Google Phishing Classifier [95]	**BlackEye [96]**	**Spamhaus, etc. [97–99]**	**PhishChain [100]**
Type	Large-scale ML-based URL blacklist	ML-based IP blacklist	External feed filtering	Decentralized crowd system

(Continued)

Table 7 (continued)

	Google Phishing Classifier [95]	BlackEye [96]	Spamhaus, etc. [97–99]	PhishChain [100]
Automation mechanism	URL/content/host feature extraction → auto classification → blacklist updating	Multi-log aggregation → ML model (LR/RF) → blacklist generating	Real-time feed ingestion → firewall enforcement	User voting → blockchain record → phish score (PageRank)
Strengths	High accuracy; large-scale coverage; fast update	Low false positives; fast detection; adaptive	Real-time updates; global coverage; easy integration	Transparent; tamper-proof; resilient to abuse
Limitations	Requires massive data; less agile to new threats	Data quality dependent; complex preprocessing	Feed dependency; false positive risk	High implementation cost; user trust modeling needed

Google disclosed its large-scale ML-based phishing detection system in a 2010. This system automatically analyzes millions of webpages daily and classifies them as phishing or benign by leveraging a wide range of features such as URL, HTML, and hosting information. The classification results are reflected in real time in the Safe Browsing blacklist, which is integrated into Chrome, Firefox, Safari, as well as Gmail and the Google search engine. Remarkably, despite being trained on approximately 10 million noisy samples, the model consistently achieved a high performance with an average precision of 97.5% and recall of 94.9%, enabling automated detection and blocking of most URLs within a short time frame [95]. This design effectively counters the typically short lifespan of phishing webpages.

The BlackEye framework is designed to aggregate and analyze diverse security logs—such as those from web firewalls, network firewalls, and Intrusion Prevention Systems (IPS)—at the IP level. Using ML models, it automatically detects malicious IP addresses and generates blacklist candidates. Evaluations on datasets collected in operational security environments demonstrated that BlackEye identified malicious IPs on average 27 days earlier than manual analysis and reduced the rate of incorrectly blacklisted IPs by more than 89%, significantly lowering false positives. In particular, data refinement through Ridge regression improved training quality, while Logistic Regression (LR) and RF models were experimentally verified to effectively generate real-time blacklists [96].

At the level of network firewalls or ISPs, external security vendors such as Spamhaus and CrowdSec provide IP blocklist feeds that can be subscribed to for real-time filtering. For example, the Spamhaus data feed updates within an average of 30 s and offers a wide range of real-time integration solutions, enabling rapid response to emerging threats [97]. Palo Alto Networks' firewall systems provide an External Dynamic List (EDL) feature, allowing administrators to dynamically import IP, URL, or domain lists managed externally. Security administrators can reference these lists directly in security policies to automatically block malicious IPs or URLs. The firewall then refreshes these lists according to configured intervals, ensuring up-to-date defenses against the latest threats [98]. Bellekens further emphasized that dynamic IP blocklist

systems based on external threat feeds can update on a minute-level basis, thereby extending the scope of real-time threat response [99].

PhishChain is a decentralized URL blacklist system that employs blockchain technology to evaluate URLs submitted by users. Its basic architecture is illustrated in Fig. 11. Unlike centralized validators, it relies on majority user assessments and applies a PageRank-based truth discovery algorithm to compute a phishing score for each URL. All records are immutably stored on the blockchain, ensuring both transparency of blocking criteria and post-hoc auditability [100].

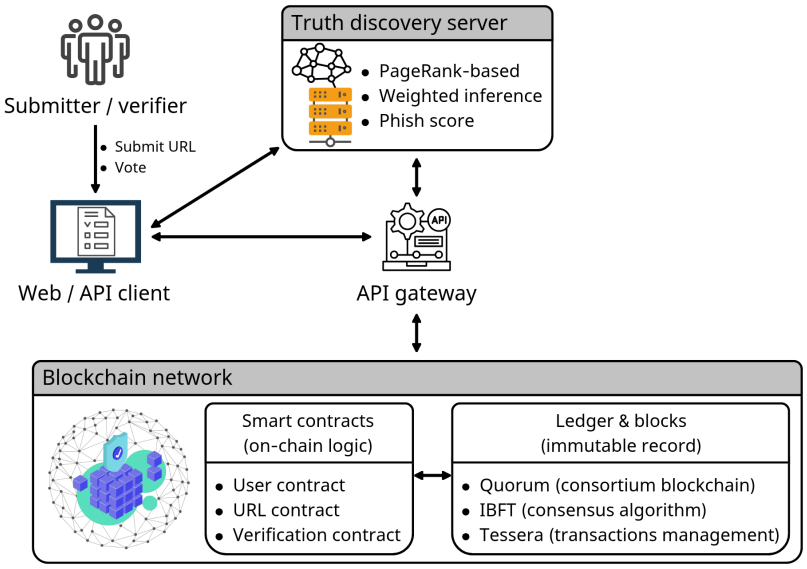

Figure 11: Decentralized crowd-sourced URL blacklisting architecture and workflow (PhishChain). The Web/API client submits URLs and votes via an API gateway; a PageRank-based truth discovery server runs off-chain, reads on-chain votes/status, and writes the computed phish score back on-chain through smart contracts. The consortium blockchain (Quorum with IBFT and Tessera) maintains the immutable ledger for transparency and auditability [100]. **Takeaway:** Decentralized validation mechanisms enhance transparency and auditability in phishing blacklisting, though they introduce complexity in trust modeling and scalability.

As summarized in Table 7, these response systems exhibit distinct strengths and limitations depending on their architectural design. Large-scale automated classifiers provide high accuracy and scalability, but may be slower to adapt to novel attacks. In contrast, real-time external feed integration offers immediacy but introduces dependency risks and potential false positives. Decentralized approaches ensure transparency and integrity, but pose challenges in system complexity and in designing robust models for user trust evaluation.

6 Benchmarks and Datasets

6.1 Public Phishing Domain Lists and Benchmark Datasets

The performance evaluation of AI-based illicit website detection models depends on the availability of consistent criteria and comparable datasets. To date, the most widely used public phishing domain lists are PhishTank and OpenPhish, upon which a variety of benchmark datasets have been constructed. This section summarizes the structure and utility of these lists, the characteristics of derivative datasets, and key considerations when employing them.

6.1.1 Public Phishing Domain Lists

PhishTank is a community-driven phishing domain verification platform operated by Cisco Talos. Users submit suspicious URLs, and the community determines their phishing status through voting. Since its launch in 2006, approximately 9.1 million suspicious URLs had been submitted and evaluated as of August 2025, with tens of thousands of them still active. PhishTank provides open access through APIs and JavaScript Object Notation (JSON)-based downloads, and its data are integrated into web browsers, email services, and various security solutions [101,102].

OpenPhish is an automated, ML-based phishing intelligence platform, primarily integrated into enterprise security solutions to detect and respond to phishing threats in real time. The platform continuously collects millions of webpages to identify phishing URLs and provides free as well as paid subscription plans, which differ in terms of update frequency and scope of phishing data. Supplied metadata include URL, target brand, IP and Autonomous System Number (ASN), SSL/TLS certificate information, country, language, and timestamp, making the dataset suitable for forensic analysis and threat intelligence [101,103]. Moreover, OpenPhish offers an academic use program, under which non-commercial users from accredited institutions such as universities and research centers may receive real-time phishing feeds and historical archives for a limited period. Such data may be employed in scholarly research with proper source acknowledgment and in compliance with legal and ethical requirements [104].

6.1.2 Public Phishing Benchmarks and Datasets

The phishing websites dataset by Ariyadasa et al. comprises approximately 80,000 instances, including 50,000 legitimate websites and 30,000 phishing websites. Each instance consists of a URL mapped to its corresponding HTML page, with the `index.sql` file containing five fields: rec_id, url, website, result (0: legitimate, 1: phishing), and created_date. Legitimate data were collected from Google searches and the Ebbu2017 Phishing Dataset, while phishing data were gathered from PhishTank, OpenPhish, and PhishRepo. Automated scripts continuously monitored PhishTank and OpenPhish to fetch pages promptly and minimize issues with disappearing resources [105].

The phishing and benign domain dataset by Hranický et al. consists of approximately 430,000 benign domains collected via Cisco Umbrella and 36,993 phishing domains sourced from PhishTank and OpenPhish. For each domain, it provides rich multidimensional features derived from DNS records, IP-related data, WHOIS/Registration Data Access Protocol (RDAP) information, SSL/TLS certificate fields, and GeoIP metadata [106]. This dataset is particularly useful for ML-based phishing detection studies focusing on network-level detection models.

The dataset developed by Feng and Yue integrates 800,000 legitimate URLs from Common Crawl with 759,485 phishing URLs from PhishTank. Designed for RNN-based models (LSTM, Gated Recurrent Unit (GRU), BiLSTM, BiGRU), it relies solely on raw URL strings as inputs without handcrafted feature extraction [107,108]. Its structure is also adaptable to Transformer-based models, making it suitable for research on pattern learning from raw sequences.

The phishing database by Krog and Chababy, maintained as an open-source threat intelligence repository on GitHub, periodically validates the activity status of known phishing domains using the PyFunceble tool. As of August 2025, it contained about 460,000 phishing domains and 720,000 phishing links [109]. Its continuously updated nature makes it valuable for real-time model testing and integration into operational threat-response systems.

Vrbančic et al. released a phishing detection dataset with 111 detailed features, 96 extracted directly from URL strings and 15 from external services. Two variants are provided: a balanced version (30,647

phishing, 27,998 legitimate; total 58,645) and an imbalanced version (30,647 phishing, 58,000 legitimate; total 88,647). Features include domain length, counts of various delimiters, presence of IP addresses, and inclusion of email addresses. Phishing URLs were sourced from PhishTank, while legitimate URLs came from Alexa top domains and community-labeled news sites. Data were acquired using a automated Python-based pipeline [110,111]. This dataset has been widely adopted for feature engineering and benchmarking of URL-based phishing detection models.

The phishing-dataset by Alvarado is a multimodal text dataset that includes URLs, emails, SMS, and HTML pages. Each sample contains text data with labels (0: benign, 1: phishing). It comprises more than 800,000 URLs, approximately 6000 SMS, about 18,000 emails, and 30,000 instances out of 80,000 collected HTML webpages. A combined_reduced version downscales URLs by 95% to balance across modalities, making it well-suited for BERT-based phishing detection experiments [112].

BitAbuse is a restoration-oriented dataset focusing on visually perturbed phishing sentences (VP sentences). It was constructed from more than 260,000 real phishing emails collected via `bitcoinabuse.com`, resulting in about 320,000 sentences, of which 26,591 VP sentences were manually annotated with original, non-perturbed labels. Restoration experiments demonstrated that Character BERT trained on BitAbuse achieved up to 96.56% accuracy in restoring VP sentences [113,114]. Owing to its real-world origin, manually restored labels, and diverse perturbation types, BitAbuse is valuable not only for adversarial robustness testing but also for applications in digital forensics, secure messaging, and high-performance text restoration.

Table 8 compares the key characteristics of these datasets.

Table 8: Datasets and benchmarks for phishing threat detection.

Dataset	Source(s)	Size	Key Characteristics	Use Cases
Ariyadasa's dataset [105]	PhishTank, OpenPhish, PhishRepo, web scraping, public data	80 K instances (30 K phishing)	Live HTML crawled; structured SQL index; balanced classes	Supervised ML/DL model training with HTML context
Hranický's dataset [106]	PhishTank, OpenPhish, Cisco Umbrella	470 K domains (37 K phishing)	Multimodal metadata: DNS, IP, WHOIS, TLS; domain-level focus	Network-based detection, graph-based models (e.g., GNNs)
Feng's dataset [107,108]	PhishTank, Common Crawl	1.56 M raw URLs (760 K phishing)	Only lexical input; no feature engineering; RNN/Transformer-ready	Deep learning on raw URL sequences; model interpretability
Krog's database [109]	GitHub repo	460 K phishing domains, 720 K phishing links (as of Aug 2025)	Continuously updated; status-checked with PyFunceble	Real-time model testing; threat intelligence system integration

(Continued)

Table 8 (continued)

Dataset	Source(s)	Size	Key Characteristics	Use Cases
Vrbančic's dataset [110,111]	PhishTank, Alexa top sites	Balanced set: 59 K (31 K phishing), imbalanced set: 89 K (31 K phishing)	111 handcrafted features; URL string–based; automated extraction	Feature importance testing; traditional ML classifiers
Alvarado's dataset [112]	PhishTank, OpenPhish, PhishRepo, web scraping, public data	800 K URLs, 6 K SMS, 18 K emails, 30 K HTML	Text modality mix; designed for BERT-based models; supports modality balancing	Text-based phishing detection (BERT); multimodal learning
BitAbuse dataset [113,114]	bitcoinabuse.com	325 K sentences (27 K VP text)	Real-world phishing emails; manually restored labels; VP character set (706 types)	Adversarial robustness testing; pretraining for restoration models

6.1.3 Considerations in Utilization

Public phishing domain blacklists present several limitations, including detection delays, short data retention periods, and potential uncertainties and biases in community labeling, as well as discrepancies in crawling cycles. Such issues can distort the training and evaluation of AI-based detection models. Skula and Kvet analyzed a decade of phishing domains from PhishTank and PhishStats, finding that approximately 78% of domains were used only once, with only about 22% being reused [115]. This implies that blacklist-based detection is realistically effective only for reused domains, thereby limiting representativeness and generalizability. Oest et al. showed that blacklist responses to phishing websites employing evasion techniques can be delayed by up to three hours compared to simple phishing sites [116]. Taken together, these findings underscore that dataset-level structural biases and detection delays can undermine the validity of model performance evaluation.

6.2 Datasets for Illicit Gambling and Spam Sites

In addition to phishing, illicit gambling websites, SMS spam, and scam content play a significant role in web-based cybercrime. For training AI models to detect such threats, it is essential to secure domain-specific datasets. As shown in Table 9, this section summarizes representative datasets available for analyzing gambling and spam-related sites and provides a comparative analysis of their structures and applicability.

Min and Lee collected 11,172 URLs of illicit gambling websites through web mining, among which 978 sites were explicitly verified as illegal by established methods. Based on information such as URLs, WHOIS data, index pages, and landing page images, they designed 17 hybrid features to construct ML detection models. This dataset combined URL-based textual data, external metadata such as WHOIS, and real-time

renderable website structures and images, forming a hybrid structure that contributed to enhancing the precision of illicit gambling site detection [9].

Zhang et al. gathered POST messages from users of illicit gambling websites in real-world network environments. Using request bodies, cookies, and request line keywords as critical data, they applied Term Frequency–Inverse Document Frequency (TF-IDF) weighted Word2Vec embeddings and Stacked Denoising AutoEncoders (SDAE) for feature stabilization and dimensionality reduction. Subsequently, they classified website types via improved agglomerative clustering, and further refined user behaviors through integrated clustering methods combining K-means, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Ordering Points to Identify the Clustering Structure (OPTICS), and Gaussian Mixture Model (GMM) [117]. This dataset provides a structure highly suitable for research on illicit gambling detection based on actual user behavioral data.

Yang et al. collected 967,954 suspicious illicit gambling websites using large-scale crawling based on the Baidu search engine and an SVM-based detection system. These websites were analyzed through diverse criteria, including webpage structural similarity, Search Engine Optimization (SEO) abuse, exploitation of Internet infrastructure, third-party payment channels, and interconnections among gambling groups. The dataset was leveraged to identify 6581 operating groups or individuals and demonstrated strong capabilities in systematically revealing profit chains of illicit gambling infrastructures [118]. Building upon this line of infrastructure-oriented analysis, Gu et al. further modeled inter-app communication flows and behavioral similarities of mobile gambling applications via heterogeneous graph structures, thereby extending detection to organized gambling app ecosystems [119].

Table 9: Datasets and benchmarks for illicit gambling/spam/scam threat detection.

Dataset	Data Collection Scale	Content Structure	Key Characteristics	Use Cases
Min's dataset [9]	11 K (illicit gambling websites), 533 K (legitimate)	URL, WHOIS, HTML index, landing page images	Manual labeling via domain verification; 17 hybrid features	Illegal gambling site detection, hybrid feature analysis
Zhang's dataset [117]	Thousands of POST messages (estimated)	Request payloads, cookies	TF-IDF + Word2Vec; SDAE-based clustering	Behavioral pattern analysis, site clustering by scam type
Yang's dataset [118]	968 K (illicit gambling websites)	Webpage structure, SEO abuse, infrastructure links, payment system data	SVM detection; SEO + infra correlation	Profit chain analysis, detection of site clusters & networks
WebAA-datasets [120,121]	9 K (illicit websites), 5 K (legitimate)	HTML structure, embedded resource links, domain name similarity	BERT + resource dependency graph; org-level labels	Group-level linkage detection, malicious org clustering

(Continued)

Table 9 (continued)

Dataset	Data Collection Scale	Content Structure	Key Characteristics	Use Cases
DIFrauD benchmark [122,123]	96 K (across 7 domains)	Multidomain text: SMS, phishing email, fake news, job scams, etc.	Multidomain deception labels; domain-agnostic features	Cross-domain deception detection benchmarking

The WebAA datasets released by Zhang et al. comprise HTML-based data that include both illicit websites (e.g., gambling, pornography) and legitimate sites. Each website is labeled at the organizational level, enabling research on inter-domain link detection and organizational identification. The WebAA model evaluates organizational linkages by analyzing three types of similarities—embedded resource dependencies, HTML text, and domain names—making this dataset suitable for training and evaluating organization-level web detection models [120,121].

The DIFrauD benchmark comprises 95,854 text documents across seven domains, including political statements, fake news, job scams, product reviews, phishing emails, Twitter rumors, and SMS spam. Each sample is assigned a binary label of "deceptive" or "non-deceptive" and is designed for training and evaluating domain-independent detection models [122,123]. Its multidomain composition and clearly defined binary labeling system make it an appropriate benchmark for assessing the cross-domain adaptability of Transformer-based LLMs, as well as for training and evaluating detection models that operate effectively under heterogeneous domain conditions.

Datasets for gambling websites typically combine technical features such as URLs, HTML, and WHOIS with content-based labeling of user behavior, reflecting the volatility of real-world environments. By contrast, SMS and spam datasets are more suited to training text-analysis models, requiring tailored processing strategies that account for their structural differences from emails.

6.3 Limitations of Dataset Construction and Directions for Improvement

The performance of AI-based illicit website detection systems is heavily influenced by the quality of datasets, including factors such as structure, diversity, labeling methodology, and timeliness. However, existing phishing and malicious website detection datasets present several limitations, which inevitably restrict both accuracy and practical applicability if left unaddressed.

6.3.1 Key Limitations

- **Data imbalance and lack of representativeness:** Recent studies have highlighted that existing datasets suffer from class imbalance, brand-specific and English-centric bias, as well as static and outdated content structures. These issues limit the generalization ability of models and hinder the accurate detection of minority classes [15,18].
- **Asynchronous data collection:** Illicit websites often have short lifespans, ranging from a few hours to several days. If collection occurs too late, they may be mislabeled as benign domains or lead to annotation errors. Furthermore, many malicious websites employ techniques such as reCAPTCHA, 404-page spoofing, and JavaScript-based delayed loading to appear benign at the time of initial collection, making synchronized data acquisition even more challenging [90]. This, in turn, results in insufficient

coverage of content manipulated by LLMs, significantly reducing the ability to detect novel attack types. Experiments show that even simple LLM-based reconstructions can reduce the accuracy of traditional ML detection models by up to 9%, while some LLM-based models exhibit performance drops of around 4% [18].

- **Lack of feature diversity:** Many existing studies rely primarily on static, text-based features such as URLs, HTML, and email bodies. This vertical feature space is limited to the information explicitly exposed within a page, making it vulnerable to advanced evasion strategies. In practice, 8%–20% of phishing pages conceal malicious content through JavaScript rendering, which often causes HTML-based models to fail in detection [124].
- **Constraints in reproducibility and accessibility:** A considerable portion of prior research depends on private or restricted datasets, while publicly available datasets remain limited. This reliance undermines the generalizability and reproducibility of research results. In particular, datasets concerning illegal online gambling or pornography websites are scarce, closed, and often require extensive data elements, which further restrict accessibility. Additionally, reproducibility is exacerbated by the lack of standardized preprocessing frameworks, as approaches to labeling, URL normalization, and parsing vary widely among researchers [125].
- **Insufficient defense against adversarial attacks:** Recent attack strategies include visual perturbations (e.g., logo removal, font obfuscation) and LLM-based text rewriting designed to evade detection. Nonetheless, benchmark datasets that incorporate such modern attack trends remain scarce, and research on adversarial robustness in illicit website detection models is still at an early stage [18]. Only recently have visual noise-based datasets such as BitAbuse emerged to reflect such evasion attempts [113].

6.3.2 Directions for Improvement and Design Proposals

To establish high-quality datasets in the future, the following considerations are necessary:

- **Enhancing diversity and real-time acquisition:** A multi-source data collection framework should integrate not only URLs, HTML, and emails but also WHOIS, SSL/TLS, DNS, and log-based user behavioral data. In addition, research on real-time rendering-based collection, multi-source correlation analysis, and synchronization at the detection stage should be pursued.
- **Maintaining class balance:** Strategies such as adjusting phishing/non-phishing ratios, ensuring balance across countries and languages, and employing techniques like Synthetic Minority Over-sampling Technique (SMOTE) and GAN-based sample generation are required to mitigate imbalance issues.
- **Expanding horizontal feature space:** To achieve robust learning, models must be trained on a multi-modal feature space that incorporates heterogeneous attributes, including technical, network-level, and visual information. Beyond traditional static (vertical) features, a hybrid approach combining vertical, horizontal, and external trust-based features should be adopted.
- **Standardizing preprocessing and evaluation methods:** Labeling criteria, URL normalization procedures, and HTML parsing strategies should be explicitly defined. Providing open-source pipelines will strengthen benchmarking reproducibility.
- **Improving resilience against adversarial learning:** It is essential to curate datasets that reflect real-world evasion strategies, including visual and linguistic perturbations. The collection of GPT-generated phishing text and the design of adversarial scenarios through red-teaming should be conducted in parallel.

6.4 Evaluation Pitfalls and Best Practices

6.4.1 Evaluation Pitfalls

While benchmark datasets such as PhishTank, OpenPhish, and other publicly available repositories have significantly advanced AI-based detection research, reported performance metrics must be interpreted with caution. Several recurring evaluation pitfalls may artificially inflate detection performance and obscure real-world generalization limits:

- **Temporal leakage:** Many studies rely on randomly shuffled train–test splits, which may inadvertently include temporally adjacent domains or closely related attack campaigns in both training and testing sets. In rapidly evolving illicit web ecosystems, domains registered within similar time windows often share structural characteristics. Without time-based splitting (e.g., rolling window evaluation), models may learn campaign-specific artifacts rather than generalizable threat patterns.
- **Domain reuse across splits:** In domain-level detection tasks, identical or near-identical domains may appear in both training and test partitions under slightly different URLs or subdomain variants. This overlap artificially boosts accuracy while failing to evaluate true generalization to unseen infrastructures. Domain-disjoint evaluation strategies are therefore recommended whenever feasible.
- **Class imbalance and unrealistic priors:** Many benchmark datasets are artificially balanced to simplify training, whereas real-world traffic distributions are highly skewed toward benign instances. Models trained under balanced conditions may exhibit inflated recall and F1-scores but suffer from elevated false positive rates when deployed in operational environments. Reporting performance under imbalanced and realistic class priors is essential for deployment validity.
- **Inconsistent labeling sources and update frequency:** Public blacklists and threat intelligence feeds vary in update cadence, verification methodology, and labeling confidence. Models trained on such datasets may inherit labeling noise or reflect delayed detection artifacts. Explicit reporting of data collection timelines and verification sources enhances reproducibility and interpretability.
- **Overfitting to static benchmarks:** Extremely high accuracy scores—often exceeding 99%—may indicate overfitting to dataset-specific feature distributions rather than robust threat understanding. Cross-dataset validation, where models trained on one dataset are evaluated on another, provides a stronger indicator of zero-day resilience.

6.4.2 Recommended Best Practices

To improve empirical rigor and align evaluation with real-world deployment constraints, future research should consider:

- Time-based train–test splits to simulate evolving threat conditions.
- Domain-disjoint evaluation to prevent infrastructure overlap.
- Cross-dataset testing to assess generalization capacity.
- Reporting operational metrics beyond accuracy and F1-score, including FPR at fixed thresholds, detection latency, and performance stability over time.
- Explicit discussion of data collection windows, labeling confidence, and dataset limitations.

By addressing these evaluation concerns, AI-based detection research can move beyond benchmark optimization toward operationally robust and zero-day–resilient defense systems. This analysis directly responds to RQ2 and strengthens the empirical grounding.

7 Limitations, Challenges, and Open Problems

7.1 Model Evasion Strategies

Although illicit website detection systems can demonstrate high accuracy, their robustness may significantly deteriorate when attackers deliberately employ strategies to bypass them. In particular, ML–based detectors are sensitive to adversarial perturbations, and under real-time attack scenarios, there is a risk that phishing sites evade detection and are exposed to users. Systematic analyses have emphasized that generalization to unseen data remains one of the core unresolved challenges in zero-day detection [126]. This section analyzes several representative cases of model evasion strategies, one of the most challenging areas in illicit website detection.

Mutation-based strategies that preserve the form of the URL while evading detection have proven highly effective in practice. Song et al. reported a 100% evasion success rate against Google Phishing Page Filter across white-box, gray-box, and black-box scenarios, and up to 81.25% against BitDefender TrafficLight. Their attack manipulated certain elements of malicious websites while maintaining the integrity of the DOM structure and functionality [127]. Shirazi et al. demonstrated that even by manipulating a single feature, the detection rate of phishing detection systems dropped to 70%, and with four features manipulated, the rate fell to 0% [128]. This result highlights that ML–based phishing detection models rely excessively on a small set of critical features, and such feature-level vulnerabilities may serve as practical evasion vectors for attackers.

PhishOracle is a framework that generates adversarial phishing pages by embedding various content-based and visual-based phishing elements into legitimate webpages. Using Google Gemini 1.5 Flash as the underlying LLM, this system evaluated the robustness of existing detectors. While CNN-based detectors (e.g., VisualPhishNet, Phishpedia) failed to properly identify most adversarial pages, the LLM-based detector maintained comparatively strong performance, with an F1-score of approximately 88.59% and a recall of about 79.51% [129]. These findings both raise awareness of novel attack strategies and suggest the promising potential of LLMs as next-generation phishing detection models.

Lee et al. investigated logo-based phishing detection models, including Siamese networks, and proposed an attack leveraging Generative Adversarial Perturbations (GAP). As illustrated in Fig. 12, their method introduced imperceptible visual perturbations into logos, resulting in an evasion rate of up to 95%. Moreover, a user study confirmed that most participants could not distinguish between the original and perturbed logos [130]. This demonstrates that perturbations nearly invisible to the human eye may also serve as effective attack surfaces against vision-based detectors.

Beyond explicit adversarial manipulation strategies, evasion may also arise when models fail to generalize to unseen attack distributions. In this sense, the inability to handle previously unseen data can itself be viewed as a structural form of model evasion. Dai et al. proposed a hybrid intrusion detection framework integrating an autoencoder-based anomaly detector with RF and XGBoost classifiers, explicitly evaluating performance on previously unseen data [131]. Their results demonstrated that incorporating reconstruction-error-based anomaly filtering significantly improved robustness, achieving near-perfect performance even when tested on unseen attack samples. This empirical evidence reinforces the importance of anomaly-aware architectures for addressing the limitations of purely signature-driven or supervised models, which often struggle to generalize beyond known attack distributions.

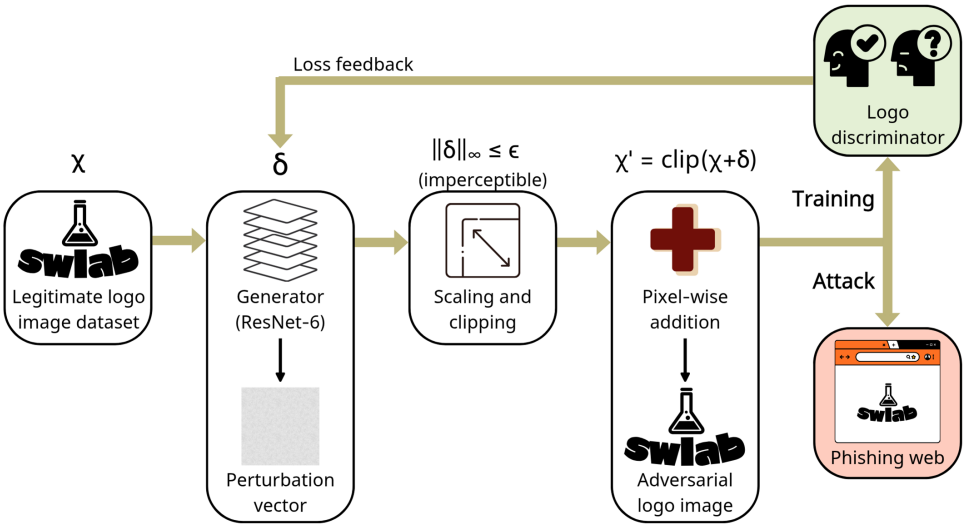

Figure 12: GAP-based adversarial logo generation and attack workflow. A generator learns a imperceptible perturbation under a magnitude constraint. After scaling and clipping, the perturbation is added pixel by pixel to a legitimate logo to produce an adversarial logo. During training, a frozen logo discriminator provides loss feedback; at attack time, the adversarial logo evades detection while remaining visually subtle [130]. **Takeaway:** Imperceptible visual perturbations can significantly degrade logo-based detectors, highlighting the fragility of vision-only models under adversarial manipulation.

7.2 Multilingual and Cross-National Challenges

The modern web is inherently multilingual, and phishing, scam, and illegal gambling websites also operate across diverse language environments. However, most detection models for illicit websites have been trained on English-centric datasets, resulting in a significant degradation of generalization performance when applied to multinational or non-English content. This section discusses the limitations of detection models arising from linguistic diversity, the corresponding countermeasures, and future research directions.

Previous studies have predominantly relied on large-scale English corpora such as Enron Spam and PhishMail. In contrast, data collection and open research for non-English languages remain extremely limited, with some datasets constructed merely by machine-translating English text. Consequently, experiments in multilingual contexts have been considerably constrained, failing to adequately capture linguistic and cultural characteristics. This limitation has frequently led to false positives, false negatives, and learning inefficiencies caused by data bias [132].

Staples conducted zero-shot learning experiments using multilingual Transformer models such as XLM-RoBERTa on English, French, and Russian datasets. In monolingual detection tasks, the model achieved 99% accuracy on English and French test sets and 95% on Russian, demonstrating strong performance across the three languages. However, when trained on French and tested on English, accuracy dropped to 72%, nearing the baseline, and training on Russian followed by testing on English also revealed a substantial decline in transferability. These results highlight the generalization limitations caused by grammatical and lexical discrepancies between languages. In contrast, when the model was trained simultaneously on French and Russian and then tested on English, it achieved 99% accuracy, demonstrating that multilingual training can lead to dramatic improvements in certain transfer scenarios [132].

To improve multilingual detection, the following issues must be considered:

- **Alleviating data scarcity and class imbalance:** A critical challenge is the lack of training data in non-English languages, coupled with imbalanced labels, which significantly degrade generalization performance.
- **Overcoming the limitations of translation-based learning:** Translated sentences often fail to capture the socio-cultural context and nuances of local expressions, highlighting the necessity of collecting native-language corpora.
- **Introducing few-shot learning and meta-learning:** Research is urgently required on learning structures that enable detection in specific languages with only a small number of examples.
- **Expanding cross-lingual embedding-based detection models:** The applicability of the latest multilingual LLMs—such as GPT, BLOOM, Gemma, PolyLM, Aya, and PANGEA—should be further explored. Robust strategies for cross-lingual transfer learning are required to enhance performance [133–137]. Fig. 13 illustrates the composition of the PANGEAINS corpus and its strategy for multilingual balance.

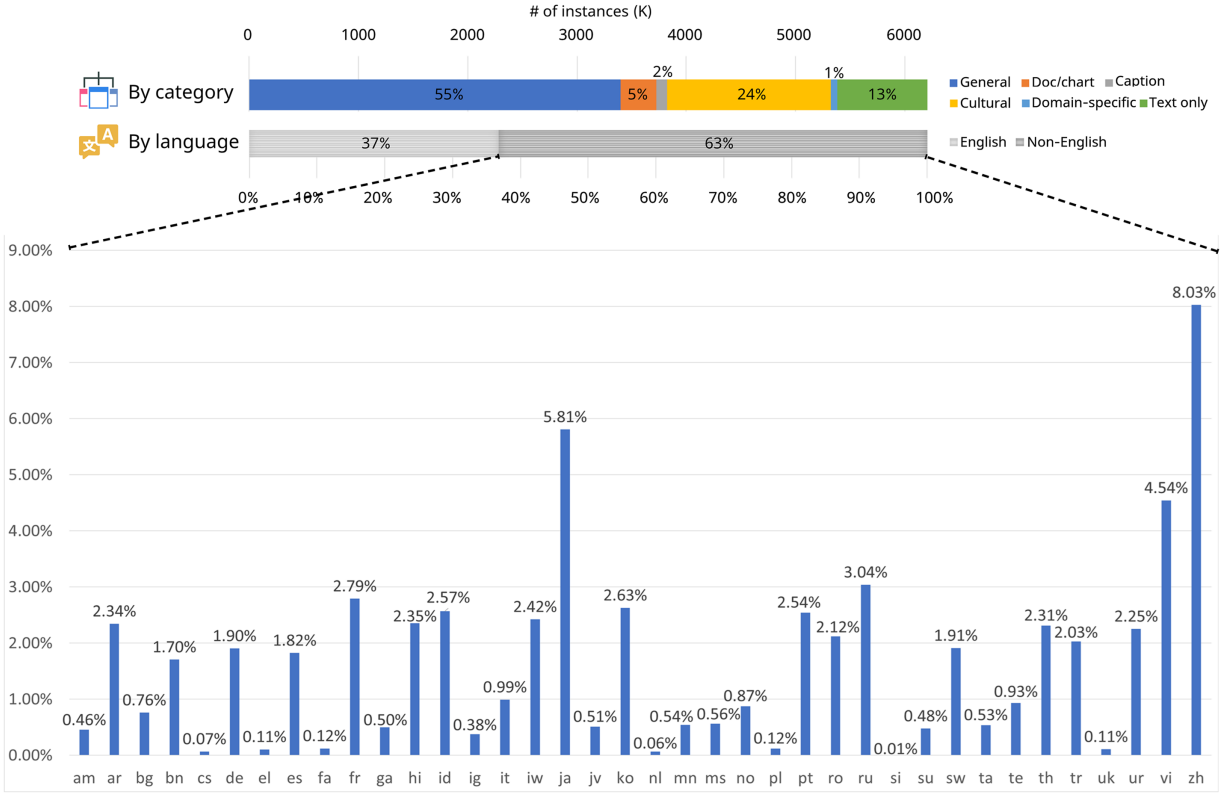

Figure 13: Composition of a 6.2 M multilingual multimodal instruction corpus (PANGEAINS). The top bar groups instances by task category: General—open-domain visual instructions, multi-turn conversations, and visual reasoning; Doc/chart—question answering over documents, tables, and charts (incl. OCR); Caption—image captioning; Domain-specific—specialized instructions (e.g., medical); Cultural—culturally identifiable images and prompts that capture region-specific context; Text-only—language-only instructions for general tasks, code, and math. The middle bar shows the 37% English vs. 63% multilingual split; the bottom bars report per-language proportions (total 39 languages) [137]. **Takeaway:** Multilingual and multimodal pretraining resources are critical for improving cross-lingual generalization in illicit web detection systems.

7.3 Practical Deployment Challenges

Even when AI-based illicit website detection models achieve high accuracy, factors such as latency and the FPR can become critical issues in real-world deployment. These aspects directly influence the practicality of detection systems, user trust, and the efficiency of security operations; therefore, it is necessary to evaluate not only accuracy but also these operational indicators in a comprehensive manner.

DL–based detectors typically involve processes such as input crawling, HTML rendering, and screenshot analysis, which may result in response delays of up to several tens of seconds in real-world environments. For example, in the RBPD system, the web crawler required up to 10 s, and the RBPD component an additional 10 s, leading to a total response time considerably longer than the average of around 2 s expected by users. Such latency not only degrades the user experience in real-time email filtering or web detection systems but also causes significant resource consumption when processing large-scale requests. To address this issue, Li et al. proposed the PhishIntel system, which adopts a fast–slow task architecture. In this design, the fast path provides immediate responses using caches and blacklists, while the slow path performs deep analysis through the RBPD model, thereby significantly improving overall system response time while maintaining zero-day detection capability [91].

On the other hand, a high FPR leads to the misclassification of benign websites as malicious, blocking users from accessing legitimate pages. This results in user inconvenience and erosion of trust in the system. Conversely, a high False Negative Rate (FNR) allows actual malicious sites to go undetected, posing direct risks to users or organizations. Dalvi et al. demonstrated that, for a neural network–based phishing detection model, the trade-off between precision and missed detections could be controlled by tuning the decision threshold. Specifically, by setting the threshold to 0.99, the model achieved an FPR of 0.0003 and an FNR of 0.0198 while maintaining an F1-score of 0.98 [138].

Similarly, Puccetti and Ceccarelli emphasized that relying solely on static metrics such as the F1-score is impractical in real-world contexts. They argued that the timeliness of detection—defined as detection latency—is a crucial measure, as the ability to detect attacks or errors quickly is essential. However, lowering thresholds to reduce latency inevitably increases the FPR, leading to alert fatigue. Thus, in real deployment, a balanced adjustment between FPR and latency is indispensable [139]. In summary, simple reliance on F1-score is insufficient; instead, policy-driven threshold tuning strategies must be applied in accordance with the security sensitivity of the deployment environment.

In addition, practical detection systems require latency-aware design that accounts for factors such as the number of URLs to be processed, frequency of detection requests, and the availability of server and network resources. Moreover, tolerance for false positives varies across user groups, suggesting that threshold values should be differentiated between security-critical environments (e.g., enterprise networks) and convenience-oriented environments (e.g., individual users). Finally, aspects such as cache update intervals, queue processing times for pending detections, and fallback detection strategies in case of detection failures should also be carefully considered to ensure robustness and effectiveness in operational settings.

7.4 Propose a Deployment-Oriented Evaluation Checklist

While numerous AI-based detection models report high benchmark accuracy, real-world deployment introduces constraints that are rarely standardized across studies. Beyond dataset performance, operational evaluation must consider latency, false positive tolerance, update cadence, and privacy limitations.

To bridge the gap between experimental results and practical adoption, we propose a deployment-oriented evaluation checklist that researchers and practitioners may adopt when assessing illicit web detection systems.

By incorporating these deployment considerations into evaluation protocols, AI-based detection research can better align with operational realities. This checklist addresses RQ3 by clarifying how architectural design, latency constraints, and institutional requirements shape the feasibility of illicit web defense systems.

7.4.1 Latency Budget by Use Case

Different operational environments impose distinct response-time constraints:

- Browser-level warnings: <1 s.
- Email gateway filtering: 1–2 s.
- Security operations center triage: batch processing permissible.
- Infrastructure-level graph analysis: periodic (hourly/daily) updates acceptable.

Models requiring webpage rendering, screenshot capture, or large-scale graph construction should explicitly report processing overhead.

7.4.2 Acceptable FPR

Tolerance for false positives varies by deployment context:

- Enterprise environments: FPR < 0.1% often required.
- Consumer-facing applications: Slightly higher tolerance may be acceptable.
- Infrastructure blocking (ISP/firewall): Extremely low FPR critical to avoid service disruption.

Studies should report FPR at fixed decision thresholds rather than relying solely on aggregate F1-score.

7.4.3 Update Cadence and Concept Drift Monitoring

Illicit web ecosystems evolve rapidly. Effective systems must specify:

- Model retraining frequency (daily/weekly/event-driven).
- Mechanisms for drift detection.
- Data refresh pipeline and blacklist update integration.

Failure to address update cadence may render high-performing models ineffective within weeks of deployment.

7.4.4 Privacy and Data Collection Constraints

Operational feasibility depends on what data can realistically be collected:

- Is HTML scraping permissible?
- Are full-page screenshots allowed under privacy regulations?
- Can DNS logs, Open-Source Intelligence (OSINT), or infrastructure metadata be aggregated?
- Are user-level telemetry signals legally usable?

Deployment viability is directly linked to regulatory compliance and infrastructure access.

7.4.5 Resource and Infrastructure Requirements

Studies should explicitly report:

- GPU/CPU requirements.

- Memory footprint.
- Dependency on cloud-based inference.
- Scalability under high traffic volumes.

Resource-intensive transformer or multimodal architectures must justify their added complexity relative to lightweight alternatives.

8 Future Directions

8.1 GAN-Based Threat Simulation for Illicit Sites

GAN have emerged as powerful simulation tools capable of generating attack scenarios that closely resemble real-world conditions for detector training. By leveraging GANs, researchers can overcome the inherent limitations of conventional training datasets, thereby enhancing zero-day detection performance, mitigating data imbalance, and improving the generalization ability of models. This section reviews GAN-based approaches for illicit site generation and detector training frameworks, and discusses potential future directions.

Sasi and Balakrishnan proposed a GAN architecture for phishing URL detection that employs a Variational Autoencoder (VAE) as the generator and a self-attention-based Transformer model as the discriminator. The VAE encodes input URLs into a low-dimensional latent space and reconstructs them through a decoder, minimizing reconstruction loss to generate refined URL sequences. In contrast, the Transformer discriminator incorporates eight parallel multi-head attention layers, normalization, and pooling operations to capture semantic patterns in URLs, followed by a sigmoid layer for binary classification. Through integrated adversarial training, the two modules iteratively optimize one another, leading to improved classification performance. Experimental results on a dataset of one million URLs demonstrated an accuracy of 97.75%, outperforming conventional DL models [140]. These findings indicate that the self-attentive architecture combining VAE and Transformer effectively captures contextual URL patterns.

Pham et al. introduced a Wasserstein GAN with Gradient Penalty (WGAN-GP) framework to generate realistic phishing URLs and integrate them into existing training datasets, thereby evaluating the performance of LSTM- and GRU-based classifiers. Unlike traditional GANs, WGAN-GP implements a critic rather than a probabilistic discriminator, aiming to minimize the Wasserstein distance for enhanced training stability. Instead of weight clipping, it applies gradient penalty to satisfy the Lipschitz condition, while the generator utilizes 1D convolutions and residual blocks optimized for textual data generation. Experimental results revealed that, particularly in data-scarce environments, augmenting training sets with WGAN-GP-generated phishing URLs significantly improved the accuracy of LSTM classifiers [141]. This highlights the effectiveness of WGAN-GP for alleviating class imbalance and strengthening minority-class representations.

Albahadili et al. employed GANs to synthesize additional URL samples to mitigate class imbalance, followed by character-level encoding using a CNN for feature extraction. Critical features were then selected via the White Shark Optimizer (WSO), a swarm intelligence algorithm, before being applied to an LSTM-based detection model. Evaluations using the ISCX-URL-2016 and PhishTank datasets achieved detection accuracies of 97.94% and 96.78%, respectively [142]. This integrated approach demonstrates that combining GAN-based data augmentation with optimization-driven feature selection can simultaneously address imbalance issues and accelerate training efficiency.

Sern et al. proposed PhishGAN, a GAN-based model designed to automatically generate homoglyph—visually similar—domain images. This system simulates punycode-based visual similarity attacks by rendering domain strings into images under diverse fonts and noise conditions, which are then used as GAN inputs to produce visually deceptive domain variants. The generated images serve as training data for

a homoglyph identifier model, which employs a Triplet Loss function to classify attack URLs mimicking legitimate domains [143]. This approach effectively mitigates font bias in traditional models and provides a scalable solution for adapting to novel visual obfuscation attacks.

Despite these advances, current GAN-based illicit site detection research still lacks standardized metrics to evaluate the structural and statistical similarity between generated URLs and those created by real attackers, which limits the reliability of model validation. Moreover, most generative models remain focused on simple substitution attacks, insufficiently capturing the diversity of adversarial tactics. Future directions should therefore include scenario-driven threat modeling that integrates adversaries' TTPs, as well as the development of lightweight GAN architectures suitable for real-time deployment or hybrid frameworks combining GANs with LLM-based simulators. Ultimately, GAN-based simulation must evolve beyond mere data augmentation toward becoming a core component for joint optimization of threat modeling and detection systems.

8.2 Few-Shot and Self-Supervised Learning for Low-Resource Scenarios

In AI-based illicit website detection environments, issues such as spear-phishing, emerging spam variants, and region-specific illegal gambling or advertising sites frequently lead to situations where the number of samples is small or class diversity is limited. Under such data-scarce conditions, few-shot learning and self-supervised learning represent promising alternatives that can enhance the generalization performance of detectors and improve the ability to capture zero-day attacks.

Li and Cheng proposed a few-shot learning–based spear-phishing detection model that combines discrete features from email headers with word embedding vectors from message bodies. The message content is embedded using a word2vec model and then processed through a simple architecture that integrates hierarchical max/min pooling with global average pooling. These vectors are concatenated with basic email features and input into ML classifiers. Experimental results showed that the proposed approach achieved superior accuracy, precision, and recall compared with existing supervised detection methods such as CNN, LSTM, and KNN graph models, particularly under low-sample conditions such as 1-shot and 5-shot learning [144]. This indicates that few-shot learning strategies are highly effective for detecting rare attack types with very limited samples.

In another study, Lu et al. experimentally demonstrated that the UniSiam model, pre-trained in a self-supervised manner without labels for the base dataset, achieved stronger generalization than supervised approaches in few-shot classification tasks, including cross-domain scenarios. The proposed UniSiam framework follows the InfoMax principle, maximizing mutual information through alignment and uniformity between augmented views. To address the bias inherent in conventional InfoNCE estimators used in contrastive learning, the method employs a low-bias Mutual Information Neural Estimator (MINE) combined with asymmetric alignment. The architecture consists of a ResNet backbone with Multi Layer Perceptron (MLP)-based projection and prediction heads; asymmetric alignment is applied during training, while only a simple linear classifier is used in the fine-tuning stage. In addition, strong data augmentation is employed to increase representation diversity and mitigate dimensional collapse [145]. Considering that large-scale unlabeled data is generally easier to obtain in practice, this result suggests that self-supervised learning can serve as a highly effective strategy for the foundational stage of detector representation learning.

Zhang et al. proposed a multi-task framework that leverages self-supervision as an auxiliary task to improve few-shot object detection for novel classes. Their method is based on Meta-DETR, a few-shot object detection model that integrates meta-learning into the Detection Transformer (DETR) architecture, augmented with a self-supervised branch that shares weights with the feature extractor and transformer. Specifically, a denoising module using contrastive denoising generates positive/negative sample pairs to

strengthen the model's discriminative capability, while a team module enforces location-based constraints so that each query prediction converges toward predefined regions, thereby improving localization accuracy. Experimental results demonstrated consistent improvements over the baseline Meta-DETR across various splits of the PASCAL VOC and MS COCO datasets, with ablation studies confirming that the self-supervised branch alone contributed to a 1.1% improvement in 1-shot detection performance [146]. Such a structure is particularly well suited for real-world security scenarios facing label scarcity—such as illicit website detection—where rapid response is critical. For instance, newly emerging phishing sites or malicious domains often lack sufficient training data, and this architecture can generalize object representations and enhance localization precision through self-supervision, thereby contributing as a foundational technique for advanced detection systems.

Yeboah et al. proposed an ensemble approach for detecting phishing emails and web attacks that combines two complementary self-supervised pre-training methods: transformation prediction for structural representation learning, and masked prediction for contextual representation learning. After applying these two methods in parallel to HTTP request logs and email body texts, the extracted features are ensembled to form a unified representation for downstream detection. Experimental evaluations showed that the ensemble self-supervised learning model outperformed individual baselines across all performance metrics: accuracy, precision, recall, and F1-score. Notably, in imbalanced datasets, the model achieved a phishing detection recall of 0.9904 and an F1-score of 0.9844 [147]. Given that phishing campaigns and malicious web requests (e.g., Structured Query Language (SQL) injection, Cross-Site Scripting (XSS)) are central vectors of modern cyber threats, this combined structural–contextual learning approach highlights the potential to detect novel, undefined attack variants. This direction suggests the possibility of developing privacy-preserving, highly generalizable detection systems capable of capturing and responding rapidly to new forms of malicious activity in security logs and web traffic.

8.3 Explainable AI and Legal Integration

Despite the high accuracy of DL–based detection systems, their opaque "black-box" nature limits both user trust and the attribution of legal responsibility. Consequently, XAI has gained prominence as a means to enhance the reliability of detection systems and to comply with legal requirements.

XAI provides human-understandable explanations and justifications for the outputs of AI-driven cybersecurity systems, thereby supporting not only security operators but also end users and regulatory authorities in making more trustworthy decisions. When applied to tasks such as illicit website identification or abnormal behavior detection, XAI improves transparency compared with conventional black-box models, reduces false positives in threat identification, and explicitly clarifies decision-making grounds. Such features also strengthen the legitimacy of digital forensics and post-incident responses [148,149]. These characteristics suggest that XAI is not merely a technical add-on, but rather a core component of trustworthy security system operations.

In the legal domain, applicable XAI technologies are evolving along three axes: (i) intrinsically interpretable models (e.g., linear regression, decision trees), (ii) post-hoc explanation methods (e.g., Local Interpretable Model-agnostic Explanations (LIME), Shapley Additive Explanations (SHAP)), and (iii) example-based approaches (e.g., counterfactual scenarios, prototypes and exceptions). These developments intersect with diverse legal reasoning paradigms—rule-based, argument-based, case-based, evidential, and hybrid frameworks—expanding the potential application of XAI. A persistent challenge is the trade-off between performance and explainability: traditional models offer interpretability but limited accuracy, whereas DL models achieve high accuracy but poor transparency. Therefore, in legal contexts, explanations

must go beyond the technical level of "why was this computed?" to the normative level of "why is this decision legally justified?", grounded in precedents, regulations, and evidence [150,151].

In cyber forensics, XAI supports reproducibility and traceability of AI decision-making through feature importance analysis, surrogate modeling, counterfactual explanations, and visualization techniques (e.g., saliency or heat maps). Unlike black-box models that present only outputs, XAI explicitly documents and visualizes the associations between inputs and outputs, the model's internal decision rules, and the relevant feature sets. These outputs can then serve as verifiable evidentiary grounds in court proceedings [152]. Extending this logic to illicit website detection, XAI enables forensic systems to explain detection outcomes not merely as binary results but through concrete feature contributions and decision pathways, thus establishing admissible forensic evidence systems and providing strong proof for regulatory or legal proceedings.

An illustrative example is the EXPLICATE framework (Fig. 14), which integrates a ML–based phishing detector, XAI methods (LIME and SHAP), and a LLM (DeepSeek v3) into a multi-layered architecture. LIME explains model decisions at the word level, while SHAP quantifies the contributions of higher-level features, thereby offering complementary interpretations. Building on these XAI outputs, DeepSeek v3 converts technical explanations into natural-language narratives accessible to general users, thereby enhancing interpretability and communicability. Experimental results demonstrated 98.4% detection accuracy, 94.2% explanation accuracy, and 96.8% explanation consistency [153]. These results underscore the practical significance of combining high detection accuracy with robust explainability, advancing user trust in illicit website and phishing detection systems, improving real-time responsiveness, and broadening applicability in digital forensics.

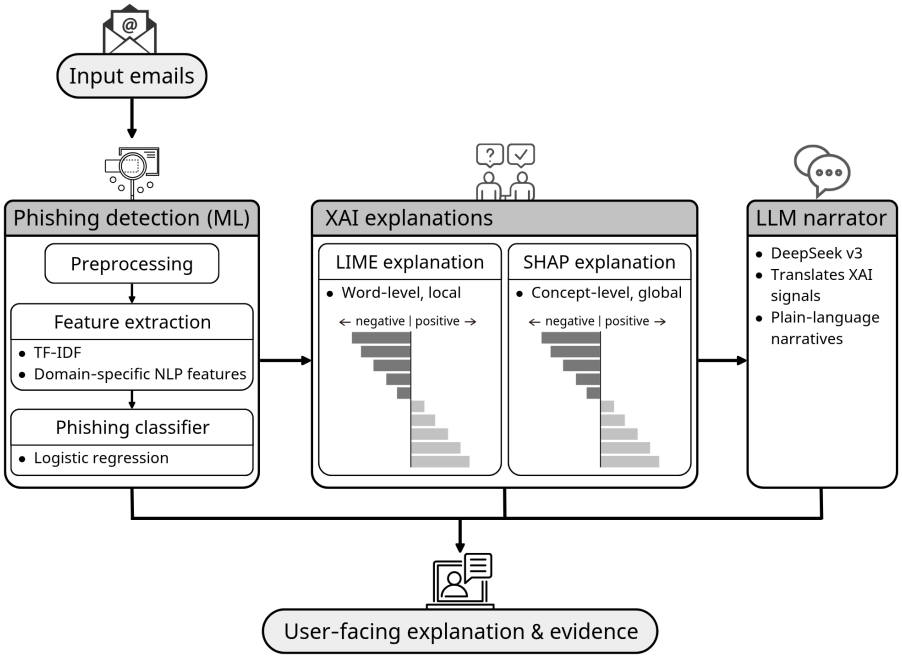

Figure 14: Dual-explanation architecture for phishing detection (EXPLICATE). Input emails are preprocessed and vectorized (TF-IDF with domain-specific NLP features) and classified by logistic regression. The XAI layer provides complementary explanations: LIME (word-level) and SHAP (concept-level). An LLM narrator (DeepSeek v3) translates these signals into plain-language narratives, aggregating into a user-facing explanation and evidence package [153]. **Takeaway:** Integrating post-hoc explainability with LLM-based narrative generation bridges the gap between high detection accuracy and legally meaningful interpretability.

Nevertheless, explanations provided by XAI are not always accurate or interpretable, which may pose serious challenges in assigning legal responsibility. Some studies have warned that XAI may even obscure accountability and be misused as a means of agency laundering, shifting responsibility onto users or vulnerable populations [154]. Thus, establishing realistic reliability criteria and verification standards for explanations is essential. Moreover, the lack of systematic performance comparison studies and standardized evaluation frameworks across XAI techniques remains a major barrier to practical deployment.

For instance, Calzarossa et al. conducted comparative experiments on traditional learning models (e.g., LR, RF, and SVM) to examine how explainability can be quantified and assessed. Their framework employs built-in explanation indicators (e.g., feature importance) but extends beyond single-instance values by repeatedly applying bootstrap sampling to validate the stability and reliability of explanations. In addition, they introduced the Lorenz Zonoid measure to quantify the concentration and parsimony of explanations. This approach enables the quantitative comparison of explanation robustness and simplicity while retaining computational efficiency, in contrast to more resource-intensive post-hoc methods such as SHAP. In practice, applying their framework to a RF model yielded explainability results within approximately four minutes, whereas SHAP required about 2 h and 19 min [155]. Such evaluation and comparison schemes, which jointly consider computational efficiency and reliability, can significantly improve the operational applicability of XAI to cybersecurity domains, including illicit website detection.

8.4 Predictive AI Encompassing the Entire Domain Ecosystem

Traditional domain-based malicious detection systems have primarily focused on reactive responses, designed to address single or short-term risks. However, attackers are increasingly organizing sophisticated domain ecosystems (e.g., registration, transfer, replication, and domain group operations), making it difficult to preemptively block threats through simple detection alone. Consequently, predictive AI strategies that forecast the overall flow of domain networks to identify potential threats in advance are gaining attention as next-generation defense measures.

Predictive security analyzes diverse data—including past attack records, user and device behaviors, and system and network logs—through AI techniques to identify anomalies and potential future threats in advance. By leveraging supervised and unsupervised learning as well as anomaly detection, these systems capture early signs of changes in domain clusters, the emergence of malicious domains, unknown attacks, insider threats, and lateral movement. Moreover, when combined with threat intelligence (e.g., Indicator of Compromise (IOC), IP addresses, domain names, OSINT, threat feeds), they enable automated alert correlation and risk scoring. The models are refined through iterative learning and retraining, and further improved via error analysis to enhance accuracy [156]. Such architectures transform conventional reactionary detection systems into intelligence-led, proactive defense frameworks, thereby substantially enhancing both defensive capability and response speed.

Group-IB's predictive AI platform exemplifies this approach by integrating diverse data sources, including internal telemetry (e.g., network and system logs, Extended Detection and Response (XDR) data) and external threat intelligence (e.g., threat intelligence feeds, Digital Risk Protection (DRP), Attack Surface Management (ASM)). Through clustering, classification, and correlation analysis, the platform identifies historical patterns and forecasts the probabilistic likelihood of future threats [157].

Darryl's proposed multi-source predictive framework incorporates signals collected from social media platforms such as Twitter (now X), Reddit, and Telegram, as well as from dark web forums, marketplaces, and paste sites. After domain-specific preprocessing and embedding, these signals are aligned along a unified timeline, followed by LSTM-based time-series learning enhanced with attention mechanisms, anomaly

detection, and confidence scoring to generate early warnings of future threats. This process captures pre-propagation indicators such as keyword/hashtag clustering, sentiment trajectories, burst detection, sudden spikes in dark web mentions, and changes in actor networks. The framework then presents predicted attack types, target categories, associated timelines, and confidence levels through a real-time dashboard [158]. By integrating early signals from both social media and the dark web, this framework offers the capability to anticipate the creation and proliferation of illicit websites, thus advancing the shift from reactive blocking toward proactive detection and response systems.

Almahmoud et al. constructed a dataset comprising 42 types of cyberattacks and 98 Pertinent Alleviation Technologies (PATs), combining large-scale data sources such as news, blogs, government advisories, Twitter feeds, and Elsevier API-based research articles. Based on this dataset, they modeled the relationship between threats and mitigation technologies as a Threats and Pertinent Technologies (TPT) graph, and employed a GPT-based extraction algorithm to semi-automatically map relevant mitigation technologies to each attack type. They subsequently developed a Bayesian variant of a Multivariate Time-series GNN (B-MTGNN) to forecast the gaps between attack trends and technology advancements over a three-year horizon, achieving higher accuracy compared to traditional time-series analysis. The experimental results indicated that gaps between threats and countermeasures will continue to widen in domains such as malware, ransomware, adversarial attacks, and vulnerability exploits, thereby underscoring the need for further investment in detection technologies, integrity monitoring, Security Information and Event Management (SIEM), and automated response [159]. This research framework moves beyond fragmented threat prediction, establishing a macro-level analytical structure that encompasses the entire cyber domain ecosystem and offers strategic insights into combating cyber threats, including illicit websites.

9 Conclusion

This survey has examined the evolution of AI-based detection strategies for illicit web ecosystems from a modality, infrastructure, and deployment-oriented perspective. Rather than treating phishing, illegal gambling, scams, and malvertising as isolated phenomena, the reviewed literature reveals that these threats operate within interconnected and rapidly mutating web infrastructures. Consequently, resilient detection cannot be achieved through any single model or feature category. Robustness instead emerges from the interaction between lexical analysis, visual inspection, structural graph modeling, and contextual sequence reasoning across the domain lifecycle.

A key observation from the synthesized studies is that reported performance gains often depend less on architectural novelty and more on dataset construction, evaluation rigor, and operational calibration. Issues such as temporal leakage, domain overlap, class imbalance, and multilingual bias significantly influence measured accuracy and may obscure real-world generalization limits. In this regard, the zero-day challenge should not be viewed merely as the detection of previously unseen samples, but as a systemic problem involving distribution shift, infrastructure mutation, and adversarial adaptation. Addressing this challenge requires not only improved models, but also evaluation protocols that better simulate evolving threat conditions.

At the system level, practical deployment constraints—such as latency budgets, false positive tolerance, update cadence, and privacy restrictions—play a decisive role in determining feasibility. The reviewed architectures demonstrate that detection effectiveness must be balanced against resource consumption and user trust considerations. Fast–slow pipelines, hybrid modality fusion, and infrastructure-level graph analysis represent promising design patterns for aligning detection capability with operational realities.

Despite its comprehensive scope, this review has several limitations. First, although we adopted a structured literature exploration strategy, the rapidly evolving nature of illicit web threats and AI techniques

implies that newly emerging studies may not have been captured. Second, many of the surveyed works rely on publicly available benchmark datasets, which may not fully reflect real-world traffic distributions, multilingual diversity, or adversarial adaptation dynamics. These limitations highlight the importance of continuous updating and cross-domain validation in future survey efforts.

Future research directions increasingly converge toward generative threat simulation, low-resource and self-supervised learning, explainable AI integration, and predictive ecosystem modeling. These trajectories signal a broader paradigm shift from reactive URL classification toward anticipatory, intelligence-led defense frameworks capable of modeling the structural evolution of illicit web domains. Importantly, sustainable mitigation of illicit web threats will depend on the coordinated integration of technical innovation, legal frameworks, institutional response mechanisms, and cross-sector collaboration.

Acknowledgement: Not applicable.

Funding Statement: This research was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea goverment (MSIT) (No. RS-2024-00439139, Development of a Cyber Crisis Response and Resilience Test Evaluation Systems). And this research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Graduate School of Virtual Convergence support program (IITP-2026-RS-2023-00254129) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). And this research was supported by the "Regional Innovation System & Education (RISE)" through the Seoul RISE Center, funded by the Ministry of Education (MOE) and the Seoul Metropolitan Government (2026-RISE-01-018-05). And this research was supported by QuadMiners Corp.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization and design, Jaeho Hwang and Moohong Min; research and analysis of existing methods and literature findings, Jaeho Hwang; data curation, Jaeho Hwang; writing—original draft preparation, Jaeho Hwang; writing—review and editing, Moohong Min; visualization, Jaeho Hwang; funding acquisition, Moohong Min. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: This paper is predominantly a review that synthesizes existing methods and literature findings. This investigation utilized only data obtained from publicly accessible sources. These datasets are accessible via the sources listed in the References section of this paper. As the data originates from publicly accessible repositories, its accessibility is unrestricted.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript

AFP	Australian Federal Police
AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
ASM	Attack Surface Management
ASN	Autonomous System Number
AUC	Area Under the Curve
BEC	Business Email Compromise
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
B-MTGNN	Bayesian Variant of a Multivariate Time-Series Graph Neural Network

C2 server	Command and Control Server
CEDD	Color and Edge Directivity Descriptor
CERT	Computer Emergency Response Team
CLD	Color Layout Descriptor
CNAME	Canonical NAME
CNN	Convolutional Neural Network
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DETR	DEtection TRansformer
DL	Deep Learning
DNS	Domain Name System
DOM	Document Object Model
DRP	Digital Risk Protection
DT	Decision Tree
EDL	External Dynamic List
FBI IC3	Federal Bureau of Investigation's Internet Crime Complaint Center
FNR	False Negative Rate
FPR	False Positive Rate
GAN	Generative Adversarial Network
GAP	Generative Adversarial Perturbations
GAT	Graph ATtention Network
GCN	Graph Convolutional Network
GCN-EW	Graph Convolutional Network with Edge Weights
GMM	Gaussian Mixture Model
GNN	Graph Neural Network
GPT	Generative Pre-Trained Transformer
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
HCG	Host-Connection Graph
HTML	HyperText Markup Language
HTTPS	HyperText Transfer Protocol Secure
IDS	Intrusion Detection System
IOC	Indicator Of Compromise
IP	Internet Protocol
IPS	Intrusion Prevention Systems
ISP	Internet Service Provider
JSON	JavaScript Object Notation
KNN	K-Nearest Neighbors
LBP	Loopy Belief Propagation
LIME	Local Interpretable Model-Agnostic Explanations
LLM	Large Language Model
LR	Logistic Regression
LSTM	Long Short-Term Memory
MINE	Mutual Information Neural Estimator
ML	Machine Learning
MLP	Multi Layer Perceptron
MPEG	Moving Picture Experts Group
NB	Naive Bayes
NLP	Natural Language Processing
OCR	Optical Character Recognition

OFS-NN	Optimal Feature Selection and Neural Network
OPTICS	Ordering Points To Identify the Clustering Structure
OSINT	Open-Source INTelligence
PATs	Pertinent Alleviation Technologies
PCA	Principal Component Analysis
RBPD	Reference-Based Phishing Detection
RDAP	Registration Data Access Protocol
RF	Random Forest
RL	Reinforcement Learning
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
RQ	Research Question
SCD	Scalable Color Descriptor
SDAE	Stacked Denoising AutoEncoders
SEO	Search Engine Optimization
SHAP	SHapley Additive exPlanations
SIA	Security Indicator Area
SIEM	Security Information and Event Management
SMOTE	Synthetic Minority Over-Sampling TEchnique
SMS	Short Message Service
SQL	Structured Query Language
SSL	Secure Sockets Layer
SVM	Support Vector Machine
TF-IDF	Term Frequency–Inverse Document Frequency
TLD	Top-Level Domain
TLS	Transport Layer Security
TPR	True Positive Rate
TPT	Threats and Pertinent Technologies
TTL	Time-To-Live
TTPs	Tactics, Techniques, and Procedures
UCI	University of California, Irvine
UIGEA	Unlawful Internet Gambling Enforcement Act
URL	Uniform Resource Locator
VAE	Variational AutoEncoder
WGAN-GP	Wasserstein Generative Adversarial Networks with Gradient Penalty
WSO	White Shark Optimizer
XAI	EXplainable Artificial Intelligence
XDR	EXtended Detection and Response
XSS	Cross-Site Scripting

References

1. Anti-Phishing Working Group (APWG). Phishing activity trends report Q1 2024. 2024 [cited 2026 Feb 16]. Available from: https://docs.apwg.org/reports/apwg_trends_report_q1_2024.pdf.
2. Federal Bureau of Investigation's Internet Crime Complaint Center (FBI IC3). Internet crime report 2024. 2024 [cited 2026 Feb 16]. Available from: https://www.ic3.gov/AnnualReport/Reports/2024_IC3Report.pdf.
3. Husain O. 99 global phishing statistics & industry trends (2023–2025) [Internet]. [cited 2026 Jan 10]. Available from: https://controld.com/blog/phishing-statistics-industry-trends/.
4. Baker E, Cartier M. Phishing trends report 2025 [Internet]. [cited 2026 Jan 10]. Available from: https://hoxhunt.com/guide/phishing-trends-report.

5. KnowBe4. Phishing threat trends report March 2025. 2025 [cited 2026 Feb 16]. Available from: https://www.knowbe4.com/hubfs/Phishing-Threat-Trends-2025_Report.pdf.
6. Bitaab M, Cho H, Oest A, Zhang P, Sun Z, Pourmohamad R, et al. Scam pandemic: how attackers exploit public fear through phishing. In: 2020 APWG Symposium on Electronic Crime Research (eCrime). Piscataway, NJ, USA: IEEE; 2020. p. 1–10. doi:10.1109/eCrime51433.2020.9493260.
7. Tang L, Mahmoud QH. A survey of machine learning-based solutions for phishing website detection. Mach Learn Knowl Extr. 2021;3(3):672–94. doi:10.3390/make3030034.
8. Jadhav A, Chandre PR. Survey and comparative analysis of phishing detection techniques: current trends, challenges, and future directions. Int J Artif Intell. 2025;14(2):853–66. doi:10.11591/ijai.v14.i2.pp853-866.
9. Min M, Lee DA. Illegal online gambling site detection using multiple resource-oriented machine learning. J Gambl Stud. 2024;40(4):2237–55. doi:10.1007/s10899-024-10337-z.
10. Musa M, Cvitić I, Peraković D. Survey of cybersecurity risks in online gambling industry. In: 8th EAI International Conference on Management of Manufacturing Systems. Cham, Switzerland: Springer Nature; 2023. p. 109–22. doi:10.1007/978-3-031-53161-3_8.
11. Rafsanjani AS, Kamaruddin NB, Behjati M, Aslam S, Sarfaraz A, Amphawan A. Enhancing malicious URL detection: a novel framework leveraging priority coefficient and feature evaluation. IEEE Access. 2024;12(2):85001–26. doi:10.1109/ACCESS.2024.3412331.
12. Edge ME, Sampaio PRF. A survey of signature based methods for financial fraud detection. Comput Secur. 2009;28(6):381–94. doi:10.1016/j.cose.2009.02.001.
13. Chiew KL, Yong KSC, Tan CL. A survey of phishing attacks: their types, vectors and technical approaches. Expert Syst Appl. 2018;106(4):1–20. doi:10.1016/j.eswa.2018.03.050.
14. Li W, Laghari SUA, Manickam S, Chong YW, Li B. Machine learning-enabled attacks on anti-phishing blacklists. IEEE Access. 2024;12:191586–602. doi:10.1109/ACCESS.2024.3516754.
15. Alghenaim M, Alkawsi G, Barnhart CR. The state of the art in AI-based phishing detection: a systematic literature review. Curr Future Trends AI Appl. 2025;1178:431–58. doi:10.1007/978-3-031-75091-5_23.
16. Kavya S, Sumathi D. Staying ahead of phishers: a review of recent advances and emerging methodologies in phishing detection. Artif Intell Rev. 2024;58(2):50. doi:10.1007/s10462-024-11055-z.
17. Thakur K, Ali ML, Obaidat MA, Kamruzzaman A. A systematic review on deep-learning-based phishing email detection. Electronics. 2023;12(21):4545. doi:10.3390/electronics12214545.
18. Thapa J, Chahal G, Gabreanu ŞV, Otoum Y. Phishing detection in the Gen-AI era: quantized LLMs vs. classical models. In: 2025 IEEE/ACIS 29th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). Piscataway, NJ, USA: IEEE; 2025. p. 856–63. doi:10.1109/SNPD65828.2025.11253009.
19. Afane K, Wei W, Mao Y, Farooq J, Chen J. Next-generation phishing: how LLM agents empower cyber attackers. In: 2024 IEEE International Conference on Big Data (BigData). Piscataway, NJ, USA: IEEE; 2024. p. 2558–67.
20. Jabbar H, Al-Janabi S. AI-driven phishing detection: enhancing cybersecurity with reinforcement learning. J Cybersecur Priv. 2025;5(2):26. doi:10.3390/jcp5020026.
21. Fraşzczak E, Fraşzczak D. A review of a website phishing detection taxonomy. In: IBIMA Conference on Artificial Intelligence and Machine Learning. Cham, Switzerland: Springer; 2024. p. 337–48. doi:10.1007/978-3-031-79086-7_26.
22. Apea NYO, Yin C. A review of the current status, categorization, and future prospects of web phishing detection methods. Int J Eng Res Technol. 2023;12(6):43–59.
23. Banks J, Waugh D. A taxonomy of gambling-related crime. Int Gambl Stud. 2019;19(2):339–57. doi:10.1080/14459795.2018.1554084.
24. Gainsbury SM, Hing N, Delfabbro PH, King DL. A taxonomy of gambling and casino games via social media and online technologies. Int Gambl Stud. 2014;14(2):196–213. doi:10.1080/14459795.2014.890634.
25. Verma RM, Dershowitz N, Zeng V, Boumber D, Liu X. Domain-independent deception: a new taxonomy and linguistic analysis. Front Big Data. 2025;8:1581734. doi:10.3389/fdata.2025.1581734.

26. Phillips R, Wilder H. Tracing cryptocurrency scams: clustering replicated advance-fee and phishing websites. In: 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). Piscataway, NJ, USA: IEEE; 2020. p. 1–8.
27. Lee K, Lim K, Kim H, Kwon Y, Kim D. 7 days later: analyzing phishing-site lifespan after detected. In: Proceedings of the ACM on Web Conference 2025. New York, NY, USA: ACM; 2025. p. 945–56. doi:10.1145/3696410.3714678.
28. Foremski P, Vixie P. The modality of mortality in domain names. 2018 [cited 2026 Feb 16]. Available from: https://www.virusbulletin.com/uploads/pdf/magazine/2018/VB2018-Vixie.pdf.
29. Affinito A, Sommese R, Akiwate G, Savage S, Claffy KC, Voelker GM, et al. Domain name lifetimes: baseline and threats. 2022 [cited 2026 Feb 16]. Available from: https://par.nsf.gov/servlets/purl/10594597.
30. Deacon A. Phishing attacks: newly registered domains still a prominent threat [Internet]. [cited 2026 Jan 10]. Available from: https://dnsrf.org/blog/phishing-attacks--newly-registered-domains-still-a-prominent-threat.
31. Li X, Wang J, Zhang X. Botnet detection technology based on DNS. Future Internet. 2017;9(4):55. doi:10.3390/fi9040055.
32. Almomani A. Fast-flux hunter: a system for filtering online fast-flux botnet. Neural Comput Appl. 2018;29(7):483–93. doi:10.1007/s00521-016-2531-1.
33. Chen S, Lang B, Xie C. Fast-flux malicious domain name detection method based on domain resolution spatial features. In: Proceedings of the 9th International Conference on Information Systems Security and Privacy (ICISSP). Setubal, Portugal: SciTePress; 2023. p. 240–51. doi:10.5220/0011872700003405.
34. Szurdi J, Houser R, Liu D. Fast flux 101: how cybercriminals improve the resilience of their infrastructure to evade detection and law enforcement takedowns. 2021 [cited 2026 Feb 16]. Available from: https://unit42.paloaltonetworks.com/fast-flux-101/.
35. Aaron G, Chapin L, Piscitello D, Rose K, Strutt C. Phishing landscape 2024. 2024 [cited 2026 Feb 16]. Available from: https://static1.squarespace.com/static/63dbf2b9075aa2535887e365/t/66cde404c8345e766972319c/1724769286084/PhishingLandscape2024.pdf.
36. Pritom MMA, Schweitzer KM, Bateman RM, Xu M, Xu S. Data-driven characterization and detection of COVID-19 themed malicious websites. In: 2020 IEEE International Conference on Intelligence and Security Informatics (ISI). Piscataway, NJ, USA: IEEE; 2020. p. 1–6. doi:10.1109/ISI49825.2020.9280522.
37. Young L. Threat spotlight: the geography and network characteristics of phishing attacks [Internet]. [cited 2026 Jan 10]. Available from: https://blog.barracuda.com/2021/04/07/threat-spotlight-geography-network-characteristics-phishing.
38. GOV.UK. Cyber security breaches survey 2023. 2023 [cited 2026 Feb 16]. Available from: https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2023/cyber-security-breaches-survey-2023.
39. Proofpoint. 2024 state of the phish–today's cyber threats and phishing protection. 2024 [cited 2026 Feb 16]. Available from: https://www.proofpoint.com/us/resources/threat-reports/state-of-phish.
40. Belyh A. 51 social engineering statistics for 2025 [Internet]. [cited 2026 Jan 10]. Available from: https://www.keevee.com/social-engineering-statistics.
41. Leahy P. Anti-phishing Act of 2005. 2005 [cited 2026 Feb 16]. Available from: https://www.congress.gov/bill/109th-congress/senate-bill/472/text.
42. GOV.UK. Fraud Act 2006. 2006 [cited 2026 Feb 16]. Available from: https://www.legislation.gov.uk/ukpga/2006/35/data.pdf.
43. Birthriya SK, Ahlawat P, Jain AK. Detection and prevention of spear phishing attacks: a comprehensive survey. Comput Secur. 2025;151(3):104317. doi:10.1016/j.cose.2025.104317.
44. Purwadi, Makhfud M, Jamaludin A. Legal accountability and policy gaps in social engineering-based phishing cybercrimes. Res Horiz. 2025;5(3):797–806. doi:10.54518/rh.5.3.2025.580.
45. Laksana AW. Cybercrime comparison under criminal law in some countries. J Pembaharuan Huk. 2018;5(2):217–26. doi:10.26532/jph.v5i2.3008.
46. Korea Management Association Consultants (KMAC). Research report on the fourth comprehensive plan for the sound development of the gambling industry. 2023 [cited 2026 Feb 16]. Available from: http://125.61.91.238:8080/SynapDocViewServer/viewer/doc.html?key=25f4af0081604bceab825742882c3aab&convType=img&convLocale=ko_KR&contextPath=/SynapDocViewServer.

47. Tran LT, Wardle H, Colledge-Frisby S, Taylor S, Lynch M, Rehm J, et al. The prevalence of gambling and problematic gambling: a systematic review and meta-analysis. Lancet Public Health. 2024;9(8):594–613. doi:10.1016/S2468-2667(24)00126-9.
48. Korean Criminological Association, KSTAT Research. 5th survey on illegal gambling. 2022 [cited 2026 Feb 16]. Available from: http://125.61.91.238:8080/SynapDocViewServer/viewer/doc.html?key=ab8dbc0258294aa383c9c58fcb508de4&convType=img&convLocale=ko_KR&contextPath=/SynapDocViewServer.
49. Leach JAS. Unlawful Internet gambling enforcement Act of 2006. 2006 [cited 2026 Feb 16]. Available from: https://www.congress.gov/109/plaws/publ347/PLAW-109publ347.pdf.
50. Setiawati S, Daulat PAS, Sunarto, Dewi S. The urgency of special regulations for online gambling in Indonesia. Int J Arts Soc Sci. 2022;5(7):108–18.
51. Rana MU, Shah MA, Al-Naeem MA, Maple C. Ransomware attacks in cyber-physical systems: countermeasure of attack vectors through automated web defenses. IEEE Access. 2024;12(6):149722–39. doi:10.1109/ACCESS.2024.3477631.
52. Sood AK, Enbody RJ. Malvertising—exploiting web advertising. Comput Fraud Secur. 2011;2011(4):11–6. doi:10.1016/S1361-3723(11)70041-0.
53. Nagunwa T. Behind identity theft and fraud in cyberspace: the current landscape of phishing vectors. Int J Cyber Secur Digit Forensics. 2014;3(1):72–84.
54. Tanti R. Study of phishing attack and their prevention techniques. Int J Sci Res Eng Manag. 2024;8(10):1–28. doi:10.55041/IJSREM38042.
55. Brennan A. AFP arrest alleged cybercriminals linked to LabHost website investigation [Internet]. [cited 2026 Jan 10]. Available from: https://www.news.com.au/technology/online/hacking/afp-arrest-alleged-cybercriminals-linked-to-labhost-website-investigation/news-story/4eaa41c935d4194e83edbb1be2baf76a.
56. Newman LH. It took 4 years to take down 'Avalanche,' a huge online crime ring [Internet]. [cited 2026 Jan 10]. Available from: https://www.wired.com/2016/12/took-4-years-take-avalanche-huge-online-crime-ring/.
57. Asif AUZ, Shirazi H, Ray I. Machine learning-based phishing detection using URL features: a comprehensive review. In: International Symposium on Stabilizing, Safety, and Security of Distributed Systems. Cham, Switzerland: Springer; 2023. p. 481–97. doi:10.1007/978-3-031-44274-2_36.
58. Bhat AE, Kavyasri MN, Arpitha HR, Sharath N, Pai K. A survey on phishing URL detection. Int J Nov Trends Innov. 2025;3(5):a248–51.
59. Opara C, Chen Y, Wei B. Look before you leap: detecting phishing web pages by exploiting raw URL and HTML characteristics. Expert Syst Appl. 2024;236:121183. doi:10.1016/j.eswa.2023.121183.
60. Opara C, Wei B, Chen Y. HTMLPhish: enabling phishing web page detection by applying deep learning techniques on HTML analysis. In: 2020 International Joint Conference on Neural Networks (IJCNN). Piscataway, NJ, USA: IEEE; 2020. p. 1–8. doi:10.1109/IJCNN48605.2020.9207707.
61. Tian Y, Yu Y, Sun J, Wang Y. From past to present: a survey of malicious URL detection techniques, datasets and code repositories. Comput Sci Rev. 2025;58(1):100810. doi:10.1016/j.cosrev.2025.100810.
62. Asiri S, Xiao Y, Alzahrani S, Li S, Li T. A survey of intelligent detection designs of HTML URL phishing attacks. IEEE Access. 2023;11:6421–43. doi:10.1109/ACCESS.2023.3237798.
63. Liu DJ, Lee JH. A CNN-based SIA screenshot method to visually identify phishing websites. J Netw Syst Manag. 2024;32(1):8. doi:10.1007/s10922-023-09784-7.
64. Abdelnabi S, Krombholz K, Fritz M. VisualPhishNet: zero-day phishing website detection by visual similarity. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. New York, NY, USA: ACM; 2020. p. 1681–98. doi:10.1145/3372297.3417233.
65. Ji F, Lee K, Koo H, You W, Choo E, Kim H, et al. Evaluating the effectiveness and robustness of visual similarity-based phishing detection models. In: 34th USENIX Security Symposium (USENIX Security 25). Berkeley, CA, USA: USENIX Association; 2025. p. 3201–20.

66. Van Dooremaal B, Burda P, Allodi L, Zannone N. Combining text and visual features to improve the identification of cloned webpages for early phishing detection. In: Proceedings of the 16th International Conference on Availability, Reliability and Security. New York, NY, USA: ACM; 2021. p. 1–10. doi:10.1145/3465481.3470112.
67. Dalgic FC, Bozkir AS, Aydos M. Phish-IRIS: a new approach for vision based brand prediction of phishing web pages via compact visual descriptors. In: 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). Piscataway, NJ, USA: IEEE; 2018. p. 1–8. doi:10.1109/ISMSIT.2018.8567299.
68. Luo X, Li Y, Cheng H, Yin L. AGCN-Domain: detecting malicious domains with graph convolutional network and attention mechanism. Mathematics. 2024;12(5):640. doi:10.3390/math12050640.
69. Guo W, Wang Q, Yue H, Sun H, Hu RQ. Efficient phishing URL detection using graph-based machine learning and loopy belief propagation. In: ICC 2025-IEEE International Conference on Communications. Piscataway, NJ, USA: IEEE; 2025. p. 2671–6. doi:10.1109/ICC52391.2025.11161346.
70. Bilot T, Geis G, Hammi B. PhishGNN: a phishing website detection framework using graph neural networks. In: 19th International Conference on Security and Cryptography. Setúbal, Portugal: SciTePress; 2022. p. 428–35. doi:10.5220/0011328600003283.
71. Sun Z, Teixeira AMH, Toor S. GNN-IDS: graph neural network based intrusion detection system. In: Proceedings of the 19th International Conference on Availability, Reliability and Security. New York, NY, USA: ACM; 2024. p. 1–12. doi:10.1145/3664476.3664515.
72. Pujol-Perich D, Suárez-Varela J, Cabellos-Aparicio A, Barlet-Ros P. Unveiling the potential of graph neural networks for robust intrusion detection. ACM SIGMETRICS Perform Eval Rev. 2022;49(4):111–7. doi:10.48550/arXiv.2107.14756.
73. Mohamed N, Nagaraj K, Melicher B, Farooqi S, Starov A, Stout B, et al. One step ahead in cyber hide-and-seek: automating malicious infrastructure discovery with graph neural networks. Unit 42. Santa Clara, CA, USA: Palo Alto Networks; 2025.
74. Kim H, Lee BS, Shin WY, Lim S. Graph anomaly detection with graph neural networks: current status and challenges. IEEE Access. 2022;10(9):111820–9. doi:10.1109/ACCESS.2022.3211306.
75. Bilot T, Madhoun NE, Agha KA, Zouaoui A. Graph neural networks for intrusion detection: a survey. IEEE Access. 2023;11:49114–39. doi:10.1109/ACCESS.2023.3275789.
76. Ju W, Yi S, Wang Y, Xiao Z, Mao Z, Li H, et al. A survey of graph neural networks in real world: imbalance, noise, privacy and OOD challenges. IEEE Trans Pattern Anal Mach Intell. 2025;48(3):3036–55. doi:10.1109/TPAMI.2025.3630673.
77. Shirazi H, Haynes K, Ray I. Towards performance of NLP transformers on URL-based phishing detection for mobile devices. Int J Ubiquitous Syst Pervasive Netw. 2022;17(1):35–42. doi:10.5383/JUSPN.17.01.005.
78. Maneriker P, Stokes JW, Lazo EG, Carutasu D, Tajaddodianfar F, Gururajan A. URLTran: improving phishing URL detection using transformers. In: MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM). Piscataway, NJ, USA: IEEE; 2021. p. 197–204. doi:10.1109/MILCOM52596.2021.9653028.
79. Kumar J. Detecting URL phishing using BERT and DistilBERT classifiers. In: Proceedings of the 12th International Conference on Soft Computing for Problem Solving (SocProS 2023). Singapore: Springer; 2024. p. 613–24. doi:10.1007/978-981-97-3180-0_40.
80. Mahendru S, Pandit T. SecureNet: a comparative study of DeBERTa and large language models for phishing detection. In: 2024 IEEE 7th International Conference on Big Data and Artificial Intelligence (BDAI). Piscataway, NJ, USA: IEEE; 2024. p. 160–9. doi:10.1109/BDAI62182.2024.10692765.
81. Kiazolu AA. Phishing website detection using BERT and LSTM for URL and content analysis. Int J Bus Manag Econ Rev. 2024;7(3):146–65. doi:10.2139/ssrn.5697143.
82. Meléndez R, Ptaszynski M, Masui F. Comparative investigation of traditional machine-learning models and transformer models for phishing email detection. Electronics. 2024;13(24):4877. doi:10.3390/electronics13244877.
83. Koide T, Nakano H, Chiba D. ChatPhishDetector: detecting phishing sites using large language models. IEEE Access. 2024;12(12):154381–400. doi:10.1109/ACCESS.2024.3483905.
84. Marchal S, François J, State R, Engel T. PhishStorm: detecting phishing with streaming analytics. IEEE Trans Netw Serv Manag. 2014;11(4):458–71. doi:10.1109/TNSM.2014.2377295.

85. Verma RM, Zeng V, Faridi H. Data quality for security challenges: case studies of phishing, malware and intrusion detection datasets. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. New York, NY, USA: ACM; 2019. p. 2605–7. doi:10.1145/3319535.3363267.
86. Khan SA, Khan W, Hussain A. Phishing attacks and websites classification using machine learning and multiple datasets (a comparative analysis). In: International Conference on Intelligent Computing. Cham, Switzerland: Springer; 2020. p. 301–13. doi:10.1007/978-3-030-60796-8_26.
87. Aung ES, Yamana H. Segmentation-based phishing URL detection. In: IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. New York, NY, USA: ACM; 2021. p. 550–6. doi:10.1145/3486622.3493983.
88. Sánchez-Paniagua M, Fidalgo E, Alegre E, Alaiz-Rodríguez R. Phishing websites detection using a novel multipurpose dataset and web technologies features. Expert Syst Appl. 2022;207:118010. doi:10.1016/j.eswa.2022.118010.
89. Prasad A, Chandra S. PhiUSIIL: a diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning. Comput Secur. 2024;136(4):103545. doi:10.1016/j.cose.2023.103545.
90. Hannousse A, Yahiouche S. Towards benchmark datasets for machine learning based website phishing detection: an experimental study. Eng Appl Artif Intell. 2021;104:104347. doi:10.1016/j.engappai.2021.104347.
91. Li Y, Tan HK, Meng Q, Lock ML, Cao T, Deng S, et al. PhishIntel: toward practical deployment of reference-based phishing detection. In: Companion Proceedings of the ACM on Web Conference 2025. New York, NY, USA: ACM; 2025. p. 2863–6. doi:10.1145/3701716.3715192.
92. Gnanesh A, Deepesh DA, Hegde B, Vyasamudri S, Sarasvathi V. Real time phishing detection using lexical analysis and visual similarity. In: International Conference on Broadband Communications, Networks and Systems. Cham, Switzerland: Springer; 2024. p. 157–75. doi:10.1007/978-3-031-81168-5_15.
93. Pawar S, Pawar S, Patil S, Sawant A, Sonawane S. Phishing detection using machine learning, web scraping, and real-time alerts. Int J Innov Res Technol. 2025;12(1):2342–8.
94. Roy SS, Nilizadeh S. PhishLang: a real-time, fully client-side phishing detection framework using MobileBERT. arXiv:2408.05667. 2024.
95. Whittaker C, Ryner B, Nazif M. Large-scale automatic classification of phishing pages. In: 17th Annual Network & Distributed System Security Symposium (NDSS). Reston, VA, USA: Internet Society; 2010. 2010 p.
96. Jeon D, Tak B. BlackEye: automatic IP blacklisting using machine learning from security logs. Wirel Netw. 2022;28(2):937–48. doi:10.1007/s11276-019-02201-5.
97. Security Zones. Spamhaus IP blocklists [Internet]. [cited 2026 Jan 10]. Available from: https://www.securityzones.net/spamhaus/ip-blocklists/.
98. Palo Alto Networks TECHDOCS. Network security: security policy. 2025 [cited 2026 Feb 16]. Available from: https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/network-security/security-policy-administration.pdf.
99. Bellekens X. Comprehensive insights into IP blocklists: an in-depth guide [Internet]. [cited 2026 Jan 10]. Available from: https://www.lupovis.io/comprehensive-insights-into-ip-blocklists-an-in-depth-guide.
100. Vidyakeerthi S, Nabeel M, Elvitigala C, Keppitiyagama C. PhishChain: a decentralized and transparent system to blacklist phishing URLs. In: Companion Proceedings of the Web Conference 2022. New York, NY, USA: ACM; 2022. p. 286–9. doi:10.1145/3487553.3524235.
101. Bell S, Komisarczuk P. An analysis of phishing blacklists: Google safe browsing, OpenPhish, and PhishTank. In: Proceedings of the Australasian Computer Science Week Multiconference. New York, NY, USA: ACM; 2020. p. 1–11. doi:10.1145/3373017.3373020.
102. Cisco Talos Intelligence Group (Talos). PhishTank: out of the net, into the tank [Internet]. [cited 2026 Jan 10]. Available from: https://phishtank.org/.
103. OpenPhish. About OpenPhish [Internet]. [cited 2026 Jan 10]. Available from: https://openphish.com/kb.html.
104. OpenPhish. OpenPhish academic use program [Internet]. [cited 2026 Jan 10]. Available from: https://openphish.com/academic_use.html.
105. Ariyadasa S, Fernando S, Fernando S. Phishing websites dataset. Mendeley Data. 2021. doi:10.17632/n96ncsr5g4.1.

106. Hranický R, Horák A, Polišenský J, Pouč P, Ondryáš O. Phishing and benign domain dataset. Zenodo. 2023. doi:10.5281/zenodo.8364668.
107. Feng T, Yue C. Visualizing and interpreting RNN models in URL-based phishing detection. In: Proceedings of the 25th ACM Symposium on Access Control Models and Technologies. New York, NY, USA: ACM; 2020. p. 13–24. doi:10.1145/3381991.3395602.
108. Feng T, Yue C. PhishingDataset [Repository]. [cited 2026 Jan 10]. Available from: https://github.com/vonpower/PhishingDataset.
109. Krog M, Chababy N. Phishing database [Repository]. [cited 2026 Jan 10]. Available from: https://github.com/Phishing-Database/Phishing.Database.
110. Vrbančič G, Fister I Jr, Podgorelec V. Datasets for phishing websites detection. Data Brief. 2020;33(6):106438. doi:10.1016/j.dib.2020.106438.
111. Vrbančič G. Phishing websites dataset. Mendeley Data. 2020. doi:10.17632/72ptz43s9v.1.
112. Alvarado E. Phishing-dataset [Repository]. [cited 2026 Jan 10]. Available from: https://huggingface.co/datasets/ealvaradob/phishing-dataset.
113. Lee H, Lee C, Lee Y, Lee J. BitAbuse: a dataset of visually perturbed texts for defending phishing attacks. In: Findings of the association for computational linguistics (NAACL 2025). Kerrville, TX, USA: ACL; 2025. p. 4367–84. doi:10.18653/v1/2025.findings-naacl.247.
114. Chung-Ang University AutoML Lab. BitAbuse [Repository]. [cited 2026 Jan 10]. Available from: https://huggingface.co/datasets/AutoML/bitabuse.
115. Skula I, Kvet M. Domain blacklist efficacy for phishing web-page detection over an extended time period. In: 2023 33rd Conference of Open Innovations Association (FRUCT). Piscataway, NJ, USA: IEEE; 2023. p. 257–63. doi:10.23919/FRUCT58615.2023.10142999.
116. Oest A, Safaei Y, Zhang P, Wardman B, Tyers K, Shoshitaishvili Y, et al. PhishTime: continuous longitudinal measurement of the effectiveness of anti-phishing blacklists. In: 29th USENIX Security Symposium (USENIX Security 20). Berkeley, CA, USA: USENIX Association; 2020. p. 379–96.
117. Zhang Z, Han D, Wu S, Sun W, Shi S. Identification and detection of illegal gambling websites and analysis of user behavior. Comput Sci Inf Syst. 2025;22(3):859–79. doi:10.2298/CSIS240930019Z.
118. Yang H, Du K, Zhang Y, Hao S, Li Z, Liu M, et al. Casino royale: a deep exploration of illegal online gambling. In: Proceedings of the 35th Annual Computer Security Applications Conference. New York, NY, USA: ACM; 2019. p. 500–13. doi:10.1145/3359789.3359817.
119. Gu Z, Gou G, Liu C, Yang C, Zhang X, Li Z, et al. Let gambling hide nowhere: detecting illegal mobile gambling apps via heterogeneous graph-based encrypted traffic analysis. Comput Netw. 2024;243:110278. doi:10.1016/j.comnet.2024.110278.
120. Zhang T, Jia D, Fu X, Zhang Z, Liu Q. WebAA: website association analysis via multi-resource similarity computation. In: International Conference on Computational Science. Cham, Switzerland: Springer; 2025. p. 50–64. doi:10.1007/978-3-031-97629-2_4.
121. Zhang T. WebAA-datasets [Repository]. [cited 2026 Jan 10]. Available from: https://github.com/SevenZhang123/WebAA-Datasets.
122. Boumber DA, Qachfar FZ, Verma RM. Domain-agnostic adapter architecture for deception detection: extensive evaluations with the DIFrauD benchmark. In: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). Torino, Italy: ELRA and ICCL; 2024. p. 5260–74.
123. Boumber DA, Qachfar FZ, Verma RM. DIFrauD [Repository]. [cited 2026 Jan 10]. Available from: https://huggingface.co/datasets/difraud/difraud.
124. Yang P, Zhao G, Zeng P. Phishing website detection based on multidimensional features driven by deep learning. IEEE Access. 2019;7:15196–209. doi:10.1109/ACCESS.2019.2892066.
125. Alhuzali A, Alloqmani A, Aljabri M, Alharbi F. In-depth analysis of phishing email detection: evaluating the performance of machine learning and deep learning models across multiple datasets. Appl Sci. 2025;15(6):3396. doi:10.3390/app15063396.

126. Por LY, Dai Z, Leem SJ, Chen Y, Yang J, Binbeshr F, et al. A systematic literature review on AI-based methods and challenges in detecting zero-day attacks. IEEE Access. 2024;12:144150–63. doi:10.1109/ACCESS.2024.3455410.
127. Song F, Lei Y, Chen S, Fan L, Liu Y. Advanced evasion attacks and mitigations on practical ML-based phishing website classifiers. Int J Intell Syst. 2021;36(9):5210–40. doi:10.1002/int.22510.
128. Shirazi H, Bezawada B, Ray I, Anderson C. Adversarial sampling attacks against phishing detection. In: Data and applications security and privacy XXXIII (DBSec 2019). Cham, Switzerland: Springer; 2019. p. 83–101. doi:10.1007/978-3-030-22479-0_5.
129. Kulkarni A, Balachandran V, Divakaran DM, Das T. From ML to LLM: evaluating the robustness of phishing web page detection models against adversarial attacks. Digit Threat Res Pract. 2025;6(2):1–25. doi:10.1145/3737295.
130. Lee J, Xin Z, See MNP, Sabharwal K, Apruzzese G, Divakaran DM. Attacking logo-based phishing website detectors with adversarial perturbations. In: European Symposium on Research in Computer Security. Cham, Switzerland: Springer; 2023. p. 162–82. doi:10.1007/978-3-031-51479-1_9.
131. Dai Z, Por LY, Chen YL, Yang J, Ku CS, Alizadehsani R, et al. An intrusion detection model to detect zero-day attacks in unseen data using machine learning. PLoS One. 2024;19(9):e0308469. doi:10.1371/journal.pone.0308469.
132. Staples D. A comparison of machine learning algorithms for zero-shot cross-lingual phishing detection [master's thesis]. Fredericton, NB, Canada: University of New Brunswick; 2023.
133. BigScience Workshop. BLOOM: a 176B-parameter open-access multilingual language model. arXiv:2211.05100. 2022.
134. Gemma Team Google DeepMind. Gemma 3 technical report. arXiv:2503.19786. 2025.
135. Wei X, Wei H, Lin H, Li T, Zhang P, Ren X, et al. PolyLM: an open source polyglot large language model. arXiv:2307.06018. 2023.
136. Üstün A, Aryabumi V, Yong ZX, Ko WY, D'souza D, Onilude G, et al. Aya model: an instruction finetuned open-access multilingual language model. arXiv:2402.07827. 2024.
137. Yue X, Song Y, Asai A, Kim S, de Dieu Nyandwi J, Khanuja S, et al. Pangea: a fully open multilingual multimodal LLM for 39 languages. arXiv:2410.16153. 2025.
138. Dalvi S, Gressel G, Achuthan K. Tuning the false positive rate/false negative rate with phishing detection models. Int J Eng Adv Technol. 2019;9:7–13. doi:10.35940/ijeat.A1002.1291S52019.
139. Puccetti T, Ceccarelli A. Detection latencies of anomaly detectors: an overlooked perspective? In: 2024 IEEE 35th International Symposium on Software Reliability Engineering (ISSRE). Piscataway, NJ, USA: IEEE; 2024. p. 37–48. doi:10.1109/ISSRE62328.2024.00015.
140. Sasi JK, Balakrishnan A. Generative adversarial network-based phishing URL detection with variational autoencoder and transformer. Int J Artif Intell. 2024;13(2):2165–72. doi:10.11591/ijai.v13.i2.pp2165-2172.
141. Pham TD, Pham TTT, Hoang ST, Ta VC. Exploring efficiency of GAN-based generated URLs for phishing URL detection. In: 2021 International Conference on Multimedia Analysis and Pattern Recognition (MAPR). Piscataway, NJ, USA: IEEE; 2021. p. 1–6. doi:10.1109/MAPR53640.2021.9585287.
142. Albahadili AJS, Akbas A, Rahebi J. Detection of phishing URLs with deep learning based on GAN-CNN-LSTM network and swarm intelligence algorithms. Signal Image Video Process. 2024;18(6):4979–95. doi:10.1007/s11760-024-03204-2.
143. Sern LJ, David YGP, Hao CJ. PhishGAN: data augmentation and identification of homoglyph attacks. In: 2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI). Piscataway, NJ, USA: IEEE; 2020. p. 1–6. doi:10.1109/CCCI49893.2020.9256804.
144. Li Q, Cheng M. Spear-phishing detection method based on few-shot learning. In: International Symposium on Advanced Parallel Processing Technologies. Singapore: Springer; 2023. p. 351–71. doi:10.1007/978-981-99-7872-4_20.
145. Lu Y, Wen L, Liu J, Liu Y, Tian X. Self-supervision can be a good few-shot learner. In: European Conference on Computer Vision (ECCV 2022). Cham, Switzerland: Springer; 2022. p. 740–58. doi:10.1007/978-3-031-19800-7_43.
146. Zhang G, Duan L, Wang W, Gong Z, Ma B. Multi-task self-supervised few-shot detection. In: Chinese Conference on Pattern Recognition and Computer Vision (PRCV). Singapore: Springer; 2023. p. 107–19. doi:10.1007/978-981-99-8555-5_9.

147. Yeboah PN, Kayes ASM, Rahayu W, Pardede E, Mahbub S. A framework for phishing and web attack detection using ensemble features of self-supervised pre-trained models. TechRxiv. 2025. doi:10.36227/techrxiv.173603362.21995515/v1.
148. Charmet F, Tanuwidjaja HC, Ayoubi S, Gimenez PF, Han Y, Jmila H, et al. Explainable artificial intelligence for cybersecurity: a literature survey. Ann Telecommun. 2022;77(11):789–812. doi:10.1007/s12243-022-00926-7.
149. Capuano N, Fenza G, Loia V, Stanzione C. Explainable artificial intelligence in cybersecurity: a survey. IEEE Access. 2022;10(2):93575–600. doi:10.1109/ACCESS.2022.3204171.
150. Richmond KM, Muddamsetty SM, Gammeltoft-Hansen T, Olsen HP, Moeslund TB. Explainable AI and law: an evidential survey. Digit Soc. 2024;3(1):1. doi:10.1007/s44206-023-00081-z.
151. Kesari A, Sele D, Ash E, Bechtold S. A legal framework for explainable artificial intelligence. In: Center for Law & Economics Working Paper Series. Zürich, Switzerland: ETH Zurich; 2024. doi:10.3929/ethz-b-000699762.
152. Alam S, Altiparmak Z. XAI-CF—examining the role of explainable artificial intelligence in cyber forensics. Eng Appl Artif Intell. 2026;167(4):113892. doi:10.1016/j.engappai.2026.113892.
153. Lim B, Huerta R, Sotelo A, Quintela A, Kumar P. EXPLICATE: enhancing phishing detection through explainable AI and LLM-powered interpretability. arXiv:2503.20796. 2025.
154. Lima G, Grgić-Hlača N, Jeong JK, Cha M. The conflict between explainable and accountable decision-making algorithms. In: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency. New York, NY, USA: ACM; 2022. p. 2103–13. doi:10.1145/3531146.3534628.
155. Calzarossa MC, Giudici P, Zieni R. An assessment framework for explainable AI with applications to cybersecurity. Artif Intell Rev. 2025;58(5):150. doi:10.1007/s10462-025-11141-w.
156. Kacheru G. The future of cyber defence: predictive security with artificial intelligence. Int J Adv Res Basic Eng Sci Technol. 2021;7(12):46–55.
157. Kharbanda J. Predictive AI: the “quiet catalyst” behind the future of cybersecurity [Internet]. [cited 2026 Jan 10]. Available from: https://www.group-ib.com/blog/predictive-ai/.
158. Darryl FR. Cybersecurity threat forecasting via social media signals and dark web monitoring: a multi-source predictive analytics framework. Int J Sci Res Comput Sci Inf Technol. 2025;6(1):1–8.
159. Almahmoud Z, Yoo PD, Damiani E, Choo KKR, Yeun CY. Forecasting cyber threats and pertinent mitigation technologies. Technol Forecast Soc Change. 2025;210(10):123836. doi:10.1016/j.techfore.2024.123836.