SoK: PHILTER: Uncovering Security and Functional Gaps in AI-based Phishing Website Detection Literature via an LLM-based Reasoning Framework

Mahbub Alam[†], Muhammad Lutfor Rahman[‡], Sonjoy Kumar Paul[†], Amy W. Hays[†], Aftab Hussain[†], Md Imanul Huq[†], and Nitesh Saxena[†]

[†]*Texas A&M University*
{mahbub.alam, skpaul, amy.hays, ahussain, imanulhuq, nsaxena}@tamu.edu

[‡]*California State University San Marcos,* mlrahman@csusm.edu

Abstract

Phishing websites remain a dominant enabler of cybercrime. In response, many academic AI-based phishing website detection methods have been developed, often inspiring the design of real-world systems. Although most studies report high accuracy, it remains unclear whether they meet real-world requirements such as resilience to evolving phishing tactics, robustness on diverse benign pages, interpretability, and privacy. We present PHILTER (PHishing detection literature Inspection via LLMs and Targeted Expert Review), a scalable framework for qualitatively assessing phishing website detection studies across four functionality and three security metrics. PHILTER leverages LLMs to extract evidence and draft rationales, which experts then validate and use to produce the final assessment. Applying it to 55 academic approaches reveals systemic gaps. No study fulfills all functionality and security requirements. None show evidence of effectively addressing diverse phishing tactics. Most approaches struggle to preserve privacy and adapt to evolving attacker strategies, and many risk elevated false alarms in practice due to limited testing on diverse benign pages. We also introduce a taxonomy of detection strategies (feature-based, similarity-based, identity-based, and hybrid) that highlights design trade-offs and helps explain these shortcomings. Our study reveals that accuracy-driven evaluation overlooks blind spots that undermine practical effectiveness and exposes a key open challenge: achieving high accuracy while fulfilling all functionality and security requirements. We provide actionable recommendations to guide the design of future defenses that pursue this simultaneous goal against evolving and adaptive phishing campaigns.

1 Introduction

Phishing is a cyberattack in which an adversary impersonates a trusted entity to steal sensitive information, such as login credentials, credit card numbers, or personal details. It remains a prevalent entry point for cybercriminals, exploiting human vulnerabilities to gain unauthorized access to systems and data. Over the past two decades, phishing has caused widespread disruption to organizations worldwide, driving financial losses, data breaches, and identity theft [10].

Many high-impact cybercrimes, including Business Email Compromise (BEC), ransomware, and large-scale data breaches, often begin with a phishing attack [44, 83, 96]. According to the FBI's Internet Crime Complaint Center (IC3), phishing was the most frequently reported cybercrime in 2024, with 193,407 incidents, down from 298,878 in 2023. Yet total losses from all cybercrimes surged from $12.5 billion in 2023 to $16.6 billion in 2024, underscoring phishing's persistent role as a gateway for high-impact attacks despite year-to-year fluctuations in volume [43, 44].

Phishing is executed through diverse tactics and channels, including email, social media, QR codes, and SMS [102]. Among these, phishing websites remain a dominant vector: in 2024, 45% of phishing emails contained a link to a phishing site [34]. Such sites often mimic legitimate platforms [1] with convincing branding and subtle URL variations to trick users into divulging credentials or other sensitive information.

The phishing landscape is evolving rapidly. Attackers now employ AI-driven techniques, multi-channel campaigns, and emerging tactics such as QR code phishing. Large language models (LLMs) and chatbots enable scalable generation of persuasive emails and fake websites. In 2024, 67.4% of phishing attacks incorporated AI, intensifying the need for equally adaptive defenses [34].

In response, industry and academia have developed increasingly sophisticated *AI-based phishing website detection* systems. Examples include Google Safe Browsing [28, 106], Microsoft Defender for Office 365 [72], and Cloudflare's automated detection [17], as well as numerous academic models [60]. Notably, Whittaker et al. [106] describe how academic research directly informed the design of Google's Safe Browsing system.

Despite these advances, the broader research landscape remains fragmented, with methods differing widely in detection logic, input modalities, evaluation protocols, and reporting practices. These approaches often report high benchmark

Table 1: Key findings and recommendations.

Key Finding	Recommendation
Limited tactic diversity. No study reports tactic-labeled evaluations, making effectiveness against specific tactics unclear.	Develop community-maintained, tactic-annotated phishing benchmarks with per-tactic performance evaluation to reveal blind spots and drive research on overlooked attack strategies.
Poor generalization on benign pages. Most methods lack evaluation on diverse benign pages, risking elevated false alarms.	Controlled inclusion of benign pages with legitimate brand elements and low-reputation sites to ensure realistic false-positive evaluation.
Underexplored interpretability. Most approaches provide only limited or no decision-level explanation, reducing transparency and trust.	Develop interpretability frameworks such as suspicious-element highlighting, visual overlays, and model-agnostic XAI to improve transparency and trust.
Limited drift adaptation. Most approaches lack mechanisms for handling drift, leaving them vulnerable to evolving phishing tactics.	Adopt time-aware evaluation and develop adaptive methods such as incremental learning, self-supervised adaptation, or stable drift-resilient features.
Limited resistance to active attacks. Most approaches lack rigorous evaluation against evasive manipulations, leaving their robustness in practice uncertain.	Develop systematic evaluation of active attacks (e.g., obfuscation, delayed content, redirects, manipulated visuals, adversarial examples) and design defenses that anticipate future evasions.
Privacy gaps. Most approaches transmit sensitive data externally, creating privacy risks.	Develop privacy-preserving approaches such as lightweight client-side detection, federated learning, and anonymized feature sharing.
Complementary strength integration is underexplored, preventing systems from satisfying all requirements simultaneously.	Combine methods with complementary strengths strategically so that systems can satisfy all requirements without inheriting individual weaknesses.

accuracy, yet provide little evidence on whether detectors address important but under-examined requirements, including resilience to evolving attacker tactics, robustness on diverse benign pages, interpretability, and privacy. To the best of our knowledge, no prior work has systematically assessed these approaches against a unified framework that reflects these deployment needs. This gap makes it difficult to compare strengths, uncover weaknesses, and understand trade-offs, especially given the diversity of approaches and the qualitative nature of many requirements.

Our Contributions. Main contributions are outlined below:

1. **A structured evaluation framework.** We present PHILTER (PHishing detection literature Inspection via LLMs and Targeted Expert Review), the first framework to systematically evaluate AI-based phishing website detection studies across four functionality and three security metrics.
2. **Scalable semi-automated assessment pipeline.** We design a two-stage pipeline where LLMs extract evidence and suggest preliminary assessments against codebook-defined metrics, which experts validate and finalize to ensure reliability.
3. **Strategy-based taxonomy.** We propose a taxonomy that categorizes detection methods by their underlying detection principle, feature-based, similarity-based, identity-based, and hybrid, clarifying how each approach reasons about phishing.

Key Findings and Recommendations. Our assessment reveals that existing approaches collectively fall short of meeting all functionality and security requirements, and methods that address security concerns often report reduced accuracy. These findings highlight a central open challenge: developing phishing detection systems that achieve high accuracy while simultaneously satisfying all core requirements. Our recommendations, summarized in Table 1, outline concrete steps toward addressing this challenge.

2 Related Work

Phishing website detection has received extensive attention from both academia and industry, yet prior surveys and systematization efforts lack structured frameworks for evaluating methods along dimensions relevant to real-world deployment. Existing surveys largely emphasize input types (e.g., URL, HTML, visual content), technical architecture (e.g., ML vs. DL), or reported accuracy. Prior SoKs take a broader view, but map the design space without systematically evaluating detection approaches against deployment-oriented requirements.

Early systematization efforts such as Dou et al. [32] surveyed software-based phishing detection schemes by classifying them into categories of datasets, features, techniques, and evaluation metrics, but did not evaluate detection approaches against deployment-oriented requirements. Das et al. [27] provided a broader systematization across multiple phishing vectors (email, websites, user studies), identifying four persistent challenges: real-time detection, active attackers, dataset quality, and the base-rate fallacy. However, their study did not translate these challenges into concrete evaluation dimensions, nor did it provide a methodology for systematically assessing whether individual detection methods address them.

Subsequent survey papers [2, 30, 31, 49, 60, 74, 115] primarily categorize phishing detection methods by input type, feature set, or classifier architecture. While useful for mapping the design space, such taxonomies provide limited insight into how detection methods reason about phishing or whether they remain resilient under evolving adversarial conditions.

Arp et al. [12] systematize methodological pitfalls in ML-based security research, including incomplete threat modeling, temporal leakage, inappropriate evaluation measures, and misleading interpretations under severe class imbalance. Several of our functionality and security requirements for effective AI-based phishing-website detection, such as evaluation transparency, adaptation to concept drift, and resistance to active attacks, reflect these general concerns. Other requirements, including coverage of diverse phishing tactics, benign-page diversity, interpretability, and privacy preservation, arise from the operational needs specific to phishing detection and extend beyond these general methodological pitfalls. Recent work by Ji et al. [48] highlights several limitations of similarity-based detectors, including dependence on up-to-date reference sets to remain effective on unseen brands, high false positives on brand-rich benign pages, and susceptibility to evasive manipulations such as altered logos, modified color schemes, and content obfuscation. Importantly, while their analysis focuses on visual similarity-based methods, the failures they uncover reflect deeper design and evaluation gaps that also impact other detection strategies. Building on these findings, PHILTER systematically evaluates core functionality and security requirements for effective phishing detection across all detection strategies, revealing critical gaps overlooked by prior evaluations.

3 Our PHILTER Framework

We introduce PHILTER, a structured framework that evaluates phishing detectors across seven functionality and security metrics. As shown in Figure 1, our end-to-end process starts with a curated corpus of 55 AI-based phishing website detection papers, selected and categorized according to the criteria and taxonomy in Section 5. We analyze these papers using a semi-automated pipeline (Section 3.1) that integrates LLM-assisted preliminary assessments with expert validation and correction, guided by the structured codebook in Section 4. The resulting labeled dataset is then examined to uncover trends and blind spots (Section 6), and the insights are distilled into actionable recommendations for advancing future phishing detection techniques (Section 7).

3.1 PHILTER Evaluation Pipeline

To enable consistent, scalable, and transparent evaluation of phishing website detection approaches, we implement the PHILTER evaluation pipeline, a semi-automated, two-stage workflow that combines LLM-assisted preliminary assessment with expert validation and correction. Each stage of the pipeline is guided by a structured codebook (Table 2) and illustrated in Figure 1.

LLM-assisted Preliminary Assessment. We use two LLMs, OpenAI's `o4-mini-2025-04-16` and Google's `gemini-2.5-pro`, to extract evidence, generate rationales, and propose preliminary labels that experts validate and finalize. These LLMs are selected for their strong reasoning capabilities and complementary architectural foundations, representing two leading LLM families [37, 78]. When the two models disagree, we invoke `o4-mini` with the arbitrator role to review the full paper, codebook, and prior assessments, and generate an arbitrated assessment with its own rationale and cited evidence. We explicitly prompt all LLMs to quote supporting text from the paper, a practice shown to improve factual accuracy and reduce hallucinations [16, 57]. All LLM calls use default API settings (temperature = 1 and top-p = 1 for o4-mini, and temperature = 1, top-p = 0.95, and top-k = 64 for Gemini), with custom prompts for evaluator and arbitrator roles. Prompt templates and representative outputs, illustrating how the LLM extracts evidence to justify its assessments and how disagreements between evaluators are resolved by the arbitrator, are shown in Appendix Listings 1 and 2.

Expert Validation and Correction. Two experts independently validate all LLM-generated preliminary assessments by checking the extracted evidence and rationale against the original paper and the codebook, correcting misinterpretations and resolving ambiguities to ensure each evaluation conforms to the formal criteria. Any disagreements were resolved through discussion until consensus, reducing subjective interpretation.

Reliability of LLM-assisted Assessment. Across 117 disagreement cases, the arbitrator LLM's assessment aligned with Gemini in 57.3% of cases and with o4-mini in 42.7%, indicating no model-family bias. LLM-generated rationales aligned with expert labels in 89% of cases, demonstrating their value in accelerating reliable evaluation. Expert-LLM assessment comparison and agreement rates per metric are provided in Appendix Tables 13 and 12, respectively.

3.2 Application of PHILTER

We applied PHILTER to 55 AI-based phishing detection papers, evaluating each method within its input modality and stated design scope.

Scope-Aware Assessment. Because phishing detectors differ in input modality, in whether they operate statically or interact with pages dynamically, and in whether they are designed for client-side or server-side deployment, we evaluated each PHILTER metric according to the information a system can observe and process. URL-only methods were assessed based on tactic diversity and attacker manipulations visible in lexical or structural URL patterns. Content-based systems were additionally evaluated for visual and structural cues (e.g., DOM manipulation, hidden elements), and dynamic systems for evasions observable at runtime (e.g., deferred forms, script-triggered redirects). We did not penalize methods for requirements that fall outside their stated design scope. For instance, Lee et al. [58] received a high rating for robustness against active attacks even though it did not address dynamic evasions, as those attacks fall outside its stated scope.

Integration-Aware Scoring. A metric was considered satisfied if the paper explicitly described how the requirement could be achieved through integration with complementary techniques. For example, the privacy requirement was considered fulfilled when a method described how it could combine server-side detection with client-side hashed URL lookups to protect sensitive data.

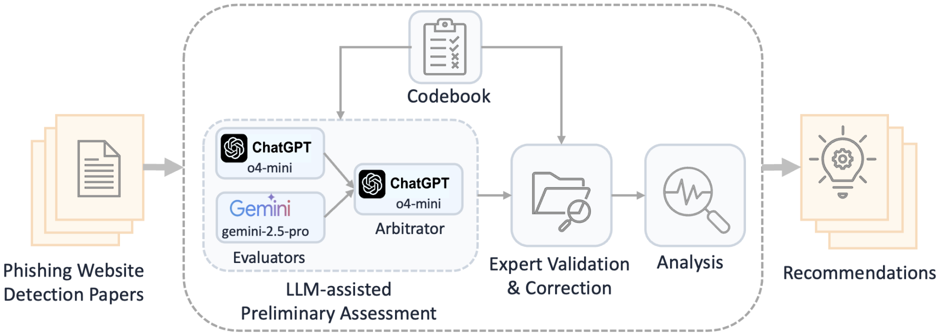

Figure 1: Overview of PHILTER framework.

3.3 Replicability and Broader Applicability

The PHILTER framework is designed to facilitate transparent, reproducible, and scalable evaluation of phishing website detection methods. Its clear definition of evaluation metrics and structured codebook enables researchers and stakeholders to accurately replicate our study.

Importantly, the functionality and security requirements captured by PHILTER can be extended to related domains such as phishing email detection [56], SMS phishing (smishing) detection [68], and scam detection [42], where detectors must cope with tactic diversity, minimize false positives on diverse benign inputs, remain interpretable, adapt to evolving campaigns, and safeguard privacy. By tailoring the PHILTER codebook to domain-specific threats and evaluation needs, the framework can be readily extended to enable consistent and scalable assessments across diverse cybercrime contexts.

4 PHILTER Metrics and Codebook

We identified the core functionality and security requirements for effective AI-based phishing detection methods by synthesizing insights from prior surveys, systematizations, analytical studies, and industry practices [12, 15, 18, 48, 55, 59, 76, 95], and developed a comprehensive evaluation codebook that operationalizes each requirement as a corresponding evaluation metric with explicit high, medium, and low fulfillment criteria (Table 2). To initialize the codebook, one author created the first version based on pilot assessments of representative papers. For independent validation, a second author applied the initial codebook to the same set of papers following open-coding practices [21]. Both authors iteratively compared their assessments, discussed disagreements, refined the criteria, and re-evaluated papers across multiple rounds. All authors reviewed subsequent revisions, provided feedback, and confirmed consensus before finalizing the codebook.

4.1 Functionality Metrics (F)

4.1.1 Diversity of Phishing Tactics (F1).

This metric evaluates whether a phishing detection method considers the wide range of tactics used in real-world phishing attacks. Phishers use a wide range of strategies to increase reach, longevity, and believability. These include misuse of HTTPS, hosting on trusted shared domains (e.g., GitHub Pages, Google Sites), deployment on compromised legitimate domains, URL redirection, subdomain spoofing (e.g., `login.paypal.com.fake.net`), homograph attacks (e.g., `app1e.com`), use of IP addresses as URLs, lexical manipulations (e.g., misspellings, hyphens, brand permutations), impersonation of less popular brands, and mimicry of internal systems such as HR portals or ticketing platforms [25, 55, 80]. Phishing detection methods can address such diversity by employing generalizable design choices, such as domain-independent features or tactic-agnostic mechanisms, that remain effective across varied hosting environments, brand targets, and evasion strategies.

4.1.2 Diversity of Benign Pages (F2).

Assesses whether a phishing detection method is evaluated on a sufficiently diverse set of benign webpages, so that the reported false positive rate reflects genuine robustness rather than performance on “easy” cases. When detectors fail to handle such diversity, false alarms increase—frustrating users, eroding trust in protective systems, and eventually leading to warning fatigue and eventual disabling of safeguards [33, 53, 76].

Many legitimate sites, especially small businesses, personal domains, or regional e-commerce shops, lack reputational indicators such as high domain rank or domain age. They may also legitimately embed elements from well-known brands (e.g., PayPal buttons, Google/Facebook OAuth widgets, social media logos) [65], which detectors can misinterpret as phishing signals. Robust evaluation should therefore include benign datasets that capture two key forms of diversity: **(i)**

Table 2: Codebook of PHILTER metric fulfillment criteria for phishing website detection studies (**C.** = Code).

C.	**Metric Name**	**High (●)**	**Medium (◐)**	**Low (○)**
Functionality Metrics (F)				
F1	Diversity of Phishing Tactics	Addresses multiple phishing tactics (or is tactic-agnostic), provides a tactic-level breakdown of the evaluation set, and reports per-tactic performance.	Addresses multiple phishing tactics (or is tactic-agnostic) without per-tactic evaluation.	Ignores phishing tactic diversity in design and evaluation.
F2	Diversity of Benign Pages	Evaluation intentionally includes both low-reputation benign domains and benign pages with legitimate brand elements.	Evaluation includes either low-reputation benign domains or benign pages containing brand elements.	Evaluation limited to high-reputation/popular benign sites.
F3	Interpretability	Provides per-decision explanations for individual predictions (e.g., visual cues, transparent rules).	Provides only global or aggregate interpretability (e.g., feature importance).	Provides no global or per-decision interpretability.
F4	Evaluation Transparency	Reports class-specific metrics and at least one aggregate metric suitable for evaluation under class imbalance.	Reports some relevant metrics but lacks full coverage.	Omits metrics needed to evaluate under class imbalance.
Security Metrics (S)				
S1	Adaptation to Concept Drift	Uses drift-aware design (e.g., uses adaptive strategies or drift-resilient features), **and** evaluates on temporally disjoint data.	Uses drift-aware design, **or** evaluates on temporally disjoint data.	Assumes static phishing tactics.
S2	Resistance to Active Attacks	Evaluates against a diverse set of active attacks within its input modality and scope.	Evaluates against only a subset of relevant active attacks.	No evaluation against active attacks.
S3	Privacy Preservation	Ensures sensitive data is not transmitted externally, **or** uses privacy-preserving mechanisms (e.g., hashed URL lookups, federated learning).	Partial privacy safeguards (e.g., transmitting truncated or sanitized URLs).	Sensitive data is transmitted to external servers for inference.

low-reputation or long-tail domains (e.g., Alexa/Tranco low-rank sites, personal domains, recently registered domains, or regional businesses); and **(ii)** legitimate brand element usage (e.g., PayPal/OAuth login widgets, social media buttons, embedded logos).

Including these harder benign subsets alongside popular, well-established domains provides a more faithful measure of detector robustness. Without such diversity, evaluations risk underestimating false positive rates by excluding benign pages most susceptible to misclassification.

4.1.3 Interpretability (F3).

Assesses whether a phishing detection method provides clear reasoning for its decisions [18]. Interpretability helps users, developers, and auditors understand why a site is flagged as phishing or legitimate, including which features or signals drive the outcome. Explanations may come from inherently transparent models (e.g., decision trees) or post hoc Explainable AI (XAI) methods such as SHAP, LIME, Integrated Gradients, or Grad-CAM [67, 94]. Interpretability fosters trust and error analysis by allowing developers to trace misclassifications, security teams to validate behavior, and users to accept warnings.

Interpretability requirements vary by input modality. In content-based methods, explanations may point to suspicious page regions [112] (e.g., heat-maps or bounding-box overlays), visual elements, or structural cues; in URL-based models, they may surface lexical patterns, domain structures, or contextual signals.

4.1.4 Evaluation Transparency (F4).

Evaluates whether the paper transparently and comprehensively reports the empirical evaluation of its phishing detection method, using performance metrics that meaningfully reflect real-world conditions, especially class imbalance, where phishing sites are rare [15, 35]. Accuracy alone is misleading. For example, a model that labels all pages as legitimate may achieve high accuracy while failing entirely to detect phishing sites. A robust evaluation should report the following class-specific metrics:

Precision: Measures the proportion of predicted phishing sites that are truly phishing, emphasizing the importance of reducing false alarms.

Recall: Measures the proportion of phishing sites correctly detected, highlighting the need to minimize missed threats.

F1-score: Provides the harmonic mean of precision and recall, yielding a single value that captures both types of error.

In addition, the evaluation should include at least one aggregate metric that meaningfully reflects overall classification quality, especially in the presence of class imbalance. Acceptable options include:

Area Under the ROC Curve (AUC-ROC): Quantifies how well a model ranks phishing sites above legitimate ones across all decision thresholds. Its reliability decreases under class imbalance.

Area Under the Precision–Recall Curve (AUC-PRC): Captures the tradeoff between precision and recall, and is especially informative in highly imbalanced settings such as phishing detection [89].

Matthews Correlation Coefficient (MCC): Assesses classification quality by incorporating all confusion matrix components (TP, FP, TN, FN), making it suitable even for hard-label classifiers [19].

Together, these metrics provide a comprehensive evaluation of phishing detection, capturing class-specific behavior, overall quality, and the balance between false positives and false negatives.

4.2 Security Metrics (S)

4.2.1 Adaptation to Concept Drift (S1).

Phishing strategies evolve over time, adapting to changes in technology, defenses, and user behavior. This evolution leads to concept drift: a shift in the statistical properties of phishing websites, reflected in changes to URL structures, hosting patterns, page layouts, and other feature characteristics [76]. As a result, detection models trained on historical data often degrade over time, limiting their practical utility [75].

This metric evaluates whether a detection method anticipates and addresses drift using adaptive strategies [31, 69] such as continual learning, periodic retraining, or stable drift-resilient features that remain effective despite attacker adaptations (e.g., semantic or visual identity features) [46, 71]. Methods that avoid drift-prone signals may offer long-term robustness without frequent retraining.

To be meaningful, evaluation should be time-aware, for example by training on older data and testing on newer samples [69], or by validating models in live settings. Such protocols ensure that resilience claims extend to future phishing attacks.

4.2.2 Resistance to Active Attacks (S2).

This metric assesses whether a phishing detection method is resilient to active attacks—deliberate manipulations or evasions intended to bypass phishing detectors. Such attacks may target commonly used features, for example by inserting junk characters to distort lexical statistics, replacing text with images, injecting Base64-encoded scripts, obfuscated JavaScript, junk HTML, or invisible elements [4]. They may also involve adversarial examples crafted to mislead classifiers [39, 79], or delivery-stage evasions such as dynamic content loading (e.g., via AJAX), deferred execution (e.g., login forms revealed only after interaction), or cloaking (e.g., showing benign content to crawlers) [59, 70].

4.2.3 Privacy Preservation (S3).

Evaluates whether a phishing website detection method protects user privacy during detection, particularly in deployment settings integrated into user-facing environments (e.g., browsers, email clients, mobile apps) [80]. It assesses whether sensitive data, such as full URLs (including query parameters), webpage content, or screenshots, is transmitted from the user's device to external servers. Even URL-only detection methods can pose risks if raw URLs are sent off-device, as these may contain session tokens, personal identifiers, or application-specific parameters.

Privacy-preserving design is essential for user trust, regulatory compliance (e.g., GDPR, CCPA), and adoption [23, 45]. In practice, privacy can be achieved by ensuring that sensitive or identifiable data is not transmitted off-device and by adopting privacy-aware deployment or data handling mechanisms. Common approaches include hashed URL lookups (e.g., prefix-hash-based blacklists like Google Safe Browsing), federated learning that shares only model updates (e.g., gradients or weights), and anonymized non-reversible feature vectors that cannot be traced back to the original URL.

5 Study Selection and Taxonomy

This section details our systematic process for identifying and selecting state-of-the-art AI-based phishing website detection methods, and presents the taxonomy we use to organize and analyze these approaches throughout the paper.

5.1 Study Selection Criteria

We systematically selected state-of-the-art AI-based phishing website detection methods by following the PRISMA guidelines [73], using a multi-stage process: database search, title and abstract screening, full-text analysis, and snowballing [107]. We sourced articles from DBLP [26], a major computer science bibliography, using the query:

phishing (url|page|web|site) (detect|attack|adversarial)

This query captures variations such as "phishing URL detection", "web-based phishing attack", "adversarial phishing site", and similar, returning 457 results. Filtering for studies published between 2019 and 2025 reduced the set to 342. We then excluded non-peer-reviewed works, such as theses, book chapters, and informal publications, yielding 288 articles from 178 venues.

To ensure quality, we retained conference papers from A*, A, or B ranked venues according to the CORE conference rankings [22], and journal articles from Q1 journals based on the Scimago SJR rankings [93], resulting in 109 high-quality publications across 46 venues. We then conducted title and abstract screening using the following criteria:

Inclusion Criteria: Studies must propose end-to-end phishing website detection methods that incorporate an AI component (e.g., machine learning, deep learning) as part of the detection pipeline.

Exclusion Criteria: (1) Studies that rely solely on heuristics, blacklists, or rule-based systems without any AI-driven element; (2) Studies focused on phishing in other media (e.g., email, voice, SMS, QR codes, or social platforms); (3) Studies that analyze or attack existing methods without proposing a new AI-based approach; and (4) Studies focused exclusively on preprocessing tasks (e.g., captcha-solving, logo detection) or on human-centric strategies (e.g., awareness training, behavioral analysis).

This screening yielded 92 articles. After full-text analysis, we excluded 6 that did not meet the selection criteria, leaving 86 studies. Since feature-based methods dominated this set, we applied a citation-based filter to retain only impactful or recent works and to avoid overrepresentation. This filtering was applied primarily to feature-based approaches, while less common approaches, such as similarity-based and identity-based, were retained in full. For feature-based papers, we used the following citation thresholds: ≥250 for 2019, ≥150 for 2020, ≥75 for 2021, ≥50 for 2022, ≥25 for 2023, ≥15 for 2024, and ≥0 for 2025. Citation thresholds varied by publication year to offset citation-age effects. Foundational papers, such as those introducing privacy-preserving phishing detection, proposing mechanisms for handling concept drift, or appearing in top-tier venues, were retained regardless of citation count. This resulted in a total of 38 papers.

To enhance coverage, we applied both backward and forward snowballing, examining references of selected papers and other papers citing them. While the publication date filter was relaxed, all other inclusion and exclusion criteria remained in effect. The snowballing yielded 17 additional papers published between 2010 and 2024. In total, 55 papers were selected for our final analysis. The venue distribution of the selected papers is shown in Appendix Table 11.

5.2 Taxonomy of Phishing Website Detection Methods

We classify AI-based phishing website detection methods into four principal strategies, **feature-based**, **similarity-based**, **identity-based**, and **hybrid**, based on their core detection logic. This taxonomy, developed to support PHILTER's systematic evaluation, organizes the design space in a way that highlights how different strategies reason about phishing.

5.2.1 Feature-Based Detection

Feature-based methods detect phishing by extracting features from inputs such as URLs, webpage content, and metadata, which are then analyzed using machine learning models [51, 88, 104]. Early approaches rely heavily on engineered features such as URL length, character distributions, and suspicious lexical or structural patterns [51, 88]. More recent work leverages deep learning models to automatically learn semantic and structural representations directly from raw inputs, avoiding manual feature design [90, 104]. Some systems combine both directions: for example, BGL-PhishNet [86] extracts lexical, host-based, and metadata features from URLs and related domain data, while simultaneously employing BERT and Graph Neural Networks (GNN) to capture deep semantic and structural patterns. A LightGBM classifier then integrates the manually engineered and automatically learned features, enabling robust detection across diverse attack strategies.

5.2.2 Similarity-Based Detection

Similarity-based methods detect phishing by measuring how closely a suspect page mimics a legitimate website, using visual, structural, or textual comparison [1, 63]. For instance, PhishIntention [65] leverages deep learning-based computer vision models to analyze the visual layout and elements of webpages, and employs OCR-enhanced logo matching to accurately identify attempts at brand mimicry. By comparing key page components to a curated reference set of legitimate sites, these methods enhance detection of sophisticated imitation attacks. Similarly, Phishpedia [63], VisualPhishNet [1], and DeltaPhish [24] apply techniques such as screenshot comparison, DOM structure analysis, or textual similarity to capture resemblance between phishing and legitimate sites. Such approaches are particularly effective against brand-targeted campaigns, but their reliance on reference sets can limit scalability and make them vulnerable to zero-day attacks against previously unseen brands.

5.2.3 Identity-Based Detection

Identity-based methods infer the brand or organization a webpage claims to represent and then verify whether the hosting domain is genuinely associated with that brand. This typically involves extracting brand indicators such as logos for visual identity and brand-specific keywords or named entities for textual identity, and cross-checking them against external sources like search engines, knowledge graphs, or large language models [20, 54, 58, 92]. Some approaches utilize both signals in combination: for example, Tan et al. [97] integrates logo detection with textual keyword extraction and validates the inferred brand through search results. Unlike similarity-based methods, which rely on resemblance to a curated reference set of legitimate sites, identity-based detection verifies the claimed brand directly through external sources. By avoiding dependence on static reference lists, it remains effective even against phishing pages that target emerging brands.

5.2.4 Hybrid Approaches

Hybrid approaches combine two or more detection strategies to leverage their complementary strengths. Some methods integrate identity-based and similarity-based techniques. For example, KnowPhish [61] first performs visual matching by comparing logos from the target webpage to a large, automatically constructed brand knowledge base (containing logo images, domains, and aliases). If no logo is detected or matched, it employs an LLM to extract potential brand names

from the webpage HTML and URL, enabling robust brand inference. PhishLLM [64] follows a similar identity–similarity pipeline: it applies logo detection, OCR, and image captioning to generate prompts for an LLM to infer the intended brand, and then validates the prediction by visually comparing the detected logo with those retrieved from web search results for that brand. Hybrid methods also combine feature-based detection with either similarity-based techniques [81, 85] or identity-based logic [29, 41] to broaden coverage and improve resilience. By integrating multiple sources of evidence, such as features, logos, textual cues, and external knowledge, these approaches aim to reduce the limitations of individual strategies and provide more comprehensive detection.
While the above taxonomy categorizes phishing website detection methods by their core detection strategies, systems can also be differentiated along several orthogonal dimensions, including *input source* (e.g., URL, webpage content, external metadata), *deployment mode* (client-side vs. server-side), and *detection mode* (real-time vs. non-real-time). We analyze cross-cutting trends along these additional dimensions in Section 6.2, offering a broader perspective beyond core detection strategies.

6 Evaluation Findings

Our findings, as detailed in Table 3, show that among the 55 phishing detection studies, none satisfy all seven functionality and security requirements, and only 18 achieve more than one metric at the "High" level. Most are concentrated at the "Medium" or "Low" level, highlighting that deployment-critical needs remain insufficiently addressed. In the following subsections, we analyze metric fulfillment by detection strategy, input modality, deployment and detection modes, reported accuracy, publication year, and citation count, discussing the factors driving these patterns.

6.1 Metric Fulfillment Across Detection Strategies

Table 4 highlights (with gray shading) the most common fulfillment level across the seven PHILTER metrics for each detection strategy, providing a high-level view of overall trends. In this section, we discuss these trends and gaps in detail for each strategy, as well as cross-strategy observations.

Table 3: Evaluations of individual phishing detection studies. **Ap.** = Approach. The "Count" column shows fulfillment as **#High / #Medium / #Low**, with shaded cells marking studies that reach "High" in more than one metric. Functionality (F1–F4) and security (S1–S3) metrics are defined in Table 2.

Ap.	**Paper**	**Year**	**Funct. (F)**				**Secur. (S)**			**Count**
			F1	**F2**	**F3**	**F4**	**S1**	**S2**	**S3**	
Feature	PDSMV3-DCRNN [82]	2025	◐	○	◐	●	○	○	○	1/2/4
	Akçam et al. [5]	2025	○	○	○	●	○	◐	○	1/1/5
	Kavya et al. [52]	2025	◐	○	◐	●	●	●	●	4/2/1
	BGL-PhishNet [86]	2025	◐	◐	○	●	○	○	○	1/2/4
	SPWDM [110]	2025	◐	○	◐	◐	○	◐	○	0/4/3
	AdaptPUD [113]	2025	◐	○	○	●	●	◐	○	2/2/3
	Feng et al. [36]	2024	◐	○	○	●	○	○	○	1/1/5
	WebPhish [77]	2024	◐	○	○	◐	○	○	●	1/2/4
	DEPHIDES [87]	2024	◐	○	○	◐	◐	○	●	1/3/3
	RNT-J [9]	2024	◐	○	◐	●	○	○	○	1/2/4
	Geest et al. [101]	2024	○	○	○	●	○	◐	●	2/1/4
	PhishingRTDS [13]	2024	◐	○	○	◐	○	◐	○	0/3/4
	Schesny et al. [92]	2024	○	○	○	○	○	○	○	0/0/7
	STFL [90]	2024	◐	◐	○	◐	◐	○	●	1/4/2
	LSD [51]	2023	◐	○	○	◐	○	○	●	1/2/4
	Bahaghighat et al. [14]	2023	○	○	○	●	○	○	○	1/0/6
	Jalil et al. [47]	2023	◐	○	◐	●	○	○	●	2/2/3
	PhishBERT [104]	2023	◐	○	○	◐	○	○	○	0/2/5
	Tiny-BERT [40]	2023	◐	○	○	◐	○	○	●	1/2/4
	Almomani et al. [7]	2022	◐	○	◐	◐	○	○	○	0/3/4
	Sánchez et al. [91]	2022	◐	○	○	◐	◐	○	●	1/3/3
	MLSELM [50]	2022	○	○	○	◐	○	○	○	0/1/6
	PhishNOT [6]	2022	◐	○	◐	◐	○	○	○	0/3/4
	POC [11]	2022	◐	◐	○	◐	○	●	○	1/3/3
	Gupta et al. [38]	2021	◐	○	◐	●	○	○	●	2/2/3
	Mahmoud et al. [98]	2021	○	○	○	◐	◐	○	○	0/2/5
	Xiao et al. [109]	2021	○	○	○	◐	○	○	●	1/1/5
	Wei et al. [105]	2020	◐	◐	○	●	◐	○	●	2/3/2
	LBET [8]	2020	◐	○	○	◐	○	○	○	0/2/5
	Sahingoz et al. [88]	2019	◐	○	○	◐	○	◐	●	1/3/3
	MFPD [111]	2019	◐	◐	○	◐	◐	○	○	0/4/3
	Rao et al. [84]	2019	◐	○	◐	◐	○	◐	○	0/4/3
	Li et al. [62]	2019	◐	◐	◐	◐	◐	○	●	1/5/1
	CANTINA+ [108]	2011	◐	○	◐	◐	●	○	○	1/3/3
	Whittaker et al. [106]	2010	◐	○	◐	◐	●	○	●	2/3/2
Similarity	Wang et al. [103]	2024	◐	○	●	●	◐	◐	○	2/3/2
	PhishIntention [65]	2022	◐	●	●	◐	◐	●	○	3/3/1
	CatchPhish [99]	2022	○	○	○	◐	○	◐	●	1/2/4
	Phishpedia [63]	2021	◐	●	●	◐	◐	◐	○	2/4/1
	VisualPhishNet [1]	2020	◐	◐	○	◐	◐	◐	○	0/5/2
	Abeywardena et al. [3]	2020	◐	○	○	◐	◐	○	○	0/3/4
	DeltaPhish [24]	2017	○	○	◐	◐	◐	◐	●	1/4/2
Identity	Lee et al. [58]	2024	◐	◐	●	◐	◐	●	○	2/4/1
	ChatPhishDetector [54]	2024	◐	○	●	●	◐	◐	○	2/3/2
	Tan et al. [97]	2023	◐	○	●	◐	●	◐	○	2/3/2
	Chiew et al. [20]	2015	◐	●	●	◐	◐	○	○	2/3/2
Hybrid	PhiUSIIL [81]	2024	◐	○	◐	●	●	◐	○	2/3/2
	KnowPhish [61]	2024	◐	○	●	◐	●	◐	○	2/3/2
	PhishLLM [64]	2024	◐	◐	●	◐	●	●	○	3/3/1
	DynaPhish [66]	2023	◐	◐	●	◐	●	◐	○	2/4/1
	Dooremaal et al. [100]	2021	◐	◐	○	◐	○	○	○	0/3/4
	BlackPhish [85]	2020	◐	○	◐	●	◐	◐	○	1/4/2
	PhishFencing [114]	2020	◐	○	○	◐	◐	◐	●	1/4/2
	Ding et al. [29]	2019	◐	○	◐	●	○	◐	○	1/3/3
	He et al. [41]	2011	◐	○	○	◐	◐	○	○	0/3/4

●=High, ◐=Medium, ○=Low

6.1.1 Feature-Based Methods

For functionality metrics, phishing tactic diversity (**F1**) is mostly partial (80% medium, 20% low), since evaluation datasets typically lack a per-tactic breakdown. Benign page diversity (**F2**) is the weakest dimension (83% low), as most evaluations rely on popular domains rather than diverse benign pages. Interpretability (**F3**) shows similar limitations (66% low, 34% medium), with explanations restricted to global

Table 4: Metric fulfillment vs. detection strategy (**C.** = Count). Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low

Approach	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
Feature	35	0	80	20	0	17	83	0	34	66	34	63	3	11	20	69	6	20	74	43	0	57
Similarity	7	0	71	29	29	14	57	43	14	43	14	86	0	0	86	14	14	71	14	29	0	71
Identity	4	0	100	0	25	25	50	100	0	0	25	75	0	25	75	0	25	50	25	0	0	100
Hybrid	9	0	100	0	0	33	67	33	33	33	33	67	0	44	33	22	11	67	22	11	0	89
Overall	55	0	84	16	5	20	75	18	29	53	31	67	2	16	35	49	9	36	55	33	0	67

feature importance rather than per-decision reasoning. Evaluation transparency (**F4**) is comparatively stronger (34% high; only 3% low), reflecting partial but common reporting of standard metrics such as precision, recall, F1, and AUC/MCC.

For security metrics, adaptation to concept drift (**S1**) is rarely addressed (69% low; 11% high), reflecting a reliance on static feature sets rather than adaptive learning. Resistance to active attacks (**S2**) is likewise underdeveloped (74% low; only 6% high), with most systems providing no defense against adversarial evasion. Privacy preservation (**S3**) performs better, with nearly half of methods achieving strong privacy (43% high), typically by deploying detection on the client side rather than transmitting user data externally.

6.1.2 Similarity-Based Methods

Similarity-based methods leverage the resemblance of phishing pages to high-reputation sites, which makes them effective at detecting brand-targeted attacks but limits coverage of phishing tactic diversity (**F1**), especially for tactics that do not rely on close visual or structural resemblance. Most methods achieve only medium fulfillment (71%), while the remainder are low (29%). Benign page diversity (**F2**) is similarly weak, with 57% low, 14% medium, and only 29% high, reflecting evaluations limited to high-reputation domains. Interpretability (**F3**) is stronger than in feature-based methods, with 43% achieving high fulfillment and 14% medium, though over half still provide no interpretability. Evaluation transparency (**F4**) is better addressed than most other metrics, with all methods reporting at least some standard evaluation metrics (86% medium, 14% high).

For security metrics, adaptation to concept drift (**S1**) remains weak, since most methods rely on fixed reference lists of legitimate sites. Most (86%) provide only medium fulfillment, with none reaching high. Resistance to active attacks (**S2**) is somewhat better addressed, with 14% high and 71% medium, though some remain low (14%). Privacy preservation (**S3**) shows a clear divide: a subset achieve high fulfillment through client-side deployment (29%), while the majority are low (71%) because they transmit page content to external servers, raising privacy risks.

6.1.3 Identity-Based Methods

Identity-based methods infer the intended brand from logos, page text, and URLs, then verify whether the hosting domain is genuinely associated with that brand using external sources (e.g., search engines or knowledge bases). This brand-verification focus helps catch direct impersonation but limits its coverage of phishing tactic diversity (**F1**): all methods only partially satisfy this metric (100% medium; 0% high). Benign page diversity (**F2**) is somewhat better (25% high, 25% medium), though evaluations still overlook many benign pages that legitimately display brand elements (e.g., OAuth buttons or embedded logos). These methods achieve consistently high interpretability (**F3**), as decisions are explained through brand–domain matching. Evaluation transparency (**F4**) is moderate (25% high, 75% medium).

For the security metrics, adaptation to concept drift (**S1**) is comparatively stronger (25% high, 75% medium), since automated brand indexing via search engines enables detection of newly emerging targets without manual updates. However, most works still lack temporal disjoint testing to validate long-term adaptability. Resistance to active attacks (**S2**) is moderate (25% high, 50% medium, 25% low), as few methods are evaluated against diverse evasion strategies such as logo obfuscation or adversarial perturbations. Privacy preservation (**S3**) remains weak (100% low), reflecting the reliance on external lookups that transmit webpage content or screenshots off-device.

6.1.4 Hybrid Methods

Hybrid methods integrate multiple detection strategies, such as feature-based, visual similarity, and identity verification, to detect phishing attacks that single approaches might miss. This broader scope helps them address a wider variety of phishing tactics, but phishing tactic diversity (**F1**) remains only partial across all evaluated studies (100% medium), since none perform per-tactic evaluation. Benign page diversity (**F2**) remains limited (33% medium, 67% low), as evaluations often omit (i) regional or low-reputation benign pages and (ii) benign pages that legitimately contain brand elements. Interpretability (**F3**) varies: 33% of works provide high interpretability, 33% medium, and 33% none, often because some integrated components remain black-box. Evaluation transparency (**F4**) is comparatively stronger, with 33% high and 67% medium, as most methods report key metrics such as precision, recall, and F1, though some omit measures like AUC-ROC or MCC.

For security metrics, adaptation to concept drift (**S1**) is mixed (44% high, 33% medium, 22% low). Identity-verification components help detect new targets, but feature-based components may degrade as visual or structural features evolve. Most methods also lack temporal disjoint testing to validate long-term adaptability. Resistance to active attacks (**S2**) shows similar patterns (11% high, 67% medium, 22%

low), with hybrid approaches sometimes countering evasions against one component but rarely covering the full attack surface. Privacy preservation (**S3**) is weak overall (89% low), since most methods include identity extraction that relies on external lookups, leading to off-device transmission of webpage content or screenshots.

6.1.5 Findings Across Strategies

Phishing tactic diversity (**F1**) is uniformly weak across all strategies: none achieve high fulfillment and most are rated medium (71–100%), reflecting the absence of per-tactic evaluation. Benign page diversity (**F2**) is another major gap. Feature-based and hybrid approaches are almost entirely low, while similarity-based and identity-based show some improvement with a small share of high fulfillment (29% and 25%, respectively). Interpretability (**F3**) varies by approach. Identity-based methods stand out with 100% high fulfillment, while similarity-based and hybrid methods show a mix of high and medium fulfillment. In contrast, feature-based methods lag behind with no high fulfillment and most rated low. Evaluation transparency (**F4**) is more consistently addressed: all strategies report at least some standard metrics.

Security-related requirements remain more uneven. For adaptation to concept drift (**S1**), similarity-based methods achieve mostly medium fulfillment (86%), while feature-based methods perform poorly (69% low). Identity-based and hybrid methods show comparatively stronger adaptability. Resistance to active attacks (**S2**) is especially weak in feature-based approaches (74% low), whereas other strategies achieve a larger share of medium fulfillment. Privacy preservation (**S3**) shows the sharpest divide: feature-based methods often operate on the client side, yielding the highest proportion of strong results (43% high), while identity-based and most hybrid approaches score low due to transmitting screenshots or raw URLs to external servers.

6.2 Metric Fulfillment by Input sources, Deployment Modes, and Detection Modes

To provide a broader perspective, we analyze how requirement satisfaction varies across input sources (Table 5), deployment modes (Table 6), and detection modes (Table 7), highlighting the distinct strengths and limitations associated with these design choices.

6.2.1 Input Sources

Table 5 shows that URL-only methods perform well on privacy (**S3**, 69% high) but show low fulfillment for benign page diversity, interpretability, adaptation to concept drift, and resistance to active attacks (**F2**, **F3**, **S1**, **S2**; roughly 62–88% low). Detection methods that incorporate webpage content exhibit higher fulfillment for these four metrics, but their privacy drops sharply (**S3**, 67–90% low). Methods that rely only on external metadata (*U,M*) perform the worst overall for security, with all three metrics (**S1**, **S2**, **S3**) at 100% low. When metadata is combined with webpage content, security improves slightly but still lags behind other input configurations. Across all input variants, phishing tactic diversity (**F1**) remains consistently partial, while evaluation transparency (**F4**) is comparatively strong, with most methods rated medium or high across configurations.

Table 5: Metric fulfillment vs. input source. **C.** = Count, **U** = URL, **C** = Webpage Content, **M** = External Metadata. Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low.

Input	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
U	16	0	75	25	0	12	88	0	19	81	38	62	0	6	31	62	0	25	75	69	0	31
U, M	4	0	75	25	0	25	75	0	50	50	75	25	0	0	0	100	0	0	100	0	0	100
U, C	15	0	87	13	13	20	67	20	40	40	33	67	0	13	60	27	13	60	27	33	0	67
U, C, M	20	0	90	10	5	25	70	35	25	40	15	80	5	30	25	45	15	35	50	10	0	90

Table 6: Metric fulfillment vs. deployment mode (**C.** = Count). Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low.

Mode	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
Client-side	23	0	83	17	4	17	78	9	26	65	39	61	0	13	26	61	4	22	74	61	0	39
Server-side	32	0	84	16	6	22	72	25	31	44	25	72	3	19	41	41	12	47	41	12	0	88

6.2.2 Deployment Modes (Client-side vs. Server-side)

Client-side systems perform detection directly on the user's device (e.g., browser, email client), whereas server-side systems transmit features or content to a remote server for analysis. Table 6 shows that client-side systems perform notably better on privacy (**S3**), with 61% of methods reaching a high level of fulfillment, compared to only 12% for server-side methods. Phishing tactic diversity (**F1**) remains limited in both modes, with the majority of methods at medium and none at high. Server-side approaches, however, show comparatively stronger fulfillment for benign page diversity (**F2**), interpretability (**F3**), concept-drift adaptation (**S1**), and resistance to active attacks (**S2**). Evaluation transparency (**F4**) is consistently strong in both modes, with most methods achieving at least medium fulfillment. Overall, client-side systems trade stronger privacy for weaker fulfillment of most other requirements, while server-side systems achieve broader functionality and security at the cost of privacy.

6.2.3 Detection Modes (Real-time vs. Non-real-time)

A method is considered *real-time* if its end-to-end detection latency, including feature collection, is under 1 s. Table 7 shows

Table 7: Metric fulfillment vs. detection mode (**C.** = Count). Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low.

Mode	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
Real-time	24	0	75	25	0	17	83	8	25	67	50	50	0	12	25	62	4	33	62	54	0	46
Non-real-time	31	0	90	10	10	23	68	26	32	42	16	81	3	19	42	39	13	39	48	16	0	84

that non-real-time systems achieve comparatively higher fulfillment in benign page diversity (**F2**), interpretability (**F3**), concept-drift adaptation (**S1**), and resistance to active attacks (**S2**). In contrast, privacy preservation (**S3**) shows the opposite trend: over half of real-time methods (54%) achieve high privacy compared to only 16% of non-real-time systems. Evaluation transparency (**F4**) is consistently strong in both modes, with nearly all methods rated at least medium. Phishing tactic diversity (**F1**) remains generally at a medium level for both. Overall, real-time systems trade stronger privacy for weaker fulfillment in other requirements, whereas non-real-time systems offer relatively better interpretability, drift adaptability, and resistance to active attacks.

6.3 Other Evaluation Dimensions

The following tables extend our analysis to additional dimensions: reported accuracy (Table 8), publication year (Table 9), and citation count (Table 10). Among the three requirement satisfaction levels (high, medium, low), the gray shading marks the predominant one for each metric, making it easier to observe prevailing patterns and gaps in the literature. Below, we highlight these trends in detail for each dimension.

6.3.1 Reported Accuracy

Table 8 groups methods by the lowest reported accuracy of the primary model configuration when evaluated under functionality or security requirements. Accuracy often decreases in these settings, highlighting the trade-off between satisfying deployment-relevant criteria and maintaining benchmark performance. Benign page diversity (**F2**), interpretability (**F3**), and privacy preservation (**S3**) are mostly poorly satisfied across all bands. In the 90–100% range, adaptation to concept drift (**S1**) is usually low (44–69% low; at most 20% high), and resistance to active attacks (**S2**) is largely low or medium, with no more than 8% high. In contrast, the 80–90% band stands out: all studies achieve at least medium fulfillment for drift adaptation (**S1**) and active-attack resistance (**S2**) (33% high, 67% medium), while tactic coverage (**F1**) is uniformly medium. Interpretability (**F3**) is also stronger in this band than in others, with one-third high and another third medium. This pattern reflects the trade-off that methods addressing tactic diversity, adversarial evasions, or concept drift face in harder detection scenarios, leading to lower benchmark accuracy.

Table 8: Metric fulfillment vs. accuracy[1]. **Acc.** = Accuracy, **C.** = Count. Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low.

Acc. (%)	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
99–100	12	0	75	25	8	25	67	8	33	58	42	58	0	8	25	67	8	33	58	33	0	67
98–99	9	0	100	0	0	22	78	11	44	44	33	67	0	11	44	44	0	33	67	44	0	56
95–98	13	0	77	23	8	15	77	8	23	69	23	77	0	0	31	69	8	31	62	38	0	62
90–95	10	0	70	30	10	0	90	30	20	50	40	50	10	20	20	60	0	50	50	20	0	80
80–90	3	0	100	0	0	33	67	33	33	33	67	33	0	33	67	0	33	67	0	33	0	67
<80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N/A	8	0	100	0	0	38	62	38	25	38	0	100	0	50	50	0	25	25	50	25	0	75

[1] The "<80" accuracy band contains no papers. "N/A" indicates studies for which accuracy was not reported.

Table 9: Metric fulfillment vs. publication year (**C.** = Count). Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2. ●=High, ◐=Medium, ○=Low.

Year	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
2025	6	0	83	17	0	17	83	0	50	50	83	17	0	33	0	67	17	50	33	17	0	83
2024	14	0	86	14	0	21	79	36	14	50	43	50	7	21	36	43	14	43	43	29	0	71
2023	7	0	86	14	0	14	86	29	14	57	29	71	0	29	0	71	0	29	71	43	0	57
2022	7	0	71	29	14	14	71	14	29	57	0	100	0	0	29	71	29	14	57	29	0	71
2021	5	0	60	40	20	20	60	20	20	60	20	80	0	0	40	60	0	20	80	40	0	60
2020	6	0	100	0	0	33	67	0	17	83	33	67	0	0	83	17	0	50	50	33	0	67
2019	5	0	100	0	0	40	60	0	60	40	20	80	0	0	40	60	0	60	40	40	0	60
<2019	5	0	80	20	20	0	80	20	60	20	0	100	0	40	60	0	0	20	80	40	0	60

6.3.2 Publication Year

Table 9 shows that while most requirements remain weakly fulfilled, interpretability, evaluation transparency, resistance to active attacks, and adaptation to concept drift show signs of improvement over time. Phishing tactic diversity (**F1**) is only partially satisfied, with no works achieving high fulfillment, while benign page diversity (**F2**) and privacy preservation (**S3**) remain consistently low. Interpretability (**F3**) reached high levels in 14–36% of works between 2021–2024, though absent elsewhere except in a few influential pre-2019 papers. Evaluation transparency (**F4**) has steadily improved, with 2025 being the first year where the majority of works completely fulfill this metric. Resistance to active attacks (**S2**) is concentrated in recent years, with 14–29% of works at high levels in 2022, 2024, and 2025, while all earlier years show none. Concept-drift adaptation (**S1**) shows recovery after a gap in 2019–2022, with 21–33% of works achieving high levels since 2023. Overall, evaluation transparency shows consistent improvements, and interpretability, concept-drift handling, and active-attack resistance have improved in recent years, whereas F1, F2, and S3 remain persistently underaddressed.

6.3.3 Citation Count

As shown in Table 10, most metrics remain insufficiently satisfied across all citation ranges. Papers with more than

Table 10: Metric fulfillment vs. citation count (**C.** = Count). Citation source: Google Scholar. Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2.

#Citation	**C.**	**Functionality (F)**												**Security (S)**								
		F1 (%)			**F2** (%)			**F3** (%)			**F4** (%)			**S1** (%)			**S2** (%)			**S3** (%)		
		●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○	●	◐	○
>200	9	0	100	0	0	33	67	0	56	44	22	78	0	22	33	44	0	22	78	56	0	44
100–200	8	0	88	12	25	12	62	25	25	50	12	88	0	0	62	38	0	38	62	25	0	75
50–100	9	0	67	33	11	0	89	11	33	56	22	78	0	0	44	56	11	22	67	33	0	67
20–50	11	0	82	18	0	18	82	18	27	55	45	55	0	27	9	64	9	36	55	27	0	73
<20	18	0	83	17	0	28	72	28	17	56	39	56	6	22	33	44	17	50	33	28	0	72

●=High, ◐=Medium, ○=Low

200 citations show higher fulfillment for privacy preservation (**S3**), with 56% reaching high satisfaction, while phishing-tactic coverage (**F1**) and benign-page diversity (**F2**) remain largely unaddressed across all ranges. Adaptation to concept drift (**S1**) follows a non-monotonic trend, with 22–27% high satisfaction in the highest and some lower citation bands, but none in the mid-range. Papers with fewer than 50 citations achieve higher rates of high satisfaction for evaluation transparency (**F4**). For resistance to active attacks (**S2**), papers with fewer than 100 citations show 9–17% high satisfaction, while none of the papers with more than 100 citations reach high satisfaction. Overall, citation count does not consistently predict requirement satisfaction. Highly cited papers demonstrate stronger privacy preservation (**S3**), but other metrics either show no clear trend (**F1**, **F2**), non-monotonic patterns (**S1**), or higher fulfillment among lower-cited papers (**S2**, **F4**).

6.4 Statistical Significance Analysis

We performed chi-square tests on key dimensions, including detection strategies, deployment modes, and detection modes, to determine which differences vary significantly across method categories and which ones remain consistent. The tests confirm statistically significant differences across multiple metrics ($p < 0.05$). For detection strategy, differences in F2, F3, S1, and S2 meet the significance threshold. Deployment mode differences are significant for S2 and S3, and detection mode differences are significant for F4 and S3. For example, client-side systems provide significantly stronger privacy preservation than server-side systems (S3, $p = 0.0005$). All other PHILTER metrics (i.e., F1, F4, S3 under detection strategy; F1–F4, S1 under deployment mode; and F1–F3, S1–S2 under detection mode) show no statistically significant variation, indicating that these gaps are broadly shared across the evaluated approaches.

6.5 Key Findings

1. **Limited tactic diversity.** All 55 approaches fall short of high fulfillment: 84% are rated medium and 16% low. Because datasets lack phishing-strategy labels and studies omit per-tactic evaluation, it remains unclear which tactics are effectively detected and which are missed.
2. **Poor generalization on benign pages.** Most approaches (75%) are rated low, as evaluations often exclude low-reputation domains and legitimate pages with embedded brand elements. This lack of diversity raises the risk of elevated false positives in real-world settings.
3. **Underexplored interpretability.** A majority of studies (53%) provide no decision-level explanation, limiting transparency and trust. Interpretability is stronger in similarity-, identity-, and hybrid-based approaches, but remains weakest in feature-based methods.
4. **Limited drift adaptation.** Nearly half of methods (49%) show low fulfillment, lacking both adaptive mechanisms and temporally split evaluation. This raises concerns about robustness as phishing tactics evolve.
5. **Limited resistance to active attacks.** A majority of studies (55%) are rated low and another 36% medium, with very few achieving high fulfillment. They generally lack rigorous evaluation against evasive manipulations, leaving resilience to active attacks uncertain.
6. **Privacy gaps.** Two-thirds of methods (67%) are rated low because they transmit sensitive data off-device, such as full URLs, content, or screenshots. The few that achieve high privacy (33%) are mostly client-side systems, but these often neglect other deployment-critical needs.
7. **Complementary strengths remain underexplored.** PHILTER reveals that each detection strategy excels on some metrics while failing to fulfill others. For example, identity-based methods provide strong drift adaptation but violate privacy by sending page content externally, while similarity-based methods lack drift adaptation due to reliance on static reference sets. However, existing approaches do not leverage the complementary strengths of different strategies to satisfy all requirements simultaneously.

7 Recommendations

Our analysis highlights several targeted opportunities to strengthen phishing website detection methods:

1. **Enable tactic-aware evaluation.** None of the 55 studies reported per-tactic results, leaving it unclear which attacks detectors capture. We call for developing community-maintained, tactic-annotated phishing benchmarks with per-tactic performance evaluation to reveal blind spots and drive research on overlooked attack strategies.
2. **Improve benign-page generalization.** Most approaches overlook evaluation on diverse benign pages that trigger false positives in practice. We call for controlled inclusion of benign pages with legitimate brand elements and low-reputation regional sites in benign-page benchmarks.
3. **Enhance interpretability.** Most methods lack decision-level explanations, limiting transparency and trust. We call for standardized frameworks such as suspicious-element highlighting, visual overlays, and model-agnostic XAI that

work across detection approaches to support auditing, strengthen user trust, and maintain accuracy.

4. **Strengthen drift adaptation.** Current methods rarely account for concept drift, leaving long-term robustness uncertain. We call for time-aware evaluation protocols that train on historical data and test on future samples, along with research into adaptive mechanisms such as incremental learning, self-supervised adaptation, and stable feature design that maintain effectiveness against evolving attacker tactics.
5. **Strengthen robustness against active attacks.** Most studies offer only superficial evaluation against active attacks, including evasive manipulations and adversarial perturbations. We call for systematic testing across obfuscated HTML, delayed content, redirects, manipulated visuals, and crafted adversarial examples to expose weaknesses, and for research on attack-aware defenses that anticipate future attacks through adversarial example generation and training.
6. **Integrate privacy-preserving designs.** Many approaches expose privacy risks by transmitting sensitive user data externally. Research should explore lightweight client-side detection, federated learning, and anonymized feature sharing to preserve privacy without reducing effectiveness.
7. **Integrate Complementary Strengths.** Different detection strategies address isolated PHILTER requirements but also carry distinct weaknesses. We recommend strategically integrating complementary modules to address this tension. For example, a server-side identity module can detect emerging brand impersonations on crawled webpages and refresh the reference sets used by client-side similarity-based detectors. This integration combines the drift-adaptation strength of identity-based methods with the privacy-preservation strength of similarity-based methods without inheriting their weaknesses. Such modular integration offers a practical path toward satisfying all PHILTER requirements simultaneously.

8 Discussion

Limitations. Our evaluation is limited by the completeness and clarity of information reported in academic papers. As the analyzed works come from peer-reviewed, reputable venues, we assume their reported details are accurate. Because the assessment of research papers is inherently subjective, we use a structured codebook, LLM-assisted preliminary assessment, and independent expert review to promote consistency and reduce bias. LLMs assist with evidence extraction and preliminary assessments, but final decisions are based on independent expert validation, with disagreements resolved through discussion. These safeguards help mitigate LLM hallucinations and human subjectivity. While LLM behavior may evolve with model updates, transparent evidence citation and expert validation help ensure reproducibility and interpretability.

Future Work. Our findings highlight concrete directions for curating phishing benchmarks and evaluating deployed systems. Tactic-annotated phishing datasets can be constructed by enumerating common attack strategies (e.g., homographs, redirection, cloaking, QR-phishing) and collecting representative samples for each tactic. For benign pages, future work can construct diverse benign-page benchmarks by sampling domains across different reputation levels from publicly available resources (e.g., Tranco) and by crawling real-world benign samples. Their characteristics (e.g., legitimate brand elements, domain reputation) can be analyzed to ensure that the resulting datasets capture the necessary diversity and provide evaluation on unseen benign samples. Evaluating deployed systems on such datasets would reveal blind spots across both attack coverage (**F1**) and benign-page generalization (**F2**). This evaluation pipeline can then be extended to include other PHILTER requirements such as human-study based assessments for interpretability (**F3**), testing on temporally split data and real-world feeds to assess drift adaptation (**S1**), evaluations against adversarial attacks and attacker manipulations (**S2**), and checks on whether user-sensitive data leaves the user's device to assess privacy preservation (**S3**), allowing deployed systems to be evaluated against the full range of PHILTER metrics. Finally, PHILTER's qualitative and evidence-driven framework naturally accommodates additional dimensions by extending the codebook. For instance, practitioners and researchers can define high, medium, and low criteria for computational cost—an increasingly important requirement for emerging hybrid and agentic systems. PHILTER can then analyze reported computational footprints (e.g., cost, latency, resource usage) to assess how well an approach satisfies these criteria.

9 Conclusion

This study shows that benchmark accuracy often masks deployment-critical weaknesses in phishing website detection. By applying PHILTER to 55 academic approaches, we find that no method fulfills all functionality and security requirements, and that gains in robustness often come at the cost of reduced accuracy. The central open challenge is building phishing detection systems that maintain accuracy while satisfying all functionality and security requirements. Our recommendations outline concrete steps toward this goal, including guidance for curating realistic benchmarks, improving evaluation practices, and integrating complementary strengths across detection strategies to support deployment needs.

Acknowledgments

We thank the anonymous reviewers and our shepherd for their guidance and thoughtful feedback, which greatly improved our paper. This work was supported in part by NSF grants

CNS-2154507, OAC-2139358, CNS-2201465, and a DOD grant FA9550-23-1-0453.

Ethical Considerations

This work is a Systematization of Knowledge (SoK) based entirely on the analysis of publicly available, peer-reviewed academic papers. Our evaluation corpus contains no live phishing content, proprietary datasets, or user-generated data. We did not collect, store, or interact with active phishing or benign websites at any stage of the study.

Stakeholders. The primary stakeholders are the academic community, practitioners, and end users. Researchers and practitioners benefit from a structured evaluation of functionality and security of AI-based phishing detection studies. End users may indirectly benefit as future detection systems address identified gaps. The research team faced no exposure to harmful or disturbing content, since all analysis was limited to published literature.

Harms and Mitigations. Potential harms include (i) inadvertent mislabeling due to incomplete reporting or assessment errors, which could mislead practitioners or misrepresent prior work, and (ii) adversaries inferring weaknesses across broad classes of methods. We mitigate these risks by (a) grounding all labels in a public codebook with explicit criteria, (b) citing supporting evidence for each label, (c) validating and correcting all LLM-generated assessments through expert review, and (d) presenting results in aggregate, framing them as constructive requirements for defenders, while withholding raw data or tactic-specific exploit recipes that could be directly abused. We did not probe live systems and therefore implicated neither terms of service nor legal constraints.

Ethical Principles. In line with the Menlo Report, this study promotes *Beneficence* by advancing defenses against phishing; *Respect for Persons* by ensuring no personal data or user privacy was impacted; *Justice* by openly sharing evaluation resources with the community; and *Respect for Law and Public Interest* by adhering to legal norms and community best practices.

Decision. Given the negligible risks, strong mitigations, and significant benefits for the security community, we determined it is both appropriate and beneficial to conduct and publish this study.

Open Science

We make all artifacts supporting the main contributions of this paper publicly available at https://doi.org/10.5281/zenodo.18199829 to facilitate independent verification and reproducibility. The repository includes the PHILTER codebook, the implementation of the LLM-assisted preliminary assessment pipeline, LLM-generated preliminary assessments and expert-validated final assessments for all 55 analyzed papers, and the analysis scripts used to reproduce all tables and aggregate statistics reported in the paper.

References

[1] Sahar Abdelnabi, Katharina Krombholz, and Mario Fritz. Visualphishnet: Zero-day phishing website detection by visual similarity. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 1681–1698, 2020.

[2] Rahmad Abdillah, Zarina Shukur, Masnizah Mohd, and Ts Mohd Zamri Murah. Phishing classification techniques: A systematic literature review. *IEEE Access*, 10:41574–41591, 2022.

[3] Kalana Abeywardena, Jiawei Zhao, Lexi Brent, Suranga Seneviratne, and Ralph Holz. Triplet mining-based phishing webpage detection. In *2020 IEEE 45th Conference on Local Computer Networks (LCN)*, pages 377–380, 2020.

[4] Bhupendra Acharya and Phani Vadrevu. {PhishPrint}: Evading phishing detection crawlers by prior profiling. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3775–3792, 2021.

[5] Öznur Şifa Akçam, Adem Tekerek, and Mehmet Tekerek. Development of BiLSTM deep learning model to detect URL-based phishing attacks. *Computers and Electrical Engineering*, 123:110212, 2025.

[6] Mohammed M Alani and Hissam Tawfik. Phishnot: a cloud-based machine-learning approach to phishing url detection. *Computer Networks*, 218:109407, 2022.

[7] Ammar Almomani, Mohammad Alauthman, Mohd Taib Shatnawi, Mohammed Alweshah, Ayat Alrosan, Waleed Alomoush, and Brij B Gupta. Phishing website detection with semantic features based on machine learning classifiers: a comparative study. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 18(1):1–24, 2022.

[8] Yazan Ahmad Alsariera, Victor Elijah Adeyemo, Abdullateef Oluwagbemiga Balogun, and Ammar Kareem Alazzawi. AI meta-learners and extra-trees algorithm for the detection of phishing websites. *IEEE Access*, 8:142532–142542, 2020.

[9] Faisal S Alsubaei, Abdulwahab Ali Almazroi, and Nasir Ayub. Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics. *IEEE Access*, 12:8373–8389, 2024.

[10] Anti-Phishing Working Group. Phishing activity trends reports, 2025. Accessed: 2025-03-25.

[11] Giovanni Apruzzese and VS Subrahmanian. Mitigating adversarial gray-box attacks against phishing detectors. *IEEE Transactions on Dependable and Secure Computing*, 20(5):3753–3769, 2022.

[12] Daniel Arp, Erwin Quiring, Feargus Pendlebury, Alexander Warnecke, Fabio Pierazzi, Christian Wressnegger, Lorenzo Cavallaro, and Konrad Rieck. Dos and don'ts of machine learning in computer security. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3971–3988, 2022.

[13] Sultan Asiri, Yang Xiao, Saleh Alzahrani, and Tieshan Li. Phishingrtds: A real-time detection system for phishing attacks using a deep learning model. *Computers & Security*, 141:103843, 2024.

[14] Mahdi Bahaghighat, Majid Ghasemi, and Figen Ozen. A high-accuracy phishing website detection method based on machine learning. *Journal of Information Security and Applications*, 77:103553, 2023.

[15] Mohamed Bekkar, Hassiba Kheliouane Djemaa, and Taklit Akrouf Alitouche. Evaluation measures for models assessment over imbalanced data sets. *J Inf Eng Appl*, 3(10), 2013.

[16] Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Massimiliano Ciaramita, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, et al. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv:2212.08037*, 2022.

[17] Javier Castro, Justin Paine, and Rachael Truong. How cloudflare is using automation to tackle phishing head on, March 2025. Accessed: 2025-03-19.

[18] Fabrício Ceschin, Marcus Botacin, Albert Bifet, Bernhard Pfahringer, Luiz S Oliveira, Heitor Murilo Gomes, and André Grégio. Machine learning (in) security: A stream of problems. *Digital Threats: Research and Practice*, 5(1):1–32, 2024.

[19] Davide Chicco and Giuseppe Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1):6, 2020.

[20] Kang Leng Chiew, Ee Hung Chang, Wei King Tiong, et al. Utilisation of website logo for phishing detection. *Computers & Security*, 54:16–26, 2015.

[21] Ylona Chun Tie, Melanie Birks, and Karen Francis. Grounded theory research: A design framework for novice researchers. *SAGE open medicine*, 7:2050312118822927, 2019.

[22] Computing Research and Education Association of Australasia (CORE). Core conference rankings portal, 2025. Accessed: 2025-05-26.

[23] Wikipedia contributors. General data protection regulation, 2025. Accessed: 2025-08-22.

[24] Igino Corona, Battista Biggio, Matteo Contini, Luca Piras, Roberto Corda, Mauro Mereu, Guido Mureddu, Davide Ariu, and Fabio Roli. Deltaphish: Detecting phishing webpages in compromised websites. In *European Symposium on Research in Computer Security*, pages 370–388, 2017.

[25] Carlo Marcelo Revoredo da Silva, Eduardo Luzeiro Feitosa, and Vinicius Cardoso Garcia. Heuristic-based strategy for phishing prediction: A survey of url-based approach. *Computers & Security*, 88:101613, 2020.

[26] Dagstuhl. DBLP computer science bibliography. Accessed: 2025-02-12.

[27] Avisha Das, Shahryar Baki, Ayman El Aassal, Rakesh Verma, and Arthur Dunbar. Sok: a comprehensive reexamination of phishing research from the security perspective. *IEEE Communications Surveys & Tutorials*, 22(1):671–708, 2019.

[28] Google Developers. Google Safe Browsing API Reference, 2025.

[29] Yan Ding, Nurbol Luktarhan, Keqin Li, and Wushour Slamu. A keyword-based combination approach for detecting phishing webpages. *Computers & Security*, 84:256–275, 2019.

[30] Dinil Mon Divakaran and Adam Oest. Phishing detection leveraging machine learning and deep learning: A review. *IEEE Security & Privacy*, 20(5):86–95, 2022.

[31] Nguyet Quang Do, Ali Selamat, Ondrej Krejcar, Enrique Herrera-Viedma, and Hamido Fujita. Deep learning for phishing detection: Taxonomy, current challenges and future directions. *IEEE Access*, 10:36429–36463, 2022.

[32] Zuochao Dou, Issa Khalil, Abdallah Khreishah, Ala Al-Fuqaha, and Mohsen Guizani. Systematization of knowledge (sok): A systematic review of software-based web phishing detection. *IEEE Communications Surveys & Tutorials*, 19(4):2797–2819, 2017.

[33] Vincent Drury and Ulrike Meyer. Certified phishing: taking a look at public key certificates of phishing websites. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, pages 211–223, 2019.

[34] Egress. Must-know phishing statistics for 2025, March 2025. Accessed: 2025-03-25.

[35] Ayman El Aassal, Shahryar Baki, Avisha Das, and Rakesh M Verma. An in-depth benchmarking and evaluation of phishing detection research for security needs. *IEEE Access*, 8:22170–22192, 2020.

[36] Fang Feng, Qingguo Zhou, Zebang Shen, Xuhui Yang, Lihong Han, and JinQiang Wang. The application of a novel neural network in the detection of phishing websites. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15, 2024.

[37] Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next-generation agentic capabilities. Technical report, Google DeepMind, June 2025.

[38] Brij B Gupta, Krishna Yadav, Imran Razzak, Konstantinos Psannis, Arcangelo Castiglione, and Xiaojun Chang. A novel approach for phishing urls detection using lexical based machine learning in a real-time environment. *Computer Communications*, 175:47–57, 2021.

[39] Qingying Hao, Nirav Diwan, Ying Yuan, Giovanni Apruzzese, Mauro Conti, and Gang Wang. It doesn't look like anything to me: using diffusion model to subvert visual phishing detectors. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 3027–3044, 2024.

[40] Daojing He, Xin Lv, Shanshan Zhu, Sammy Chan, and Kim-Kwang Raymond Choo. A method for detecting phishing websites based on tiny-bert stacking. *IEEE Internet of Things Journal*, 11(2):2236–2243, 2023.

[41] Mingxing He, Shi-Jinn Horng, Pingzhi Fan, Muhammad Khurram Khan, Ray-Shine Run, Jui-Lin Lai, Rong-Jian Chen, and Adi Sutanto. An efficient phishing webpage detector. *Expert Systems with Applications*, 38(10):12018–12027, 2011.

[42] Mo Houtti, Abhishek Roy, Venkata Narsi Reddy Gangula, and Ashley Marie Walker. A survey of scam exposure, victimization, types, vectors, and reporting in 12 countries. *arXiv:2407.12896*, 2024.

[43] Internet Crime Complaint Center (IC3). 2023 internet crime report.

[44] Internet Crime Complaint Center (IC3). 2024 internet crime report.

[45] IEEE Digital Privacy Initiative. What is privacy-by-design and why it's important?, 2025.

[46] Ankit Kumar Jain and Brij B Gupta. Phishing detection: analysis of visual similarity based approaches. *Security and Communication Networks*, 2017(1):5421046, 2017.

[47] Sajjad Jalil, Muhammad Usman, and Alvis Fong. Highly accurate phishing url detection based on machine learning. *Journal of Ambient Intelligence and Humanized Computing*, 14(7):9233–9251, 2023.

[48] Fujiao Ji, Kiho Lee, Hyungjoon Koo, Wenhao You, Euijin Choo, Hyoungshick Kim, and Doowon Kim. Evaluating the effectiveness and robustness of visual similarity-based phishing detection models. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 3201–3220, 2025.

[49] Dina Jibat, Sarah Jamjoom, Qasem Abu Al-Haija, and Abdallah Qusef. A systematic review: Detecting phishing websites using data mining models. *Intelligent and Converged Networks*, 4(4):326–341, 2023.

[50] Lakshmana Rao Kalabarige, Routhu Srinivasa Rao, Ajith Abraham, and Lubna Abdelkareim Gabralla. Multilayer stacked ensemble learning model to detect phishing websites. *IEEE Access*, 10:79543–79552, 2022.

[51] Abdul Karim, Mobeen Shahroz, Khabib Mustofa, Samir Brahim Belhaouari, and S Ramana Kumar Joga. Phishing detection system through hybrid machine learning based on url. *IEEE Access*, 11:36805–36822, 2023.

[52] S Kavya and D Sumathi. Multimodal and temporal graph fusion framework for advanced phishing website detection. *IEEE Access*, 2025.

[53] Doowon Kim, Haehyun Cho, Yonghwi Kwon, Adam Doupé, Sooel Son, Gail-Joon Ahn, and Tudor Dumitras. Security analysis on practices of certificate authorities in the HTTPS phishing ecosystem. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, pages 407–420, 2021.

[54] Takashi Koide, Hiroki Nakano, and Daiki Chiba. Chatphishdetector: Detecting phishing sites using large language models. *IEEE Access*, 2024.

[55] Aditya Kulkarni, Vivek Balachandran, and Tamal Das. Phishing webpage detection: Unveiling the threat landscape and investigating detection techniques. *IEEE Communications Surveys & Tutorials*, 2024.

[56] Santosh Kumar Birthriya and Ankit Kumar Jain. A comprehensive survey of phishing email detection and protection techniques. *Information Security Journal: A Global Perspective*, 31(4):411–440, 2022.

[57] Jakub Lála, Odhran O'Donoghue, Aleksandar Shtedritski, Sam Cox, Samuel G Rodriques, and Andrew D White. PaperQA: Retrieval-augmented generative agent for scientific research. *arXiv:2312.07559*, 2023.

[58] Jehyun Lee, Peiyuan Lim, Bryan Hooi, and Dinil Mon Divakaran. Multimodal large language models for phishing webpage detection and identification. In *2024 APWG Symposium on Electronic Crime Research (eCrime)*, pages 1–13, 2024.

[59] Jehyun Lee, Pingxiao Ye, Ruofan Liu, Dinil Mon Divakaran, and Mun Choon Chan. Building robust phishing detection system: an empirical analysis. *NDSS MADWeb*, 2020.

[60] Wenhao Li, Selvakumar Manickam, Yung-Wey Chong, Weilan Leng, and Priyadarsi Nanda. A state-of-the-art review on phishing website detection techniques. *IEEE Access*, 2024.

[61] Yuexin Li, Chengyu Huang, Shumin Deng, Mei Lin Lock, Tri Cao, Nay Oo, Hoon Wei Lim, and Bryan Hooi. {KnowPhish}: Large language models meet multimodal knowledge graphs for enhancing {Reference-Based} phishing detection. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 793–810, 2024.

[62] Yukun Li, Zhenguo Yang, Xu Chen, Huaping Yuan, and Wenyin Liu. A stacking model using url and html features for phishing webpage detection. *Future Generation Computer Systems*, 94:27–39, 2019.

[63] Yun Lin, Ruofan Liu, Dinil Mon Divakaran, Jun Yang Ng, Qing Zhou Chan, Yiwen Lu, Yuxuan Si, Fan Zhang, and Jin Song Dong. Phishpedia: A hybrid deep learning based approach to visually identify phishing webpages. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3793–3810, 2021.

[64] Ruofan Liu, Yun Lin, Xiwen Teoh, Gongshen Liu, Zhiyong Huang, and Jin Song Dong. Less defined knowledge and more true alarms: Reference-based phishing detection without a pre-defined reference list. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 523–540, 2024.

[65] Ruofan Liu, Yun Lin, Xianglin Yang, Siang Hwee Ng, Dinil Mon Divakaran, and Jin Song Dong. Inferring phishing intention via webpage appearance and dynamics: A deep vision based approach. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1633–1650, 2022.

[66] Ruofan Liu, Yun Lin, Yifan Zhang, Penn Han Lee, and Jin Song Dong. Knowledge expansion and counterfactual interaction for {Reference-Based} phishing detection. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 4139–4156, 2023.

[67] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[68] Ameen R Mahmood and Sarab M Hameed. Review of smishing detection via machine learning. *Iraqi Journal of Science*, pages 4244–4259, 2023.

[69] Samuel Marchal and N Asokan. On designing and evaluating phishing webpage detection techniques for the real world. In *11th USENIX Workshop on Cyber Security Experimentation and Test (CSET 18)*, 2018.

[70] Sourena Maroofi, Maciej Korczyński, and Andrzej Duda. Are you human? resilience of phishing detection to evasion techniques based on human verification. In *Proceedings of the ACM Internet Measurement Conference*, pages 78–86, 2020.

[71] Changqing Miao, Jianan Feng, Wei You, Wenchang Shi, Jianjun Huang, and Bin Liang. A good fishman knows all the angles: A critical evaluation of Google's phishing page classifier. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 2486–2500, 2023.

[72] Microsoft. Safe links in Microsoft Defender for Office 365, 2025. Accessed: 2025-03-19.

[73] David Moher, Alessandro Liberati, Jennifer Tetzlaff, Douglas G Altman, Prisma Group, et al. Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement. *International journal of surgery*, 8(5):336–341, 2010.

[74] Bilal Naqvi, Kseniia Perova, Ali Farooq, Imran Makhdoom, Shola Oyedeji, and Jari Porras. Mitigation strategies against the phishing attacks: A systematic literature review. *Computers & Security*, 132:103387, 2023.

[75] Adam Oest, Yeganeh Safaei, Penghui Zhang, Brad Wardman, Kevin Tyers, Yan Shoshitaishvili, and Adam Doupé. PhishTime: Continuous longitudinal measurement of the effectiveness of anti-phishing blacklists. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 379–396, 2020.

[76] Adam Oest, Penghui Zhang, Brad Wardman, Eric Nunes, Jakub Burgis, Ali Zand, Kurt Thomas, Adam Doupé, and Gail-Joon Ahn. Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale. In *29th USENIX Security Symposium (USENIX Security 20)*, 2020.

[77] Chidimma Opara, Yingke Chen, and Bo Wei. Look before you leap: Detecting phishing web pages by exploiting raw url and html characteristics. *Expert Systems with Applications*, 236:121183, 2024.

[78] OpenAI. OpenAI o3 and o4-mini System Card. System card, OpenAI, June 2025.

[79] Thomas Kobber Panum, Kaspar Hageman, René Rydhof Hansen, and Jens Myrup Pedersen. Towards adversarial phishing detection. In *13th USENIX Workshop on Cyber Security Experimentation and Test (CSET 20)*, 2020.

[80] Rana Pourmohamad, Steven Wirsz, Adam Oest, Tiffany Bao, Yan Shoshitaishvili, Ruoyu Wang, Adam Doupé, and Rida A Bazzi. Deep dive into client-side anti-phishing: A longitudinal study bridging academia and industry. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, pages 638–653, 2024.

[81] Arvind Prasad and Shalini Chandra. Phiusiil: A diverse security profile empowered phishing url detection framework based on similarity index and incremental learning. *Computers & Security*, 136:103545, 2024.

[82] Y Bhanu Prasad and Venkatesulu Dondeti. Pdsmv3-dcrnn: A novel ensemble deep learning framework for enhancing phishing detection and url extraction. *Computers & Security*, 148:104123, 2025.

[83] Proofpoint. What is phishing? - definition, types of attacks & more, 2025. Accessed: 2025-03-25.

[84] Routhu Srinivasa Rao and Alwyn Roshan Pais. Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Computing and applications*, 31:3851–3873, 2019.

[85] Routhu Srinivasa Rao and Alwyn Roshan Pais. Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach. *Journal of Ambient Intelligence and Humanized Computing*, 11(9):3853–3872, 2020.

[86] S Remya, Manu J Pillai, BS Aparna, Somula Rama Subbareddy, and Yong Yun Cho. BGL-PhishNet: Phishing website detection using hybrid model-BERT, GNN, and LightGBM. *IEEE Access*, 13:47552–47569, 2025.

[87] Ozgur Koray Sahingoz, Ebubekir BUBE, and Emin Kugu. Dephides: Deep learning based phishing detection system. *IEEE Access*, 12:8052–8070, 2024.

[88] Ozgur Koray Sahingoz, Ebubekir Buber, Onder Demir, and Banu Diri. Machine learning based phishing detection from urls. *Expert Systems with Applications*, 117:345–357, 2019.

[89] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.

[90] Ido Sakazi, Edita Grolman, Yuval Elovici, and Asaf Shabtai. Stfl: Utilizing a semi-supervised, transfer-learning, federated-learning approach to detect phishing url attacks. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10, 2024.

[91] Manuel Sánchez-Paniagua, Eduardo Fidalgo Fernández, Enrique Alegre, Wesam Al-Nabki, and Víctor González-Castro. Phishing url detection: A real-case scenario through login urls. *IEEE Access*, 10:42949–42960, 2022.

[92] Michael Schesny, Nico Lutz, Thomas Jägle, Felix Gerschner, Marco Klaiber, and Andreas Theissler. Enhancing website fraud detection: A ChatGPT-based approach to phishing detection. In *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1494–1495, 2024.

[93] SCImago Research Group. Scimago journal & country rank, 2025. Accessed: 2025-05-26.

[94] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[95] Ivan Skula and Michal Kvet. A framework for preparing a balanced and comprehensive phishing dataset. *IEEE Access*, 2024.

[96] Station X. Phishing statistics: Trends, facts, and insights, 2025. Accessed: 2025-03-25.

[97] Colin Choon Lin Tan, Kang Leng Chiew, Kelvin SC Yong, Yakub Sebastian, Joel Chia Ming Than, and Wei King Tiong. Hybrid phishing detection using joint visual and textual identity. *Expert systems with applications*, 220:119723, 2023.

[98] Lizhen Tang and Qusay H Mahmoud. A deep learning-based framework for phishing website detection. *IEEE Access*, 10:1509–1521, 2021.

[99] Lucas Candeia Teixeira, Júlio César Gomes De Barros, Bruno José Torres Fernandes, and Carlo Marcelo Revoredo Da Silva. Catchphish: model for detecting homographic attacks on phishing pages. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08, 2022.

[100] Bram Van Dooremaal, Pavlo Burda, Luca Allodi, and Nicola Zannone. 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*, pages 1–10, 2021.

[101] RJ Van Geest, Giuseppe Cascavilla, Joris Hulstijn, and Nicola Zannone. The applicability of a hybrid framework for automated phishing detection. *Computers & Security*, 139:103736, 2024.

[102] Gaurav Varshney, Rahul Kumawat, Vijay Varadharajan, Uday Tupakula, and Chandranshu Gupta. Anti-phishing: A comprehensive perspective. *Expert Systems with Applications*, 238:122199, 2024.

[103] Mengli Wang, Lipeng Song, Luyang Li, Yuhui Zhu, and Jing Li. Phishing webpage detection based on global and local visual similarity. *Expert Systems with Applications*, 252:124120, 2024.

[104] Yanbin Wang, Weifan Zhu, Haitao Xu, Zhan Qin, Kui Ren, and Wenrui Ma. A large-scale pretrained deep model for phishing URL detection. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[105] Wei Wei, Qiao Ke, Jakub Nowak, Marcin Korytkowski, Rafał Scherer, and Marcin Woźniak. Accurate and fast url phishing detector: a convolutional neural network approach. *Computer Networks*, 178:107275, 2020.

[106] Colin Whittaker, Brian Ryner, and Marria Nazif. Large-scale automatic classification of phishing pages. In *NDSS*, volume 10, page 2010, 2010.

[107] Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pages 1–10, 2014.

[108] Guang Xiang, Jason Hong, Carolyn P Rose, and Lorrie Cranor. Cantina+ a feature-rich machine learning framework for detecting phishing web sites. *ACM Transactions on Information and System Security (TISSEC)*, 14(2):1–28, 2011.

[109] Xi Xiao, Wentao Xiao, Dianyan Zhang, Bin Zhang, Guangwu Hu, Qing Li, and Shutao Xia. Phishing websites detection via CNN and multi-head self-attention on imbalanced datasets. *Computers & Security*, 108:102372, 2021.

[110] Lixia Xie, Hao Zhang, Hongyu Yang, Ze Hu, and Xiang Cheng. A scalable phishing website detection model based on dual-branch tcn and mask attention. *Computer Networks*, 263:111230, 2025.

[111] Peng Yang, Guangzhen Zhao, and Peng Zeng. Phishing website detection based on multidimensional features driven by deep learning. *IEEE Access*, 7:15196–15209, 2019.

[112] Victor Zeng, Xin Zhou, Shahryar Baki, and Rakesh M Verma. Phishbench 2.0: A versatile and extendable benchmarking framework for phishing. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 2077–2079, 2020.

[113] Zilaing Zhang, Jinmin Wu, Ning Lu, Wenbo Shi, and Zhiquan Liu. Adaptpud: An accurate url-based detection approach against tailored deceptive phishing websites. *Computer Networks*, page 111303, 2025.

[114] Zhaoyu Zhou, Lingjing Yu, Qingyun Liu, Yang Liu, and Bo Luo. Tear off your disguise: Phishing website detection using visual and network identities. In *International Conference on Information and Communications Security*, pages 763–780, 2019.

[115] Rasha Zieni, Luisa Massari, and Maria Carla Calzarossa. Phishing or not phishing? A survey on the detection of phishing websites. *IEEE Access*, 11:18499–18519, 2023.

Appendix

Table 11: Distribution of selected papers across venues.

Conference (Rank)	#	Q1 Journal	#
USENIX Security (A*)	5	IEEE Access	11
ACM CCS (A*)	1	Computers & Security	7
NDSS (A*)	1	Expert Systems with Applications	5
ESORICS (A)	1	Computer Networks	4
IJCNN (B)	2	JAIHC	3
ARES (B)	1	ACM TOPS	1
COMPSAC (B)	1	Computer Communications	1
ICASSP (B)	1	Computers and Electrical Engineering	1
ICICS (B)	1	Future Generation Computer Systems	1
LCN (B)	1	IEEE Internet of Things Journal	1
eCrime (Unranked)	1	IEEE TDSC	1
		IJSWIS	1
		Journal of Information Secur. and Appl.	1
		Neural Computing and Applications	1

Listing 1: LLM Prompt Templates

```
# Evaluator Prompt
You are an expert in phishing detection research.
You are given:
1. A codebook for a specific evaluation metric
(definition, possible values, and criteria).
2. The full text of a phishing website detection paper.
Your task is to evaluate the method proposed in the paper
against the specified metric.
Instructions:
- Base your reasoning strictly on the paper content.
- Focus on technical and methodological sections.
- Assign one of the allowed values from the codebook.
- Justify your choice with explanation.
- Support it with direct evidence from paper.
Output Format:
- Return a JSON object using this schema:
{
  "value": "<value>", "why": "<Explanation>",
  "evidence": "<page, section, supporting quote>"
}

# Arbitrator Prompt
You are expert resolving disagreement between assessments
You are given:
1. A codebook for a specific evaluation metric
(definition, possible values, and criteria).
2. The full text of a phishing website detection paper.
3. Two assessments with conflicting verdicts.
Instructions:
- Review both assessments using the codebook and paper.
- Assign a value and provide rationale.
- Justify your conclusion with explanation.
- Support it with direct evidence from the paper.
Output Format:
- Return a JSON object using this schema:
{
  "value": "<value>", "why": "<Explanation>",
  "evidence": "<page, section, supporting quote>"
}
```

Listing 2: LLM Assessment Outputs

```
# Evaluator A (ChatGPT)
{
 "value": "Medium",
 "why": "VisualPhishNet employs a tactic-agnostic, visual
   similarity design that [truncated]",
 "evidence": "Page 6, Section 6.7 (Robustness and
   Security Evaluation): We studied 7 types of
   perturbations. [truncated]"
}

# Evaluator B (Gemini)
{
 "value": "High",
 "why": "The proposed method, VisualPhishNet, is tactic-
   agnostic by design, as it learns a generalizable
   visual profile [truncated]",
 "evidence": [ { "page": 2, "section": "2.1.2 Image-based
   similarity approaches", "quote": "Thus, our model and
   dataset collection [truncated]"}, ...]
}

# Arbitrator Output
{
  "value": "Medium",
  "why": "VisualPhishNet is tactic-agnostic by design and
    evaluates robustness against a variety [truncated]",
  "evidence": "Throughout Sections 6.7 and 6.8 (pp.8-10),
    the paper evaluates performance against [truncated]"
}
```

Table 12: LLM vs Expert agreement on PHILTER metrics.

Metric	Agreement
F1. Diversity of Phishing Tactics	48/55 (87.27%)
F2. Diversity of Benign Pages	49/55 (89.09%)
F3. Interpretability	51/55 (92.73%)
F4. Evaluation Transparency	54/55 (98.18%)
S1. Adaptation to Concept Drift	45/55 (81.82%)
S2. Resistance to Active Attacks	51/55 (92.73%)
S3. Privacy Preservation	47/55 (85.45%)
Overall	345/385 (89.61%)

Table 13: Per-paper agreement between LLM-generated labels and expert-validated labels across all metrics. Cells marked in gray indicate a disagreement. **L**: LLM, **H**: Human Expert. Functionality (F1–F4) and security (S1–S3) metric codes are defined in Table 2.

Ap.	**Method/Author**	**Functionality Metrics (F)**								**Security Metrics (S)**					
		F1		**F2**		**F3**		**F4**		**S1**		**S2**		**S3**	
		L	**H**	**L**	**H**	**L**	**H**	**L**	**H**	**L**	**H**	**L**	**H**	**L**	**H**
Feature	PDSMV3-DCRNN [82]	◐	◐	○	○	◐	◐	●	●	○	○	○	○	○	○
	Akçam et al. [5]	○	○	○	○	○	○	●	●	○	○	◐	◐	○	○
	Kavya et al. [52]	◐	◐	○	○	●	◐	●	●	●	●	●	●	●	●
	BGL-PhishNet [86]	◐	◐	◐	◐	○	○	●	●	○	○	○	○	○	○
	SPWDM [110]	◐	◐	○	○	◐	◐	◐	◐	○	○	◐	◐	○	○
	AdaptPUD [113]	◐	◐	○	○	○	○	●	●	●	●	●	◐	○	○
	Feng et al. [36]	◐	◐	○	○	○	○	●	●	○	○	○	○	○	○
	WebPhish [77]	◐	◐	○	○	○	○	◐	◐	○	○	○	○	○	●
	DEPHIDES [87]	◐	◐	○	○	○	○	◐	◐	◐	◐	○	○	○	●
	RNT-J [9]	◐	◐	○	○	◐	◐	◐	●	○	○	○	○	○	○
	Geest et al. [101]	○	○	○	○	○	○	●	●	○	○	◐	◐	○	●
	PhishingRTDS [13]	◐	◐	○	○	◐	○	◐	◐	○	○	◐	◐	○	○
	Schesny et al. [92]	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	STFL [90]	◐	◐	◐	◐	○	○	◐	◐	◐	◐	○	○	●	●
	LSD [51]	◐	◐	○	○	◐	○	◐	◐	○	○	○	○	○	●
	Bahaghighat et al. [14]	○	○	○	○	○	○	●	●	○	○	○	○	○	○
	Jalil et al. [47]	◐	◐	○	○	◐	◐	●	●	○	○	○	○	●	●
	PhishBERT [104]	◐	◐	○	○	○	○	◐	◐	○	○	○	○	○	○
	Tiny-BERT [40]	◐	◐	○	○	○	○	◐	◐	○	○	○	○	○	●
	Almomani et al. [7]	◐	◐	○	○	◐	◐	◐	◐	○	○	○	○	○	○
	Sánchez et al. [91]	◐	◐	○	○	○	○	◐	◐	◐	◐	○	○	●	●
	MLSELM [50]	○	○	○	○	○	○	◐	◐	○	○	○	○	○	○
	PhishNOT [6]	◐	◐	○	○	◐	◐	◐	◐	○	○	○	○	○	○
	POC [11]	◐	◐	◐	◐	○	○	◐	◐	○	○	●	●	○	○
	Gupta et al. [38]	◐	◐	○	○	◐	◐	●	●	○	○	○	○	●	●
	Mahmoud et al. [98]	○	○	○	○	○	○	◐	◐	◐	◐	○	○	○	○
	Xiao et al. [109]	○	○	○	○	○	○	◐	◐	○	○	○	○	○	●
	Wei et al. [105]	◐	◐	◐	◐	○	○	●	●	◐	◐	○	○	●	●
	LBET [8]	◐	◐	○	○	○	○	◐	◐	○	○	○	○	○	○
	Sahingoz et al. [88]	◐	◐	○	○	○	○	◐	◐	○	○	◐	◐	●	●
	MFPD [111]	◐	◐	○	◐	○	○	◐	◐	◐	◐	○	○	○	○
	Rao et al. [84]	◐	◐	○	○	◐	◐	◐	◐	○	○	◐	◐	○	○
	Li et al. [62]	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	○	○	●	●
	CANTINA+ [108]	◐	◐	○	○	◐	◐	◐	◐	●	●	○	○	○	○
	Whittaker et al. [106]	◐	◐	◐	○	◐	◐	◐	◐	●	●	○	○	●	●
Similarity	Wang et al. [103]	○	◐	○	○	●	●	●	●	●	◐	◐	◐	○	○
	PhishIntention [65]	●	◐	◐	●	●	●	◐	◐	●	◐	◐	●	○	○
	CatchPhish [99]	○	○	○	○	○	○	◐	◐	◐	○	◐	◐	○	●
	Phishpedia [63]	◐	◐	●	●	●	●	◐	◐	●	◐	◐	◐	○	○
	VisualPhishNet [1]	◐	◐	◐	◐	○	○	◐	◐	●	◐	◐	◐	○	○
	Abeywardena et al. [3]	◐	◐	○	○	○	○	◐	◐	●	◐	○	○	○	○
	DeltaPhish [24]	○	○	○	○	◐	◐	◐	◐	◐	◐	◐	◐	●	●
Identity	Lee et al. [58]	●	◐	◐	◐	●	●	◐	◐	◐	◐	◐	●	○	○
	ChatPhishDetector [54]	◐	◐	○	○	●	●	●	●	◐	◐	◐	◐	○	○
	Tan et al. [97]	○	◐	○	○	◐	●	◐	◐	●	●	◐	◐	○	○
	Chiew et al. [20]	◐	◐	●	●	●	●	◐	◐	◐	◐	○	○	○	○
Hybrid	PhiUSIIL [81]	◐	◐	○	○	◐	◐	●	●	●	●	◐	◐	○	○
	KnowPhish [61]	◐	◐	◐	○	●	●	◐	◐	◐	●	◐	◐	○	○
	PhishLLM [64]	●	◐	○	◐	●	●	◐	◐	●	●	◐	●	○	○
	DynaPhish [66]	◐	◐	●	◐	●	●	◐	◐	●	●	◐	◐	○	○
	Dooremaal et al. [100]	○	◐	◐	◐	○	○	◐	◐	◐	○	○	○	○	○
	BlackPhish [85]	◐	◐	○	○	◐	◐	●	●	●	◐	◐	◐	○	○
	PhishFencing [114]	◐	◐	○	○	○	○	◐	◐	◐	◐	◐	◐	●	●
	Ding et al. [29]	●	◐	○	○	◐	◐	●	●	○	○	◐	◐	○	○
	He et al. [41]	◐	◐	○	○	○	○	◐	◐	●	◐	○	○	◐	○

●=High, ◐=Medium, ○=Low