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

Review Article

The Evolution of Phishing Detection: From Static Signatures to Autonomous Generative Agents

***Ashwarya Singh*[1*], *Kalpana Mishra*[2]**

[1] Research Scholar, Dept. of Computer Science Engineering, JNCT, Bhopal, India
jpahwary1999@gmail.com
[2] Asst. Professor, Dept. of Computer Science Engineering, JNCT, Bhopal, India
kalpana.cse@jnctbhopal.ac.in

**Corresponding Author* *jpahwary1999@gmail.com*

DOI–10.55083/irjeas.2025.v13i04004

Abstract: Phishing is still one of the most common and harmful threats in cybersecurity. It is the main way that most data breaches happen. As a result, it is very important to create strong and quick detection methods. This paper offers an extensive literature analysis on the progression of phishing detection systems, outlining their development from initial static methods to the contemporary state-of-the-art. We start by looking at classic solutions, such blacklist-based and heuristic-based systems, and pointing out how they don't work well against new, zero-day threats. Next, we look at the big change that machine learning brought about, which made it possible to create more flexible solutions by using feature engineering from URLs and visual similarity analysis of webpages. A lot of attention is being paid to the rise of reference-based detection systems, which check the validity of web pages by comparing them to a database of real brands. We critically examine advanced dynamic systems such as DynaPhish, which try to automate knowledge base expansion, revealing their intrinsic fragility and reliance on inflexible logic. Lastly, we look at the cutting edge of phishing detection, which is defined by the use of generative AI and autonomous agents. We contend that Large Language Model (LLM)-powered agents, endowed with human-like reasoning, multi-modal analysis, and dynamic tool utilisation, constitute a possible remedy to the shortcomings of previous approaches. This study brings together the most important progress, points out ongoing

problems, and suggests that the future of phishing defence rests in making smart, self-driving systems that can think and change in real time to deal with the changing nature of modern phishing threats.

Keywords: Phishing Detection, Cybersecurity, Machine Learning, Autonomous Agents, Large Language Models (LLM), Generative AI, Reference-Based Detection are some of the words that come to mind.

I. INTRODUCTION

The widespread use of digital technologies has radically changed the way the world economy and society work, making it possible for people to engage with each other in ways that have never been possible before [6]. But this change to digital has also made people, businesses, and governments more vulnerable to a new type of advanced cyber danger [9]. Phishing is one of the most common and harmful types of social engineering. It tries to get sensitive information by pretending to be a trustworthy person in electronic conversations [1, 9]. Phishing attacks are no just annoying; they are the main cause of most cybersecurity problems. Industry studies show that they are responsible for almost 90% of all data breaches around the world [2, 9]. This means that stopping phishing is the most important thing for everyone in the cybersecurity field to do [100].

The threat is changing and getting worse all the time, with attacks becoming more complex, more frequent, and more varied [10]. Attackers use localised contexts, like India's Unified Payments Interface (UPI) [10, 11], and create fake websites that look quite real to target important industries like Banking, Financial Services, and Insurance (BFSI) [13, 14]. The economic effects are significant, as digital channels are increasingly being used to commit financial fraud [7, 16]. This combination of high-volume, multi-platform, and technologically advanced threats shows a major flaw in current security systems [18].

Phishing detection has changed over time in response to attacks that are getting more complicated. Early defences used static, signature-based tactics that were easy to use but were rapidly outsmarted by enemies. Machine learning brought about a new level of flexibility, allowing systems to learn from data and find strange patterns in URLs and webpage content. More lately, reference-based detection has become more popular. This method checks the legitimacy of a webpage by comparing its visual and structural identity to that of a known valid page.

This research offers a critical evaluation of this evolutionary trajectory. We will examine the fundamental methodology in phishing detection, ranging from early blacklist and heuristic-based approaches to contemporary machine learning and reference-based systems. A major goal will be to find the built-in problems that each generation of technology has, especially the problems of scalability, flexibility, and not being able to deal with new threats. Next, we look at the new idea of using generative artificial intelligence (GAI) and autonomous agents that are powered by Large Language Models (LLMs) as a way to change things. This analysis seeks to illustrate that the advancement of phishing detection represents a distinct path towards enhanced intelligence

and autonomy, ultimately resulting in systems capable of emulating human cognitive functions to offer a more robust defence against the continually developing threat of phishing.

II. Early and traditional ways to find phishing

Simple, rule-based systems were the first line of defence against phishing. These early methods can be grouped into two main groups: blacklist-based methods and heuristic-based content filtering. Even if they offered some security, their static and reactive nature was not enough to keep up with a changing threat scenario.

A. Methods Based on Blacklists

Keeping a collected list of the addresses of bad websites is the most straightforward way to ban them. This is how blacklist-based detection works: it checks a URL against a list of known phishing domains. If it finds a match, it blocks access [12]. Google Safe Browsing, PhishTank, and OpenPhish are some of the most well-known services that use this paradigm. They mostly rely on community reporting and manual verification to fill their databases [15, 17].

The fundamental benefit of blacklisting is that it is quite accurate; if a URL is on a well-maintained blacklist, it is almost always malicious, which means that there are very few false positives. But this strategy has a major and basic flaw: it only reacts [23]. Cybercriminals may quickly and cheaply register new domains and start phishing campaigns. This means that there is a big time gap between when a new assault starts and when it is added to a blacklist. During this time, users are fully defenceless [12]. Industry investigation has demonstrated that this reactive characteristic significantly constrains the effectiveness of blacklists against emerging phishing websites in the wild [15, 18]. Attackers take advantage of this issue even more by employing URL shortening services and domain generating algorithms to quickly make disposable attack vectors that make it impossible to keep blacklists up to date by hand [14].

B. Heuristic and content-based filtering

Heuristic-based solutions were made to get over the problems with simple blacklisting. These algorithms look for suspicious traits or "heuristics" [12] in the URL, email, or webpage instead of just looking at a list of known malicious URLs. This includes looking for strange things in the

URL structure (like too long, using IP addresses, or misleading subdomains), looking for signs of spoofing in email headers, and looking for keywords that are often used in phishing emails (like "verify your account," "urgent," and "password") [22].

These methods were a step towards a more proactive defence because they could flag a suspect message even if they had never seen its identical signature before. Khonji et al. conducted a thorough survey that outlines the diverse range of parameters that can be integrated into heuristic models, addressing everything from the linguistic characteristics of URLs to the structural composition of HTML webpages [12, 16].

But heuristic-based approaches are known for having a lot of false positives. A real email from a bank could include the same keywords as a phishing email, which could cause it to be wrongly reported. Also, attackers rapidly learnt how to get around these filters. They used advanced obfuscation methods, like putting harmful material in photos or utilising complicated JavaScript to mask their true purpose from static analysis engines [24]. This game of cat and mouse showed that heuristics

could help blacklists, but they weren't a good answer on their own. They did not have the semantic comprehension and contextual reasoning necessary to distinguish between authentic urgency and malevolent deception [25].

II. THE INCREASE OF MACHINE LEARNING IN PHISHING DETECTION

The inherent rigidity of static rules and heuristics underscored the necessity for more sophisticated and dynamic defence systems. This led to the widespread use of machine learning (ML), which let systems learn the complicated patterns of phishing assaults from large datasets instead of having to rely on rules that were written by hand. Machine learning (ML) methods have mostly looked at two main things: getting predictive features from URLs and looking at the visual content of webpages for evidence of impersonation.

C. Feature Engineering from URLs

Researchers started using machine learning to classify URLs in real time based on their inherent properties because they knew that the URL itself was a rich source of information [17]. ML models break down the URL into a set of predictive attributes, while blacklist methods just look for a match between the URL and a string. These can be things like structural features (how long the domain is, how many subdomains it has, or whether it has special characters like "@" or "-"), word-based features (whether it has brand names or sensitive phrases), and more advanced network-based features (how old the domain is, WHOIS information).

A significant progress in this domain was the utilisation of Natural Language Processing (NLP) methodologies to examine the semantic content of URLs. Sahingoz et al. created a system for real-time detection that used machine learning to construct feature sets using Word Vectors and other NLP-based features taken straight from URLs [16, 18]. Their approach could find harmful URLs more accurately than standard heuristic methods by modelling URL components as vectors in a high-dimensional space. This allowed them to capture subtle semantic correlations [18]. This method worked especially well for finding new phishing sites since the model could use what it had learnt about harmful URLs to find new ones without having seen the specific domain previously. These URL-only approaches are strong, but they have some built-in limits. For example, they can't stop attacks that come from compromised-but-legitimate domains or those that employ content that seems like it belongs on a page with a safe URL.

D. Detection Based on Visual Similarity and Reference

Visual deception is a key part of phishing. Attackers carefully create webpages that "purport to act on behalf of a legitimate third party with the intent of misleading viewers" [11, 14]. This resulted in the creation of visual similarity-based detection, a method that checks the authenticity of a webpage by comparing its look to that of the real brand it says it represents.

This method was a big step forward because it didn't just look at the delivery vector (URL) but also at the payload (the webpage). The Phishpedia system was the first to use a hybrid deep learning method to visually find phishing websites [13, 19]. It works on the idea that phishing sites typically look different from real ones. Phishpedia uses two deep learning models: one for accurately

recognising logos and another for recognising brands. This creates a reference-based system that examines if the logo found fits the brand identity that was claimed [13, 17].

The PhishIntention architecture added a Credential Requirement Page (CRP) classifier [14, 11] on top of this base. The main goal of most phishing attacks is to steal sensitive user credentials [14, 12]. This is why this was done. PhishIntention uses a screenshot and a study of the page's HTML using its CRP

classifier to figure out not only the brand being impersonated but also the bad purpose of the page. These reference-based methods showed a big improvement in detection accuracy by focussing on the real visual identity of a brand. Their effectiveness is fundamentally contingent upon the quality and comprehensiveness of their foundational information base regarding protected brands.

III. PROGRESS AND CONSTRAINTS IN DYNAMIC REFERENCE-BASED SYSTEMS

Reference-based detection was a great method, but its first uses had a serious difficulty with scalability. The efficiency of any such system is "heavily contingent upon the comprehensiveness of its protected brand list," which must be regularly updated to be effective [9, 17, 15]. Because new online companies and services are popping up so quickly, manually curating a knowledge base is not a long-term or proactive task. This difficulty led to the creation of dynamic systems that automatically extend their knowledge base.

A. Dynamic Knowledge Expansion: The DynaPhish Case

The Dynaphish system was designed primarily to solve the problem of keeping the knowledge base up to date [9, 16]. It is a big step forward in reference-based identification since it uses external tools like the Google Search engine and the Google Logo Detector to automatically add new information to its knowledge base [9, 17].

DynaPhish's operational logic is meant to work as a simpler verification method. When it thinks it might be on a phishing page, it initially tries to get a logo. Then, it uses a service that can find logos to figure out whose brand it belongs to. When you type this brand name into an online

search engine, it gives you the best results. Finally, it goes to these top-ranked websites, gets their logos, and compares them to the logo on the page that was thought to be suspicious. If a good match is identified, the brand is added to its knowledge base, and the original page is put into a category based on domain matching [25]. Theoretically, this automated pipeline for expanding and verifying knowledge lets the system learn about new brands on the fly, which makes it more flexible than older systems that didn't change.

B. Serious Problems with Programmatic Dynamic Approaches

Even while the DynaPhish system has a unique design, a close look at it shows that its strict, programmatic logic is weak and likely to break down in real life. It works because of a chain of dependencies, and if there is a mistake at any one phase, the whole operation can fail.

External APIs are important: The whole DynaPhish process depends on the correctness of the Google Logo Detector API [25, 26]. If this service gives the wrong brand name or doesn't find a brand at all, the whole procedure of searching and checking is pointless, which is a false negative. This one

point of failure shows how weak it is to depend on a black-box external service that doesn't have a way to reason or fix mistakes.

Incomplete Brand Representation: The system has a lot of trouble with different versions of brand logos. The study in the original dissertation found that DynaPhish only accurately detected two of the 103 AT&T phishing samples [26]. The system only got the most recent official logo from the AT&T website, which is why this happened. It didn't pass the representation validation for all the other samples that employed AT&T logos that were older but still well-known [26]. This shows that a basic threshold for matching logos isn't enough to show the full range of a brand's visual identity.

Too stringent filtering and heuristics: DynaPhish uses a series of filters to improve its search results. For example, it won't show sites that are on a prohibited list. These strict heuristics are meant to make things more accurate, yet they can really make things worse. The system for Instagram didn't find 113 out of 119 phishing samples because its filters automatically left out search results that included the real "instagram.com" domain. This meant that it could never find

the right reference logos [21–24]. This shows how strict regulations might make it harder to find things and learn new things.

Technical and Evasion Failures: DynaPhish's automatic online driver for getting logos can be stopped by normal web security procedures. When trying to check the brand "Bitkub," for instance, security verification pages (like Cloudflare) stopped the system from getting to the target websites and getting reference logos. This caused all samples of that brand to not be detected at all [23, 24].

These failures show that while automating knowledge expansion is an important step forward, systems like DynaPhish that use deterministic, step-by-step logic aren't strong enough or flexible enough to handle the intricacies of the current web. We need a smarter and more adaptable way to do things.

IV. THE NEW PARADIGM: GENERATIVE AI AND AUTONOMOUS AGENTS

The fragility of programmatic dynamic systems underscores a critical deficiency: the lack of authentic logic and flexibility. The most recent change in how to find phishing attempts tries to fill this gap by using the game-changing abilities of generative artificial intelligence (GAI) and autonomous agents. This method goes beyond static algorithms and tries to make systems that can think, plan, and interact with data in a way that is similar to how people think.

A. Establishing the Fundamental Technologies

Generative AI is a type of machine learning model that can learn the underlying statistical patterns of a huge training corpus and then use that knowledge to make new, synthetic artefacts like text, images, and audio [17, 18]. OpenAI's GPT series is one of the most well-known examples of GAI [25].

Autonomous agents are software programs that can work on their own, see what's going on around them, and do things to reach certain goals without being told what to do by a person [18, 188]. The relationship between GAI and autonomous agents is very strong. LLMs give agents the cognitive "engine" or "brain" that lets them grasp natural language, think about difficult problems, and plan out steps to take [10].

B. The Basics of LLM-Powered Agents

Recent advances in AI research have set the stage for the development of advanced LLM-powered agents. Park et al. came up with the idea of Generative Agents, which showed that computational beings driven by an LLM might operate like real people in an interactive setting [19, 12]. These agents retain memories of their experiences, contemplate them to develop advanced insights, and utilise these memories for future planning [19].

From a technological point of view, frameworks like ReAct (Reasoning and Acting), which Yao et al. came up with, have proven very important [24]. ReAct demonstrated that LLMs could enhance performance on intricate tasks by producing interconnected reasoning traces and task-specific actions [24, 20]. This lets the model make, keep, and change high-level plans while also getting new information from other sources, such a search engine [24].

Additionally, systems such as HuggingGPT have illustrated the capability of employing a large language model (LLM) as a central controller to oversee and coordinate a wide range of specialised AI models [20, 21]. This method lets the LLM assign jobs to the best model, like finding objects in an image or recognising voice, which makes it easier to solve hard, multi-modal problems [198]. Other studies have concentrated on creating specialised datasets and tuning approaches, including AgentInstruct and AgentFLAN, to improve the overall agent-like functionalities of LLMs [22, 23].

C. Using Autonomous Agents to Find Phishing

The ideas that guide these agentic frameworks provide a straightforward remedy for the shortcomings evident in systems such as DynaPhish. A self-directed, agent-based method for finding phishing changes the challenge from one of strict, programmatic validation to one of flexible, goal-oriented research.

An LLM agent can be given a high-level goal, like "Find the brand associated with this webpage and make sure it's real." The agent can then choose which tools to use and in what order, using a toolkit that includes web search, image search, and vision analysis. For instance, if the first analysis of the language is unclear, it can choose to look for an image of the logo. If it comes across a different version of a logo, it can come up with new search terms like "old AT&T logos" to get more complete information, which fixes the representation failure encountered in DynaPhish. The system is much more resistant to confusion and surprise problems, such security verification sites, because it can think and plan ahead.

The KnowPhish system is a first step in this approach; it uses LLMs to make "oneshot" brand predictions based on the HTML content of a webpage [4,10]. But it only uses HTML, which is a big problem because it doesn't take into account the rich visual information in logos, page layout, and general design, which is important for accurate detection [2]. The source dissertation suggests a real agent-based system that would use vision models to evaluate screenshots combined with text and HTML analysis. This all-encompassing method, which combines the reasoning and planning abilities of frameworks like ReAct with inputs from several sources, is similar to how an expert would gather and combine facts to reach a conclusion. It replaces rigid, hard-coded logic with flexible, smart thinking. This is a big step ahead in the search for a phishing detection system that can really adapt.

V. CONCLUSION

There has been a constant technological arms race in the fight against phishing. This review has followed the development of detection methods, starting with static blacklists and heuristics that were easy to get around. Machine learning made systems more flexible by letting them learn from the characteristics of harmful URLs and the visual content of online sites. This led to the creation of reference-based detection, a strong method that nonetheless had trouble keeping up with the huge and never-ending task of keeping a complete list of real brands.

Dynamic systems like DynaPhish tried to fix this problem by automating it, but as we have seen, their strict, programmatic logic created new sites of failure, showing that we need more advanced reasoning skills. These systems are weak because they can't handle changes to logos, they are vulnerable to restricted heuristics, and they can't use standard online security features. All of these problems point to the same conclusion: a fixed algorithm can't beat a smart and changing enemy.

The new idea of autonomous agents powered by Large Language Models is the next phase in this evolution. We can get past the problems with prior methods by giving detecting systems the ability to think, plan, and interact with external tools and multi-modal information in real time. An LLM agent can do what a human cybersecurity specialist does when they investigate, but it can do it much faster and on a much larger scale. This method promises to make a defence that is stronger, more flexible, and smarter, able to deal with the new and confusing phishing attempts of today. There are still problems with the cost of computing and the time it takes to run operations, but the results of this research strongly suggest that the future of effective phishing detection lies in the continuous development and improvement of autonomous, reasoning agents.

REFERENCES

[1] M. Jakobsson and S. Myers, Eds., *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft.* Hoboken, NJ, USA: John Wiley & Sons, 2006.

[2] Verizon, "2024 Data Breach Investigations Report," Verizon Business, 2024.

[3] Ministry of Electronics & Information Technology, "Digital India Programme Annual Report," Government of India, 2023.

[4] CERT-In, "India Ransomware Report 2023," Indian Computer Emergency Response Team, Ministry of Electronics & Information Technology, Government of India, 2024.

[5] National Payments Corporation of India, "Fraud and Risk Management in UPI," NPCI Publications, 2023.

[6] R. Basnet and S. Mukkamala, "A survey of phishing and its detection techniques," in *Proc. 46th Annual Southeast Regional Conference,* 2008, pp. 1–6.

[7] CloudSEK, "Digital Risk Monitoring Report: BFSI Sector," CloudSEK XVigil, 2023.

[8] Reserve Bank of India, "Annual Report on Banking Frauds," RBI Press, 2024.

[9] R. Liu, Y. Lin, Y. Zhang, P. H. Lee, and J. S. Dong, "Knowledge expansion and counter-factual interaction for reference-based phishing detection," in *Proc. 32nd USENIX Security Symposium (USENIX Security 23),* 2023, pp. 4139–4156.

[10] Y. Li *et al.,* "Knowphish: Large language models meet multimodal knowledge graphs for enhancing reference-based phishing detection," *arXiv preprint* arXiv:2403.02253, 2024.

[11] C. Whittaker, B. Ryner, and M. Nazif, "Large-scale automatic classification of phishing pages," in *Proc. Network and Distributed System Security Symposium (NDSS),* 2010.

[12] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: A literature survey,"

IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091–2121, 2013.
[13] Y. Lin *et al.*, "Phishpedia: A hybrid deep learning based approach to visually identify phishing webpages," in *Proc. 30th USENIX Security Symposium (USENIX Security 21)*, 2021,
pp. 3793–3810.
[14] R. Liu, Y. Lin, X. Yang, S. H. Ng, D. M. Divakaran, and J. S. Dong, "Inferring phishing intention via webpage appearance and dynamics: A deep vision based approach," in *Proc. 31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1633–1650.
[15] S. Bell and P. Komisarczuk, "An analysis of phishing blacklists: Google Safe Browsing, OpenPhish, and PhishTank," in *Proc. Australasian Computer Science Week Multiconference*, 2020, pp. 1–11.
[16] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.
[17] D. Baidoo-Anu and L. O. Ansah, "Education in the era of generative artificial intelligence (AI): Understanding the potential benefits of ChatGPT in promoting teaching and learning," *Journal of AI*, vol. 7, no. 1, pp. 52–62, 2023.
[18] T. Bösser, "Autonomous agents," in *International Encyclopedia of the Social & Behavioral Sciences*, N. J. Smelser and P. B. Baltes, Eds. Pergamon, 2001, pp. 1002–1006.
[19] J. S. Park *et al.*, "Generative agents: Interactive simulacra of human behavior," in *Proc. 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*, 2023.
[20] Y. Shen *et al.*, "HuggingGPT: Solving AI tasks with ChatGPT and its friends in Hugging Face," *Advances in Neural Information Processing Systems*, vol. 36, 2024.Z. Yang et al., "AppAgent: Multimodal agents as smartphone users," arXiv preprint
[21] arXiv:2312.13771, 2023.
[22] Z. Chen *et al.*, "Agent-Flan: Designing data and methods of effective agent tuning for large language models," *arXiv preprint* arXiv:2403.12881, 2024.
[23] A. Zeng *et al.*, "AgentTuning: Enabling generalized agent abilities for LLMs," *arXiv preprint* arXiv:2310.12823, 2023.
[24] S. Yao *et al.*, "ReAct: Synergizing reasoning and acting in language models," *arXiv preprint* arXiv:2210.03629, 2022.

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

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

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

Cite this Article

Ashwarya Singh, Kalpana Mishra. The Evolution of Phishing Detection: From Static Signatures to Autonomous Generative Agents. International Research Journal of Engineering & Applied Sciences (IRJEAS). 12(4), pp. 38-47, 2025.10.55083/irjeas.2025.v13i04004