

REVIEW ARTICLE

Phishing Detection Techniques: A Review of Rule-Based, Machine Learning and Deep Learning Approaches

Mr. B. S. Bukka*

Assistant Professor, Department of Computer Science, Sangameshwar College (Autonomous), Solapur, Maharashtra, India

* Corresponding author

OPEN ACCESS

This article is licensed under CC BY-NC 4.0.

**ARTICLE HISTORY**

Received: 23 Sep 2026

Accepted: 05 Oct 2026

Published: 14 Oct 2026

CORRESPONDING AUTHOR

Mr. B. S. Bukka, Department of Computer Science, Sangameshwar College (Autonomous), Solapur, Maharashtra, India.

HOW TO CITEBukka, B. S. (2026). Phishing Detection Techniques: A Review of Rule-Based, Machine Learning and Deep Learning Approaches. *Global Journal of Advanced Science and Technology*, 1(1), 17–21.

© 2026 The Author(s).

ABSTRACT

Phishing, the use of fake websites and messages to trick people into revealing passwords, bank details or one-time passwords, remains one of the most common ways in which online accounts are compromised. In India, the spread of mobile banking and UPI payments has made SMS and messaging-app phishing a daily risk for ordinary users. This review examines the main techniques developed to detect phishing websites and links. It covers list-based methods such as blacklists, heuristic and rule-based methods, content and visual-similarity methods, machine learning on features extracted from URLs and web pages, and deep learning models that learn directly from the characters of a URL. For each family it discusses how the method works, what it requires and where it fails. The literature shows that blacklists are fast and precise but miss newly created sites, that machine learning on lexical URL features detects most phishing links with little computation, and that deep learning models perform best when large training sets are available. All approaches face the same problems: attackers adapt quickly, public datasets age, and reported accuracies are hard to compare. The review concludes that layered defences, combining lists, lightweight classifiers on the user's device and user awareness, are most effective, and it identifies the need for datasets and models suited to Indian languages and SMS-based phishing.

Keywords: Phishing; Cybersecurity; URL classification; Machine learning; Deep learning; Blacklists

1. Introduction

Most people in India have received a message like this: your electricity will be disconnected tonight, your bank account has been blocked, or your parcel is waiting at the post office, followed by a link. The link leads to a page that looks like the real one and asks for a login, a card number or a one-time password. This is phishing, and it works because it attacks people rather than computers.

Phishing exploits the gap between what a website looks like and what it is. Dhamija et al. (2006) showed participants a set of genuine and fake websites and found that the best fake site fooled 90 per cent

of them; many ignored the browser's security indicators altogether. Because user training alone cannot close this gap, a large body of research has developed automatic methods to detect phishing links and pages.

This review summarises that research. It follows the classification used in the survey by Khonji et al. (2013), updated to include the deep learning methods that have appeared since, and discusses which approaches suit the conditions of Indian users, who encounter phishing mostly through SMS and messaging applications on mobile phones.

2. How phishing works

A phishing attack has three parts: a lure, usually an email, SMS or social media message that creates urgency; a link to a fake site; and the fake site itself, which imitates a trusted brand and collects the victim's information. Attackers register domains that resemble real ones, use URL shorteners to hide the destination, host pages on compromised legitimate websites, and increasingly obtain valid HTTPS certificates so that the padlock icon gives no warning. Each detection method targets one or more of these parts.

3. Detection approaches

Figure 1 groups the detection methods into five families.

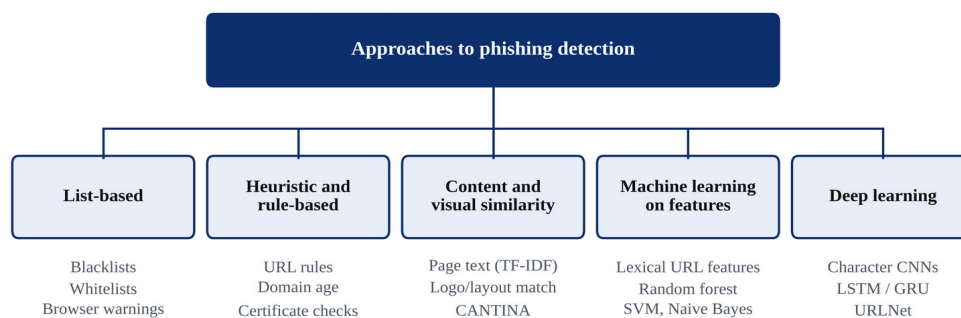


Figure 1. Families of phishing detection approaches.

3.1 List-based methods

Blacklists record URLs and domains already known to be phishing sites; browsers and email services check every link against them. They are fast and almost never wrong, but they can only block what someone has already reported. Sheng et al. (2009) tested several blacklists against newly reported phishing sites and found that most blacklists caught only a small share of them at the moment they first appeared, although coverage improved over the following hours. Since most phishing sites are active for a short time, this delay matters. Whitelists take the opposite approach, trusting only known sites, but are too restrictive for general browsing.

3.2 Heuristic and rule-based methods

Heuristic methods check for features that phishing sites often share: an IP address in place of a domain name, an unusually long URL, an @ symbol, many sub-domains, a recently registered domain, or a login form that sends data to a different domain. Rules can detect new sites that are not yet on any list. Their weakness is that legitimate sites sometimes break the same rules, and attackers soon learn which features to avoid.

3.3 Content and visual-similarity methods

Content-based methods examine the page itself. CANTINA, proposed by Zhang et al. (2007), extracts the most characteristic words on a page using TF-IDF weighting and searches for them on a search engine; if the page's own domain does not appear among the top results, the page is probably imitating another site. Visual-similarity methods compare the appearance of a page, its logo or layout, with those of frequently targeted brands. These methods catch convincing copies but require the page to be downloaded and analysed, which is slower than checking a URL.

3.4 Machine learning on extracted features

Machine learning methods learn the combination of features that separates phishing from legitimate sites. Ma et al. (2009) showed that lexical features of the URL, together with host-based features such as registration and hosting details, allowed a classifier to identify malicious URLs with high accuracy without visiting the page. Mohammad et al. (2014) used a self-structuring neural network on website features, and Abdelhamid et al. (2014) applied associative classification, which produces readable rules as well as predictions. Sahingoz et al. (2019) built a classifier using natural language processing features of the URL alone and reported accuracy close to 98 per cent with a random forest. Table 1 lists the URL features most often used.

Table 1. URL features commonly used by phishing classifiers.

Feature	Example	Why it signals risk
IP address as host	http://203.0.113.5/login	Genuine services rarely use raw IPs
URL length	Very long paths and queries	Used to hide the real domain
@ symbol or “//” in path	bank.com@evil.example	Browser ignores text before @
Number of sub-domains	secure.login.bank.example.xyz	Imitates a trusted brand name
Hyphens and look-alike words	sbi-kyc-update.example	Brand name combined with urgency
Shortened URL	Links via URL shorteners	Hides the destination
Domain age	Registered days earlier	Phishing domains are short-lived

Examples are illustrative. Sources: Ma et al. (2009); Mohammad et al. (2014); Sahingoz et al. (2019).

3.5 Deep learning

Feature-based classifiers depend on the features their designers choose. Deep learning models instead learn their own representation from raw input. URLNet, proposed by Le et al. (2018), applies convolutional neural networks to both the characters and the words of a URL and outperformed models based on hand-crafted lexical features on a large dataset. Recurrent networks such as LSTMs have been used in the same way. These models capture subtle patterns, such as unusual character sequences, but they need large training sets and more computing power, and their decisions are harder to explain to users.

4. Comparison and common problems

Table 2 compares the five families. No single approach is sufficient. Lists are precise but late; heuristics and lightweight classifiers are fast and catch new sites but produce some false alarms; content methods are accurate but slow; deep learning is powerful but demanding.

Table 2. Comparison of phishing detection approaches.

Approach	Detects new sites?	Speed and cost	Main weakness
List-based	No	Very fast; cheap	Misses sites not yet reported
Heuristic rules	Partly	Fast	False alarms; easy to evade
Content / visual	Yes	Slow; page must load	Costly; fails on new brands
Machine learning (features)	Yes	Fast on URL features	Needs updating as attacks change
Deep learning	Yes	Slower; needs GPU to train	Large data needs; low transparency

Three problems affect all of them. First, attackers adapt: once a feature becomes widely used for detection, phishing kits change to avoid it. Second, datasets age quickly, and many studies still use collections that are several years old, so high reported accuracies may not hold on today's attacks. Third, studies use different data, features and measures, which makes their results difficult to compare (Khonji et al., 2013). A model that reports 99 per cent accuracy on an old, balanced dataset may do much worse on real traffic, where legitimate links far outnumber phishing ones.

5. The Indian context

Most research has focused on email phishing in English. In India, the main channels are SMS and messaging applications, and the messages are often written in Hindi, Marathi or other Indian languages, or in a mix of English and an Indian language written in Roman script. Lures commonly imitate banks, UPI applications, electricity boards, courier services and government schemes. Detection on the phone itself, at the moment a link is opened, suits this setting better than server-side filtering, which favours lightweight URL-based classifiers over heavy content analysis.

Research in India would benefit from three things: an openly available dataset of phishing SMS and URLs targeting Indian users; classifiers for multilingual and code-mixed message text; and evaluations that measure performance on realistic, imbalanced streams of links rather than on balanced benchmark datasets.

6. Conclusion

Phishing detection has moved from blacklists to machine learning and deep learning, and each step has improved the ability to catch new attacks. Yet the best protection still comes from layers: lists to block known sites instantly, lightweight classifiers to judge new links on the user's device, content analysis where time allows, and users who know what a phishing message looks like. For Indian users, whose exposure comes mostly through mobile messages in several languages, the most useful next steps are local datasets and models designed for SMS and code-mixed text.

Conflict of interest: The author declares no conflict of interest.

Funding: This research received no external funding.

Declaration of generative AI use: During the preparation of this work the author used an AI writing assistant (Claude, Anthropic) to help structure and draft the text. The author reviewed, edited and verified the content and takes full responsibility for this article.

References

- Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014). Phishing detection based associative classification data mining. *Expert Systems with Applications*, 41(13), 5948–5959. <https://doi.org/10.1016/j.eswa.2014.03.019>
- Dhamija, R., Tygar, J. D., & Hearst, M. (2006). Why phishing works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 581–590). <https://doi.org/10.1145/1124772.1124861>
- Khonji, M., Iraqi, Y., & Jones, A. (2013). Phishing detection: A literature survey. *IEEE Communications Surveys & Tutorials*, 15(4), 2091–2121. <https://doi.org/10.1109/SURV.2013.032213.00009>
- Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). URLNet: Learning a URL representation with deep learning for malicious URL detection. *arXiv*. <https://doi.org/10.48550/arXiv.1802.03162>
- Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). Beyond blacklists: Learning to detect malicious web sites from suspicious URLs. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1245–1254). <https://doi.org/10.1145/1557019.1557153>
- Mohammad, R. M., Thabtah, F., & McCluskey, L. (2014). Predicting phishing websites based on self-structuring neural network. *Neural Computing and Applications*, 25(2), 443–458. <https://doi.org/10.1007/s00521-013-1490-z>
- Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, 117, 345–357. <https://doi.org/10.1016/j.eswa.2018.09.029>
- Sheng, S., Wardman, B., Warner, G., Cranor, L. F., Hong, J., & Zhang, C. (2009). An empirical analysis of phishing blacklists. In *Proceedings of the Sixth Conference on Email and Anti-Spam (CEAS)*.
- Zhang, Y., Hong, J. I., & Cranor, L. F. (2007). CANTINA: A content-based approach to detecting phishing web sites. In *Proceedings of the 16th International Conference on World Wide Web* (pp. 639–648). <https://doi.org/10.1145/1242572.1242659>