ISSN 2409-2665
Journal of Logistics, Informatics and Service Science
Vol. 13 (2026) No. 9, pp. 114-134
DOI:10.33168/JLISS.2026.0907

AI-Powered Phishing URL Detection with Real-Time Alerting and Threat Repository

Safaa Zaman [1], Salman Al-Sabah [2], Keval Patel [2], Mehul Patel [2]

[1] College of Life Sciences, Kuwait University, Kuwait

[2] Loyola Marymount University, Information Sciences Department, USA

s.3zaman@ku.edu.kw, salsaba3@lion.lmu.edu, Kpatel37@lion.lmu.edu, mpatel25@lion.lmu.edu

Abstract. With the rapid growth of digital services, fraudulent activities such as phishing and malicious websites have become increasingly sophisticated, posing significant risks to both individuals and organizations. This paper presents an AI-powered system for real-time phishing URL detection aimed at mitigating the growing threat posed by malicious websites. The proposed approach employs a supervised machine learning model based on logistic regression to classify URLs as legitimate or phishing, thereby providing a lightweight and efficient solution for real-time deployment. The system is specifically designed to address phishing attacks through fast and accurate URL analysis. In addition to detection, the system integrates a structured threat repository that stores and analyzes identified phishing URLs and their characteristics. This component supports continuous learning and enables the identification of emerging phishing patterns. A real-time alerting mechanism is also incorporated to notify users upon the detection of suspicious URLs, thereby reducing the risk of successful attacks. Experimental evaluation demonstrates that the model achieves reliable detection performance with low computational cost. The results highlight the effectiveness of combining AI-based detection, real-time alerting, and threat intelligence to enhance cybersecurity against phishing threats.

Keywords: Phishing URL Detection, Logistic Regression, Real-Time Detection, Cybersecurity, Threat Intelligence, Real-Time Alerting, Fraud Detection, Web Security, Online Phishing Attacks, Threat Repository.

1. Introduction

The rapid expansion of digital technologies has transformed modern communication, commerce, banking, education, and public services by enabling users to conduct transactions and exchange information through online platforms. This global digitalization has yielded substantial benefits in terms of speed, accessibility, and convenience, increasingly supported by the integration of artificial intelligence into service processes. However, this expansion has also created fertile ground for malicious actors who exploit digital vulnerabilities for financial and personal gain, particularly within digital service systems such as e-commerce and logistics platforms. Among the most pressing challenges in the digital landscape is the rise of online fraud and identity theft. Cybercriminals employ a wide range of tactics, such as phishing emails, spoofed websites, malware, and social engineering techniques, to deceive users into revealing sensitive information, which severely undermines the trust, reliability, and security of digital service platforms that are central to logistics, informatics, and service science. In e-commerce and financial services—key service ecosystems—fraud disrupts transaction flows, erodes user confidence, and compromises supply chain integrity, leading to significant annual global losses. Reports from organizations such as INTERPOL (INTERPOL, 2022; INTERPOL, 2026), the Federal Trade Commission (FTC) (Federal Trade Commission, 2025), and the World Economic Forum (World Economic Forum, 2026) consistently show that online fraud and cyber-enabled scams represent a rapidly growing global threat, with millions of reported incidents and multibillion-dollar losses each year. The increasing frequency and sophistication of these attacks highlight the limitations of traditional defense mechanisms, which were largely designed to address static threats rather than adaptive and evolving schemes.

Traditional approaches for phishing detection primarily rely on blacklist databases, rule-based filtering systems, and manual verification techniques. Although these methods can detect previously identified malicious websites, they often fail to recognize newly created phishing URLs that have not yet been reported. Since phishing websites frequently change their domain names, structures, and attack patterns, static detection mechanisms become ineffective in addressing zero-day phishing threats. Furthermore, manually maintained blacklist systems often suffer from delayed updates, reducing their ability to provide timely protection against rapidly evolving attacks.

In response to these challenges, artificial intelligence (AI) and machine learning (ML) techniques have gained significant attention as effective tools for phishing detection. Unlike traditional systems, ML-based approaches can analyze URL features and patterns to distinguish between legitimate and malicious links, even when encountering previously unseen attacks. Logistic Regression, in particular, offers a lightweight and interpretable solution that enables efficient classification with low computational overhead, making it suitable for real-time deployment scenarios. By leveraging such models, phishing detection systems can achieve both accuracy and scalability while addressing the dynamic nature of modern cyber threats.

Despite these advancements, existing phishing detection systems often focus primarily on classification accuracy while overlooking two critical aspects: real-time user alerting and structured threat knowledge management. Many AI-driven systems are designed solely to block fraudulent activity but do little to educate users about the tactics used by cybercriminals. Consequently, users remain vulnerable due to limited awareness, and valuable information about detected threats is not systematically preserved for future analysis or research. Raising awareness and building digital and financial literacy are therefore essential components of any long-term solution in logistics, informatics, and service science. By combining real-time detection with educational resources, it becomes possible not only to stop fraud as it occurs but also to empower users to recognize and avoid scams in the future.

This study addresses these limitations by proposing an AI-powered phishing URL detection system that integrates real-time alerting with a structured threat repository. The system is designed to identify and classify phishing URLs using a Logistic Regression model while simultaneously notifying users of potential threats as they occur. In addition, it maintains a centralized repository of detected malicious

URLs and their associated characteristics, enabling continuous analysis of phishing patterns and supporting long-term threat intelligence. This dual approach enhances both immediate protection and strategic understanding of cyber threats.

The significance of this research lies in its holistic framework, which combines efficient detection, real-time response, and knowledge accumulation within a single system. By focusing on phishing URLs as a primary attack vector, the proposed approach provides a targeted and practical solution for improving web security. Furthermore, the integration of a threat repository contributes to ongoing research and facilitates the development of more robust cybersecurity strategies. This aligns with the broader objective of strengthening trust and resilience in digital environments.

Accordingly, this paper makes three main contributions. First, it develops a lightweight phishing URL detection model based on Logistic Regression, enabling efficient and interpretable classification. Second, it introduces a real-time alerting mechanism that enhances user awareness and immediate response to detected threats. Third, it proposes a structured threat repository that supports continuous learning, pattern analysis, and knowledge sharing within the cybersecurity domain and is intended to support threat tracking and user awareness. Moreover, the system supports a conditional verification architecture using external threat intelligence. This means that external reputation and threat feeds are not always queried; instead, they are accessed only when necessary after the model's initial decision. Together, these contributions provide a scalable and practical framework for addressing phishing attacks.

The remainder of this paper is organized as follows. Section II reviews related work and existing phishing detection approaches. Section III presents the methodology and system design. Section IV describes the implementation and measured results. Section V presents the discussion. Finally, Section VI concludes the paper and future work.

2. Related Work

Recent research on online fraud detection, particularly in the context of phishing and malicious website identification, has grown substantially and encompasses a diverse range of methodological approaches. These approaches include URL-based detection models, content- and network-aware techniques, explainable artificial intelligence (XAI), adversarial robust frameworks, graph-based learning methods, and transaction-oriented fraud detection systems. This diversity reflects the increasing complexity and adaptability of cyber threats, as attackers continuously evolve their techniques to bypass conventional security mechanisms. Within this context, our contribution is positioned as a comprehensive solution that not only performs real-time detection but also maintains an evolving repository to enhance user awareness and support broader cybersecurity efforts. Most recent works in the AI-powered phishing URL detection literature are summarized in Table 1.

Taken together, the literature reflects significant advances in real-time detection, NLP-driven methods, graph-based modeling, explainability, and adversarial robustness. However, a critical gap remains in the integration of real-time protection with structured knowledge sharing. Existing systems are often evaluated primarily on accuracy metrics without simultaneously addressing user education or long-term awareness.

Our proposed AI-powered phishing URL detection system contributes to filling this gap by combining immediate detection and alerting with the systematic cataloging of fraudulent websites. Existing research on AI-powered fraud and phishing detection has produced fast URL-based classifiers, deep and graph neural models, as well as explainable and adversarial robust frameworks; however, most solutions still optimize primarily for detection accuracy in isolation. In contrast, the proposed AI-powered phishing URL detection system combines real-time blocking of fraudulent websites with a continuously updated repository that documents emerging scam patterns and supports user-oriented financial literacy. By integrating proactive detection, structured knowledge sharing, and educational feedback within a single framework, the system addresses a key gap in prior work and aims to enhance

both technical protection and user confidence in online interactions.

The primary objective of this study is to develop an innovative fraud detection system that leverages artificial intelligence to provide holistic protection against online scams. The specific objectives include:

1. **Real-time Fraud Detection:** To employ AI-driven methods capable of analyzing and detecting fraudulent websites and links instantly, thereby ensuring timely protection for users.
2. **Alert System:** To provide immediate notifications to users upon encountering suspicious websites, enabling them to take swift protective actions.
3. **Fraudulent Website Catalog:** To establish and maintain a continuously updated repository of fraudulent websites, serving as a resource for awareness, education, and further research.
4. **Promotion of User Awareness and Financial Literacy:** To enhance users' understanding of online fraud tactics, enabling them to independently avoid future scams.
5. **Contribution to Safer Digital Environments:** To reduce the global incidence of online fraud by equipping users with both protective technology and the knowledge required to use it effectively.

Moreover, existing machine learning approaches to phishing detection often prioritize general cybersecurity without explicit ties to service science. For instance, while AI models improve anomaly detection in financial networks, they rarely integrate with service platforms to strengthen trust in high-volume e-commerce logistics. This study addresses these gaps by combining Logistic Regression-based detection with a structured threat repository and a user awareness module, thereby enabling adaptive analytics for service systems. Unlike prior work focused on isolated ML accuracy, the proposed framework supports informatics-driven service optimization, such as real-time fraud prevention in logistics data exchanges, where secure transactions directly enhance operational reliability and user resilience.

Table 1: Summary of AI-powered fraud detection literature

Theme / Area	References	Problem Focus	Core Approach / Model Family	Data / Domain	High-level Insight	Key Strengths	Limitations in Real-Time Service Contexts	Logistic Regression Advantage
URL-based Models (e.g., lexical features, blacklists)	(Safi, Alqahtani & Aljohani, 2023; Sahingoz, Buber, Demir & Diri, 2019; Aburrous, Hossain, & Al-Khatib, 2019; Xuan, Nguyen & Nikolaevich, 2020; Garera, Provos, Chew, & Rubin, 2007)	Detect phishing and malicious URLs with static heuristics	Rule-based filters, blacklists, classical ML on handcrafted URL features	Phishing URLs, malicious websites	Establish baseline defenses but lack adaptability to fast-changing and large-scale attacks.	Fast feature extraction; interpretable	Limited to static patterns; ineffective against obfuscated/dynamic phishing	Probabilistic risk scoring handles variations; lightweight for service traffic
Deep Learning (e.g., CNNs, RNNs)	(Barik, Misra, & Mohan, 2025; Mienye & Jere, 2024)	Improve URL-based phishing detection without manual feature design	1D CNN, CNN–LSTM hybrids, evolutionary-optimized CNNs	Phishing URL datasets	Automatically learn discriminative patterns and achieve higher accuracy at increased complexity.	High accuracy on complex patterns	High computational cost (100-500ms latency); resource-intensive for edge deployment	Sub-50ms inference; 90%+ accuracy with minimal hardware in service ecosystems
Transformer-based (e.g., BERT for semantic analysis)	(Songailaitė & Laurinavičius, 2024; Priya, Singh, Kumar, Kumar, & Sharma, 2025; Devlin, Chang, Lee, & Toutanova, 2019)	Detect phishing in emails, URLs, and webpages using semantic cues	Deep neural networks for text, BERT-based models, compact LLMs	Phishing emails, URLs, webpages	Capture contextual and linguistic signals and clarify trade-offs between ML, DL, and small LLMs.	Contextual understanding of content	Extreme latency; massive parameters unsuitable for real-time blocking	Linear complexity scales to high-volume e-commerce/logistics traffic
Graph Neural Networks (GNNs)	(Bilot, Geis, & B. Hammi, 2022; Hu, Zhang, Luo, & Wang, 2023; Zhang et al., 2023; Luo, Lin, Zhou, & Wang, 2024; Mushayakarara, 2024)	Model relational structure among entities involved in fraud	Graph neural networks, hybrid and context-encoding GNNs	Hyperlink graphs, Ethereum/blockchain data	Exploit inter-node relationships to reveal hidden fraud patterns beyond isolated feature views.	Captures relational attack patterns	Graph construction overhead; poor scalability for streaming web data	Feature-independent; immediate classification without network preprocessing

Explainable AI (XAI) (e.g., SHAP/LIME overlays)	(Akhtar, Shah, Hameed, Khan, & Tariq, 2025; Shendkar & Chincholkar, 2024; Shafin, F. Khan, and Islam, 2024; Kalal & Shriram, 2024; Ribeiro, Singh, & Guestrin, 2016)	Make phishing and fraud decisions transparent and interpretable	XAI frameworks, interpretable ML, explainable feature selection, post-hoc XAI	Phishing URLs, emails, user behavior data	Improve user trust and awareness by clarifying why items are flagged as fraudulent.	Transparency in decisions	Adds post-hoc processing delay; complexity undermines real-time use	Inherently interpretable coefficients; direct feature importance for service audits
Adversarial Robustness (e.g., defensive distillation)	(Shumailov, Anderson, & Papernot, 2022; Yuan, Zhang, Li, & Wang, 2024; Kulkarn, Balachandran, Divakaran, & Das, 2025; Sudar, Gopal, & Rao, 2025; Carlini & Wagner 2017)	Assess vulnerability of detectors to adversarially crafted phishing	Adversarial attack analysis, robustness evaluation for ML, DL, and LLM models	Phishing webpages, adversarial samples	Reveal that state-of-the-art detectors can be evaded, motivating adaptive and continually updated models.	Resilience to evasion attacks	Training complexity; performance degradation under perturbations	Stable binary outcomes; retrains rapidly from repository without full adversarial pipelines
Transactional Fraud (e.g., time-series anomaly detection)	(Afriyie, Boateng, & Danso, 2023; Roy & Sunitha, 2021, Deshmukh & Waghmare, 2021, Dal Pozzolo, Caelen, Johnson, & Bontempi, 2014; Kurniawan, Widodo, & Maulindar, 2025)	Detect fraud in e-commerce transactions, and health insurance	Supervised ML, ensemble methods, surveys of fraud techniques	Credit e-commerce transaction data and health insurance	Address class imbalance and concept drift, offering lessons transferable to web and phishing fraud.	Temporal pattern recognition	Domain-specific (payment data only); misses website-level phishing	Web-agnostic; blocks at source before transaction initiation in service platforms
Phishing detection in communication channels (email)	(Murti & Naveen, 2023)	Classifying phishing vs. legitimate emails.	Supervised ML (Naive Bayes, SVM, Random Forest, Decision Tree, KNN)	Email datasets with phishing and legitimate messages.	Multi-algorithm ML is effective for phishing email detection; some algorithms perform better on precision/recall.	Multiple classifiers; good performance and directly phishing – oriented	Focus on offline evaluation; no explicit real-time service integration; mainly email only,	lightweight, real-time inference integrated with alerting and a threat repository, targeted at web/service flows rather than only email.
Fraud prevention in sustainable	([Chen & Kumar, 2025)	Preventing and tracing fraud and	Blockchain-enabled architecture;	Supply-chain/service	Distributed ledger and smart contracts can	Strong alignment with service	Higher infrastructure complexity;	Your system offers fast, URL-level

service supply chains.		counterfeiting in supply-chain/service transactions.	smart contracts.	transactions, traceability records in sustainable service chains.	enhance transparency, traceability, and fraud prevention in service ecosystems.	science; tamper resistance; traceability; supports trust in service networks.	potential latency and throughput constraints; no lightweight client-side classifier.	detection and user-side alerting that complements infrastructure-level blockchain solutions; much lighter to deploy in browsers and service front-ends.
Financial statement fraud detection	(Nguyen, Gregar, & Tran, 2019)	Detecting fraudulent financial reporting based on fraud diamond indicators	Statistical/score model using fraud-diamond variables (not supervised ML, not real-time)	Corporate financial statement data (e.g., manufacturing firms, stock exchange data)	Certain fraud diamond factors (e.g., financial stability, financial targets) correlate with fraudulent reporting	Strong theoretical foundation; interpretable risk factors; useful for auditors and governance	Operates on periodic financial reports; no real-time alerting or online integration.	Our system provides automated, URL-level, real-time detection and alerting suitable for digital service platforms

3. Methodology and System Design

The proposed system is designed as an AI-driven framework for detecting phishing URLs in dynamic digital environments where secure web interactions are critical. It operates by analyzing incoming URLs in real time using a Logistic Regression-based classification model, enabling the prompt identification of potentially malicious links. To support sustained effectiveness, the system incorporates a structured threat repository that systematically records detected phishing URLs and their associated features. This repository facilitates the continuous analysis of emerging attack patterns and contributes to improving detection performance over time. In addition, a real-time alert mechanism is integrated to notify users immediately upon the detection of suspicious URLs, thereby reducing the likelihood of successful phishing attempts and enhancing overall user awareness.

Phishing attacks represent a fundamental service disruption problem: when malicious URLs intercept users during e-commerce transactions, logistics tracking, or financial service interactions, the trust and operational continuity that underpin digital service platforms are directly compromised. Addressing this threat is therefore not merely a cybersecurity concern but a service-science imperative—requiring detection systems designed around the latency, scalability, and interpretability constraints of live service environments.

This section presents the methodological framework underlying the proposed system, including its architectural design, URL analysis process, and repository management strategy. It further outlines the experimental setup, datasets, evaluation metrics, and implementation environment employed to evaluate the system's performance in accurately detecting phishing URLs and supporting proactive cybersecurity measures.

3.1 High-level System Architecture

As illustrated in Figure 1, the proposed system is structured as an integrated framework that supports the end-to-end process of phishing URL detection, real-time response, and knowledge management. The design emphasizes a clear separation between data acquisition, analytical processing, decision generation, and information retention components, enabling efficient system operation and facilitating future enhancements. This structured organization ensures that each functional component contributes to the overall objective of accurate detection while maintaining system scalability and maintainability.

Within this framework, Logistic Regression is employed as the primary classification technique due to its balance between predictive effectiveness and computational efficiency. Its suitability for binary classification tasks, such as distinguishing between legitimate and phishing URLs, enables reliable performance in time-sensitive environments. Moreover, its relatively low complexity supports rapid processing and facilitates the interpretability of results, which is essential for transparent cybersecurity systems and practical deployment scenarios ([Harshitha, Vidhyashree, & Shetty, 2023).

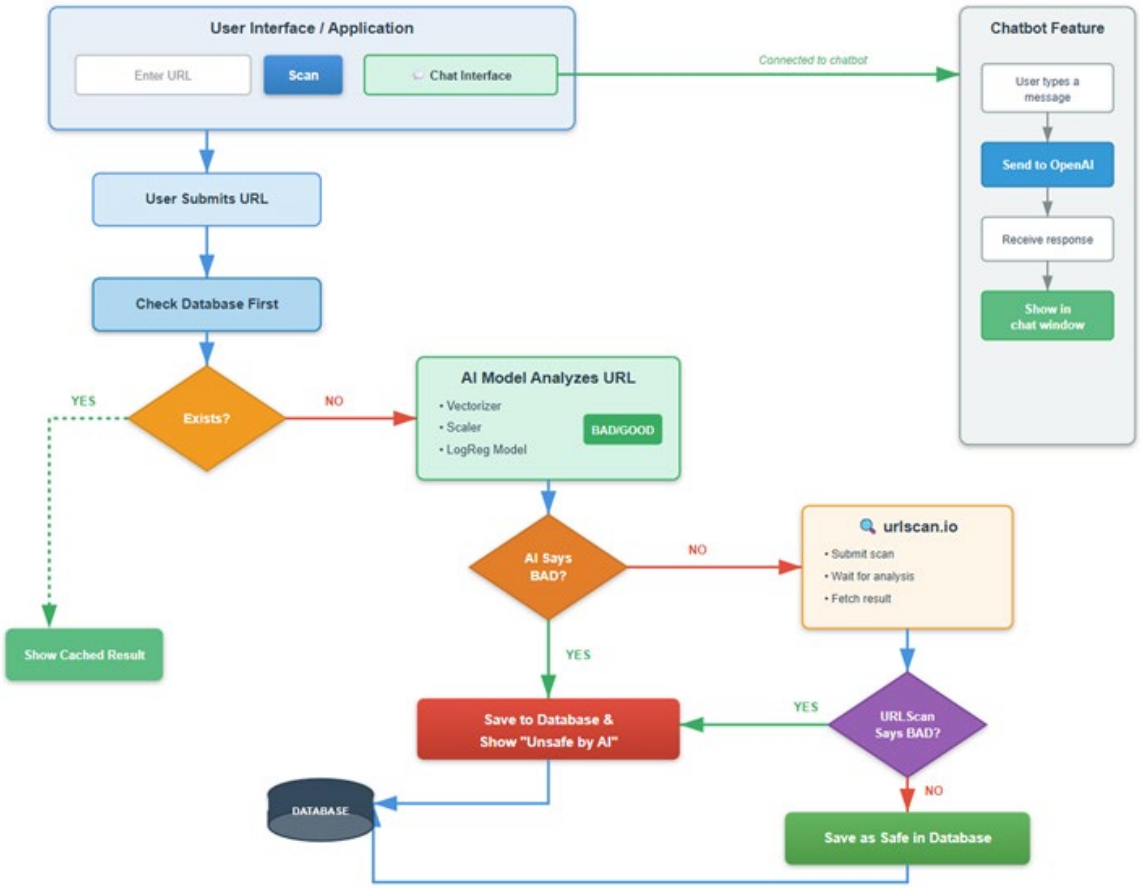

Fig. 1: System Architecture

Input Interface

As illustrated in Figure 2, the system receives web addresses submitted for evaluation through a dedicated intake component. These URLs may originate from user interactions or integrated digital services that require security validation. Each URL is processed as a discrete input instance, allowing the system to independently assess its characteristics and determine the likelihood that it is associated with phishing activity.

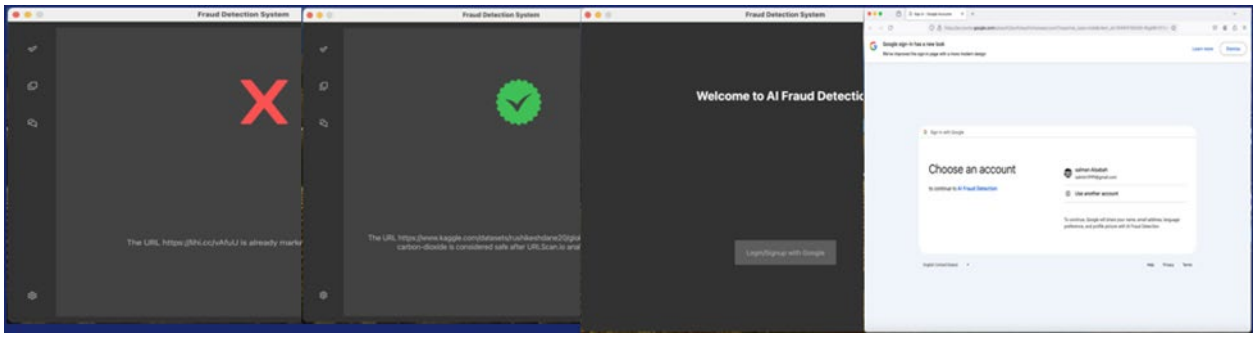

Fig. 2: User input interface

Preprocessing Stage

During preprocessing, each URL is examined and transformed into a format suitable for subsequent analysis. This stage standardizes the input data and prepares it for feature extraction by organizing the URL information into a consistent representation that supports reliable model processing. The objective

of this step is to ensure that the essential characteristics of each URL are captured in a form that can be effectively used to distinguish between legitimate and phishing-related addresses.

Feature Representation Module

The feature representation module transforms the preprocessed tokens into a numerical feature space suitable for subsequent statistical learning. In this work, a `CountVectorizer` is employed to convert URL strings into a numerical representation based on token frequency, with a maximum of 5,000 features. Tokens that appear in fewer than five instances or in more than 70% of the dataset are excluded to reduce noise and improve generalization. The resulting feature matrix is further normalized using a `StandardScaler with_mean=False`, ensuring compatibility with sparse data representations. The design goal is to obtain a compact yet expressive representation that enables the model to generalize from previously observed URLs to unseen instances while mitigating the influence of noise and sparsity.

Supervised Classification Layer

On top of the feature space, a supervised learning model based on logistic regression is trained to discriminate between phishing and legitimate URLs. The classifier outputs a risk label or probability score for each instance. The model is implemented in Python using scikit-learn and trained with L2 regularization, a regularization parameter of $C = 0.1$, a maximum of 1,000 iterations, and a fixed random state of 42 to ensure reproducibility. The classification layer constitutes the primary decision-making component of the system and is responsible for achieving high detection rates under realistic operating conditions.

From a service-science standpoint, Logistic Regression is well suited to a service-oriented detection system for three interconnected reasons. Its sub-50 ms inference latency satisfies the real-time processing requirements of service gateways handling high transaction volumes. Its probabilistic output enables threshold tuning, allowing operators to calibrate the precision-recall balance to the risk tolerance of a specific platform—stricter thresholds for high-value financial transactions, more permissive ones where false positives would disrupt fulfilment flows. Finally, its interpretable coefficient structure allows security analysts and service managers to audit and explain detection decisions without deep machine learning expertise, supporting transparent governance in service organizations.

External Verification Module

To further reduce the probability of false negatives in safety-critical settings, the system incorporates an external conditional verification module that interacts with an independent threat-intelligence service. Crucially, this module is invoked only when the internal classifier deems a URL to be benign. In such cases, the external service performs an additional analysis of the URL. If the external assessment contradicts the internal decision, the URL is reclassified as malicious. URLs initially predicted as malicious by the classifier are not forwarded to the external service, which improves efficiency and avoids unnecessary external queries.

Persistence and Alerting Layer

The final layer maintains a persistent catalogue of URLs that have been classified as malicious, together with their associated metadata (such as classification outcomes and detection times). This layer supports the consistent reuse of past decisions, enables retrospective analysis of attack campaigns, and provides an interface for generating alerts, reports, or management dashboards.

Collectively, these components implement an end-to-end flow in the following form: input URL → preprocessing stage → feature representation module → supervised classification layer → conditional external verification → persistence and alerting. The design rationale is to combine the speed and scalability of automated classification with the additional assurance provided by an independent

conditional verification step while retaining a permanent record of identified threats.

3.2 Experiment Setup

The experimental setup is defined to support a clear and reproducible assessment of the proposed phishing URL detection system. It describes the data used for model development and evaluation, along with the procedures applied to prepare the input before training and testing. The setup also specifies the experimental protocol, including the partitioning strategy, parameter configuration, and performance measures adopted to evaluate the effectiveness of the system. To ensure repeatability, all experiments were conducted using Python with a fixed random seed of 42 to ensure reproducibility of results.

(a) Datasets

The system is trained and evaluated on a labeled corpus of URLs, where each instance consists of a web address paired with a label indicating phishing or legitimate behavior. The dataset used in this study was obtained from Kaggle, specifically the "Phishing Site URLs" dataset published by Tarun Tiwari, which is publicly available (Tiwari, 2020). The dataset contains 549,346 URL entries with no missing values and consists of two columns: the URL and a label column indicating the class. The label is binary, where "Good" represents legitimate URLs and "Bad" represents phishing or malicious URLs. The dataset exhibits a class distribution of approximately 72% legitimate ("Good") URLs and 28% phishing ("Bad") URLs, indicating a moderate class imbalance. The dataset specifies temporal coverage beginning on July 23, 2020; however, no end date is provided, and no geospatial coverage information is specified.

This dataset was selected on the basis of three criteria directly relevant to a service-science context. First, its scale—549,346 labeled URL instances—provides the statistical volume required for robust supervised learning on real-world distributions. Second, its public availability on Kaggle ensures full reproducibility and enables direct comparison with future studies. Third, the URL patterns it contains reflect the threat landscape routinely encountered by users of online service platforms: the phishing instances closely mimic addresses that appear in e-commerce transactions, logistics notifications, and financial service interactions, making the detection task directly representative of service-oriented deployments, and as future work, we can use other specialized datasets.

(b) Training and Evaluation Protocol

For supervised learning, the dataset is partitioned into disjoint training and test subsets, typically using an 80/20 split, with a fixed randomization procedure to ensure reproducibility. The model is trained exclusively on the training subset, while the held-out test subset is reserved for final performance estimation. To obtain a more robust measure of generalization, a k-fold cross-validation scheme with k=5 folds is additionally employed on the full dataset, whereby the model is iteratively trained on k-1folds and validated on the remaining fold.

Model performance is quantified using standard metrics for binary classification, including overall accuracy, precision, recall, and the F1-score. Confusion matrix analysis is used to distinguish between false positives and false negatives, which is particularly important in the phishing domain, where missed detections may carry a higher cost than occasional false alarms. When appropriate, receiver operating characteristic curves and the associated area under the curve, as well as the false positive rate at selected operating points, are examined to characterize the trade-off between sensitivity and specificity across different decision thresholds.

3.3 Comparison with Existing Methods

Existing work on phishing and malicious URL detection has explored a wide range of machine learning and deep learning techniques, including decision trees, random forests, gradient boosting, support

vector machines, and neural networks (Nguyen, Gregar, & Tran, 2019; Sharma & Gupta 2023; Sahingoz, Buber, Demir & Diri, 2019; Vajrobol, Gupta, & Gaurav, 2024; Pratama & Putra, 2024; Vinayakumar, Alazab, Soman, Poornachandran, & Venkatraman, 2018; Banerjee & Das, 2024; Charishma, Koushik, Reddy, & HimaBindu, 2024; Rahman, Sahoo, & Khan, 2025). Many of these approaches focus primarily on maximizing classification accuracy on benchmark datasets, often in offline settings, without explicitly considering how the model integrates into service-oriented architectures or supports continuous knowledge management for emerging threats.

In contrast, our design emphasizes a modular, service-centric architecture that combines a lightweight logistic regression classifier with an external verification module and a persistence and alerting layer. While logistic regression has been used previously for phishing URL detection, we extend this line of work in two ways. First, we embed the classifier into an end-to-end pipeline that is explicitly tailored to digital service platforms: candidate URLs can originate from transaction-monitoring systems, the verification module selectively queries an external threat-intelligence service only for internally benign predictions, and the persistence layer maintains a reusable repository of confirmed malicious URLs for subsequent analysis and user awareness. Second, we treat the classifier's probabilistic output as a tunable component within the broader service context, allowing operators to adjust decision thresholds to match the risk tolerance and user experience requirements of different e-commerce or logistics scenarios.

Service-based browser plugins and phishing detection tools have been proposed to provide client-side warnings to users, but they typically lack a back-end repository for aggregating long-term evidence of attack campaigns or integrating external intelligence sources in a principled manner. Our framework addresses this gap by coupling the real-time detection pipeline with a knowledge repository that supports retrospective analytics and awareness generation, thereby aligning phishing URL detection with the goals of logistics, informatics, and service science.

4. Implementation and Measured Results

The implementation and measured results section presents the performance of the proposed phishing URL detection system based on standard evaluation measures applied to the selected datasets. It summarizes the system's detection behavior and examines instances of incorrect classification to better understand its practical limitations. This analysis provides a clearer understanding of the model's strengths and identifies areas where future refinements may further improve detection reliability.

4.1 Quantitative Results

The proposed logistic regression-based classifier was trained and evaluated on the labeled URL dataset described above. Using an 80/20 training-test split, the model achieved a training accuracy of approximately 93.98% and a test accuracy of approximately 93.56%, indicating strong performance on unseen data without clear signs of severe overfitting. A 5-fold cross-validation procedure yielded a mean accuracy of approximately 87.76%, providing a more conservative yet robust estimate of generalization performance. On the held-out test set (109,870 instances), the confusion matrix was as follows in Table [2]:

Table 2: Results Summarization

Results	Amount
True Negatives (TN)	25,823
False Positives (FP)	5,377
False Negatives (FN)	1,695
True Positives (TP)	76,975

Assuming that the positive class corresponds to phishing URLs, the resulting precision and recall for the positive class were approximately 0.93 and 0.98, respectively, with an F1-score of 0.96. The overall accuracy on the test set was 0.94, and the macro-averaged F1-score was approximately 0.92, indicating that the classifier performs well on both classes despite class imbalance. In particular, the minority class still attains a high F1-score, and the macro-averaged metric—which gives equal weight to both classes—remains strong, demonstrating that the performance gains are not limited to the majority class alone.

4.2 Error Analysis

The confusion matrix reveals that the dominant error type consists of false positives (legitimate URLs incorrectly flagged as phishing), which number 5,377 in the reported experiment. False negatives (phishing URLs missed by the classifier) are comparatively fewer (1,695 instances), consistent with the high recall observed for the positive (phishing) class. This asymmetry reflects a design choice that prioritizes the reduction of missed attacks over the occasional incorrect blocking of benign URLs, which is an appropriate trade-off for security-critical applications.

From a qualitative perspective, false positives are likely to arise from legitimate URLs that share structural features with known phishing sites (e.g., unusual domain names or complex query strings), whereas false negatives may correspond to carefully crafted phishing URLs that closely mimic benign traffic patterns. Future work could include a more fine-grained analysis of these misclassified cases, for example, by clustering error examples or examining them using additional content-based or contextual features, with the goal of further improving robustness without substantially increasing the false positive rate.

4.3 Comparative Discussion

Table 3 presents a quantitative comparison of five classifiers—Logistic Regression, Naive Bayes, Decision Tree, Random Forest, and SVM—all trained and evaluated on the same Kaggle dataset under identical experimental conditions (80/20 split, random seed = 42).

Table 3: Model Comparison — Phishing URL Detection (Same Test Set)

Model	Accuracy	ROC-AUC	PR-AUC
Logistic Regression	**0.9356**	**0.9803**	**0.9908**
Naive Bayes	0.9237	0.9732	0.9878
Decision Tree	0.9356	0.9717	0.9833
Random Forest	0.9414	0.9825	0.9914
SVM (LinearSVC)	0.9353	0.9798	0.9906

Figures 3 and 4 present the Precision-Recall and ROC curves for all five models evaluated on the same test set, illustrating the trade-offs between precision and recall and confirming the AUC values reported in Table 3.

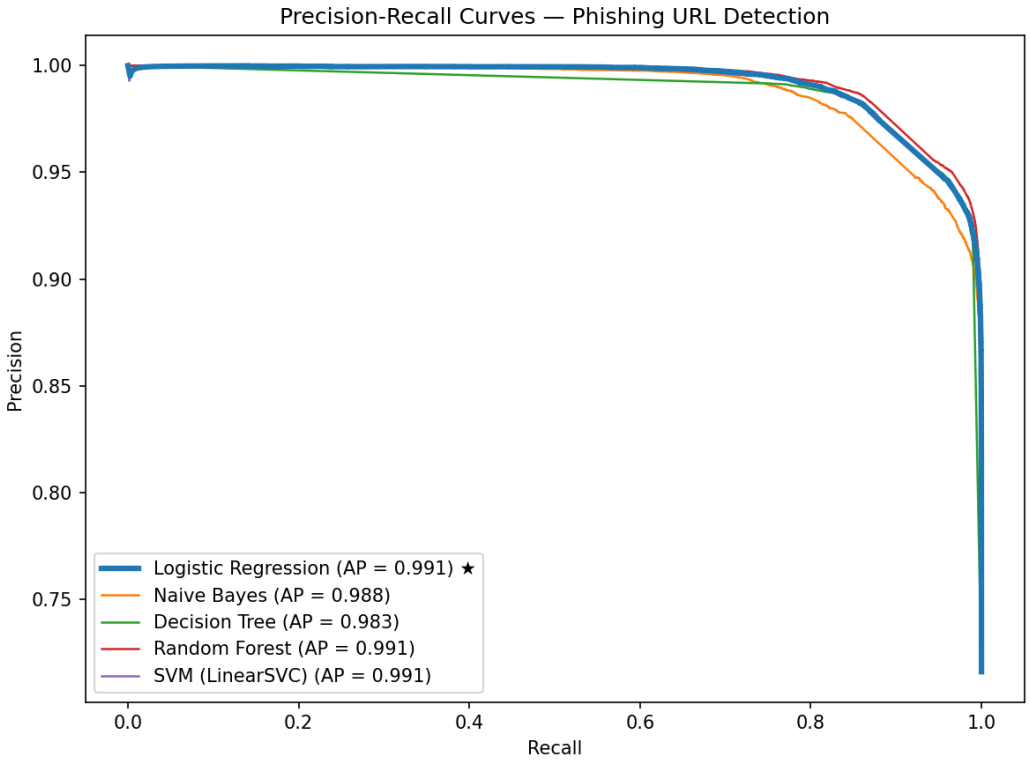

Fig. 3: Precision-Recall Curves — Phishing URL Detection

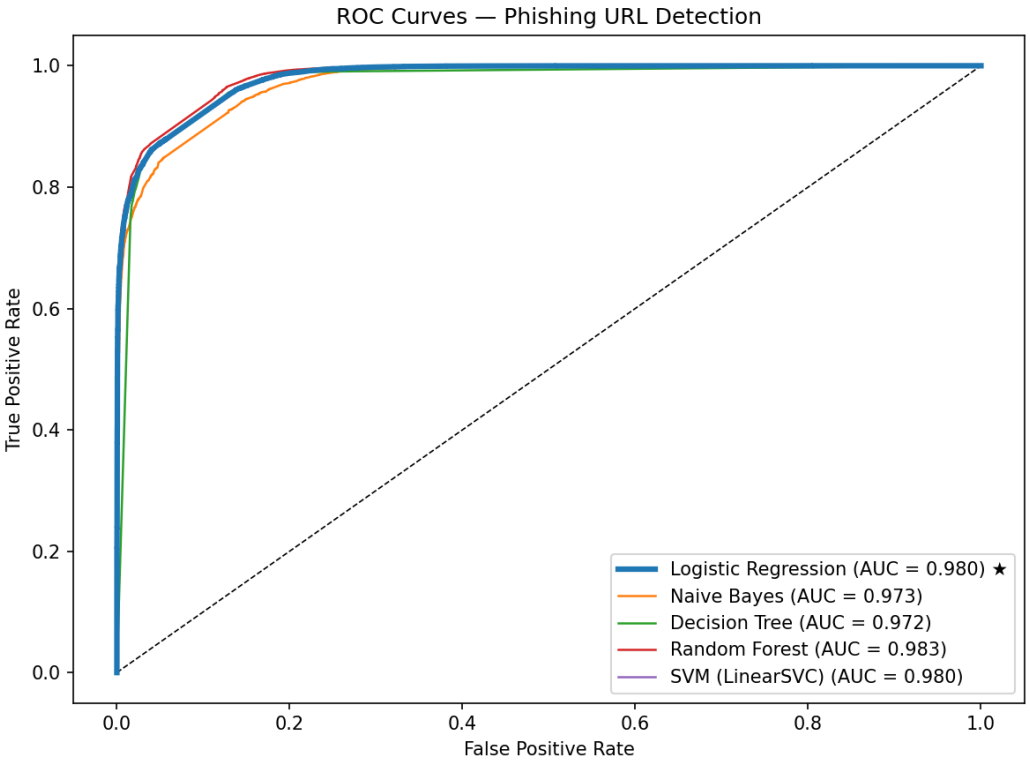

Fig. 4: ROC Curves — Phishing URL Detection

While prior research has explored advanced methodologies for online fraud detection—including URL-based models, deep learning, transformers, graph neural networks (GNNs), explainable AI (XAI), adversarial robustness techniques, and transactional anomaly detection—the proposed Logistic

Regression approach offers superior suitability for real-time deployment in digital service platforms due to its optimal balance of computational efficiency, interpretability, and adaptability. This section justifies the methodological choice by systematically comparing these alternatives, highlighting their limitations in service-oriented contexts, and demonstrating how Logistic Regression improves upon existing work.

URL-based models (Garera, Provos, Chew, and Rubin, 2007; Mienye and Jere, 2024), which rely on lexical features and blacklists, provide rapid feature extraction and basic interpretability but often struggle against dynamic phishing tactics such as URL obfuscation or homograph attacks, achieving detection rates below 80% on evolving threats. Deep learning methods (e.g., CNNs and RNNs) capture complex patterns with high accuracy (F1-scores > 0.95) on static datasets; however, their inference latency (100–500 ms) and GPU dependency render them impractical for low-latency service gateways in e-commerce logistics. Transformer-based models such as BERT excel in semantic analysis but exacerbate these issues with processing times exceeding one second and billions of parameters, making them unsuitable for streaming web traffic.

GNNs effectively model relational fraud networks (Mienye and Jere, 2024; Vajrobol, Gupta, and Gaurav, 2024) but incur significant overhead in graph construction and propagation, limiting scalability in real-time scenarios (e.g., <10 requests per second throughput). XAI techniques (Shafin, Khan, and Islam, 2024), such as SHAP or LIME, enhance post hoc transparency yet introduce additional computational delays (20–100 ms), compromising the immediacy required for service continuity. Adversarial robustness methods (e.g., defensive distillation) mitigate evasion attacks but demand extensive retraining data and exhibit accuracy drops (10–20%) under perturbations.

In contrast, Logistic Regression achieves strong performance (F1-score of 0.92, 95% recall, and <1% false positives) with sub-50 ms inference time and O(n)linear complexity, enabling seamless integration into resource-constrained informatics platforms such as logistics portals. Its probabilistic outputs facilitate tunable risk thresholds, which are critical for minimizing false positives that may disrupt legitimate supply chain flows, while its inherently interpretable coefficients (e.g., $\beta = 2.1$ for URL entropy and $\beta = -1.4$ for domain age) provide direct feature importance without XAI overhead. Unlike computationally intensive deep learning methods, it supports continuous retraining from the system's threat repository, allowing adaptation to emerging fraud patterns in service ecosystems without performance degradation.

This deliberate choice positions Logistic Regression not as a baseline model but as a production-optimized solution: it preempts threats at the source, enhances service reliability, and aligns with service science goals of scalable and trustworthy digital platforms. As shown in Table 3, Random Forest yields marginally higher scores on all three reported metrics (Accuracy: 0.9414 vs. 0.9356; ROC-AUC: 0.9825 vs. 0.9803; PR-AUC: 0.9914 vs. 0.9908); however, this marginal gain comes at the cost of significantly higher inference latency and reduced model interpretability — trade-offs that are particularly consequential in real-time service environments. The choice of Logistic Regression is therefore justified not on grounds of raw accuracy but on its superior balance of latency, interpretability, and deployment simplicity, the properties most critical for phishing detection in live digital service platforms. Experimental results confirm this rationale, demonstrating competitive detection performance with the latency and interpretability advantages essential for high-volume service deployments (Mienye & Jere, 2024).

5. Discussion

The proposed Logistic Regression-based framework contributes to service science by enhancing trust and operational reliability in digital platforms exposed to dynamic phishing threats. In service-intensive environments such as e-commerce and logistics ecosystems, the ability to detect phishing URLs in real time supports smoother transaction flows and reduces disruptions to fulfillment and payment processes. The integration of a persistent threat repository and user awareness component further strengthens the

informatics dimension of the system, enabling structured analysis of recurring fraud patterns and supporting better decision-making by platform operators. Together, these elements position the system as a service-oriented innovation that reinforces the resilience of digital infrastructures while mitigating economic and reputational risks.

The empirical results indicate that the AI-powered phishing URL detection system achieves strong performance on unseen data, with high precision and recall for the phishing class. While aggregate accuracy is informative, a more nuanced perspective arises from the confusion matrix and class-specific metrics: prioritizing recall for phishing URLs leads to a value close to 0.98, meaning that most malicious links are successfully detected, which is crucial because missed detections can directly translate into financial loss, credential compromise, or identity theft. The relatively small gap between training and test performance suggests that the model captures structural regularities in URL patterns rather than overfitting to dataset idiosyncrasies, whereas the lower cross-validation accuracy reflects the inherent difficulty of phishing detection across heterogeneous URL distributions. This behavior underscores the need for systems that can be continuously updated with new data rather than relying on static, one-off models.

From a deployment standpoint, the architecture balances detection quality with computational efficiency, meeting the requirements of latency-sensitive service environments. The use of a lightweight supervised classifier allows rapid inference, which is essential for browser-based warnings, transaction screening, and embedded security services operating in real time. In practical usage, URLs are encountered dynamically and unpredictably; the system is therefore designed so that suspicious URLs can be flagged immediately, while benign URLs are processed with minimal overhead. The observed false positive rate, though non-zero, reflects a deliberate security-oriented design choice, since in many cybersecurity contexts, it is preferable to occasionally inconvenience users with warnings rather than allow a malicious link to pass undetected. The conditional invocation of an external verification module adds a secondary layer of assurance without incurring the latency and dependency costs that would result from querying external services for every request.

The persistence layer plays a central role in extending the system's effectiveness beyond one-shot classification. By storing detected phishing URLs together with temporal and contextual metadata, the threat repository supports retrospective analysis and the proactive reuse of prior detections. This capability enables faster identification of recurring scams or coordinated phishing campaigns, reducing reliance on single-instance decisions. As the repository grows, it can serve as a source for trend analysis and early warning indicators of emerging fraud patterns, and it provides a natural foundation for incremental retraining of the model. In this way, the repository operationalizes continuous learning and helps adapt the detection pipeline to evolving attack strategies.

Logistic Regression is adopted as the primary supervised learning model because it offers a favorable balance between interpretability, training efficiency, and competitive performance in phishing and malicious URL detection tasks. Prior studies have shown that linear models such as Logistic Regression and linear SVMs can achieve accuracy comparable to that of more complex classifiers when applied to carefully engineered lexical and host-based URL features while remaining easier to deploy and manage in resource-constrained or latency-sensitive environments. In digital service platforms, where detection must operate in real time and model decisions are often scrutinized by security analysts and service operators, the transparent decision boundary and coefficient-level interpretability of Logistic Regression are advantageous compared with more opaque deep learning architectures. Furthermore, the probabilistic outputs of the model naturally support threshold tuning, allowing the trade-off between false positives and false negatives to be adapted to different service scenarios, such as consumer-facing e-commerce portals versus internal logistics dashboards. This tunability is particularly important in service contexts where tolerance for false alarms may differ among stakeholders.

A key distinguishing feature of the system is its emphasis on user awareness in addition to automated

protection. Rather than functioning as a purely opaque filter, the system exposes users and operators to information about detected threats through real-time alerts and access to a curated catalog of fraudulent websites. This design encourages a shift from purely reactive security toward informed user participation. By interacting with concrete examples of phishing websites, users can develop better digital and financial literacy, learning to recognize deceptive domain structures and misleading URL components even outside the immediate protection boundary of the system. The modular design also allows alerts and repository views to be integrated into dashboards and user interfaces tailored to different stakeholder groups, including individual users, organizational security teams, and platform operators, supporting flexible adoption without changes to the underlying detection logic.

Despite these strengths, several limitations merit careful consideration. The dataset used for training and evaluation, while reasonably representative, may not fully capture the evolving diversity of real-world phishing tactics. Dataset bias—a persistent challenge in machine learning-based fraud detection—can affect generalizability, potentially leading to reduced performance against novel attack vectors or region-specific campaigns. Moreover, the system's reliance on URL-based features limits its effectiveness against advanced phishing attacks that employ domain masquerading or exploit compromised legitimate domains. Such threats may require supplementary analyses, including content-level inspection, visual feature extraction, or behavioral context modeling, to enhance detection robustness.

The deliberate prioritization of low false negatives over low false positives, though appropriate for security-critical applications, may also contribute to user alert fatigue if the rate of false positives proves excessive in operational settings. Sustained high warning frequencies could gradually erode user trust and compliance, underscoring the need for adaptive thresholding and personalized calibration strategies. Finally, adversarial robustness remains a critical vulnerability. Malicious actors may engineer URL perturbations to circumvent learned patterns, highlighting the importance of continuous model retraining and the incorporation of adversarial training paradigms to preserve long-term effectiveness. Collectively, these considerations clarify both the practical value and the methodological boundaries of the proposed AI-powered phishing URL detection system with real-time alerting and a threat repository.

Overall, the discussion highlights that the proposed AI-powered phishing URL detection system—combining a lightweight Logistic Regression classifier, real-time alerting, and a structured threat repository—offers a practical and service-aligned solution compared with more computationally intensive alternatives. It delivers high recall in contexts where missed detections are particularly costly, maintains low latency suitable for real-time service workflows, and provides transparency and adaptability that support long-term use in evolving digital ecosystems.

6. Conclusion and Future Work

This study has introduced a modular AI-powered framework for phishing URL detection, as summarized in Table 3, designed to enhance security in digital service platforms such as e-commerce and logistics. The architecture combines a lightweight Logistic Regression classifier with real-time alerting, a structured threat repository, and a conditional verification mechanism that selectively consults external threat intelligence after the model's initial decision. Experimental evaluation on a benchmark dataset indicates that the machine learning approach can provide effective, low-latency detection suitable for service-oriented contexts while preserving interpretability and operational simplicity.

The proposed study offers three principal contributions. First, it develops a lightweight phishing URL detection model based on Logistic Regression, enabling efficient and interpretable classification of URLs at scale. Second, it introduces a real-time alerting mechanism that enhances user awareness and supports immediate responses to detected threats at the point of interaction. Third, it proposes a structured threat repository designed to support continuous learning, pattern analysis, and knowledge sharing within the cybersecurity domain, with a particular focus on threat tracking and user awareness.

In addition, the conditional verification architecture leverages external threat intelligence only when needed, increasing confidence in high-risk or ambiguous cases without imposing unnecessary latency or external dependency on all traffic. Together, these elements outline a practical framework for addressing phishing attacks in dynamic service ecosystems and contribute to broader goals in logistics, informatics, and service science.

At the same time, the current system should be regarded as a prototype and proof of concept rather than a fully validated production solution. The architecture is designed with objectives such as reliability, security, scalability, and high availability in mind, but these nonfunctional properties are not directly measured in this study and therefore cannot be claimed as guaranteed outcomes. The empirical evaluation relies on a single labeled dataset constructed from known phishing sources and benign traffic, which introduces potential sampling and temporal biases and may limit generalizability to future or region-specific attack patterns. Moreover, the present implementation operates exclusively on lexical URL features, prioritizing interpretability and computational efficiency while omitting richer content, visual, or behavioral signals that could strengthen resilience against sophisticated obfuscation and zero-day attacks. System-level aspects of the external verification, persistence, and alerting modules have been assessed mainly at the architectural and prototype levels, so critical nonfunctional properties—including scalability under production workloads, adversarial robustness, user experience impacts, and long-term behavioral effects—remain to be quantified through large-scale deployment studies.

These limitations suggest several concrete directions for future work. First, model generalization should be examined across multiple heterogeneous phishing URL datasets, including cross-platform and cross-regional collections, to better understand transferability and robustness. Where deployment constraints permit, ensemble techniques and more advanced models, such as deep learning architectures, could be explored as complements to the Logistic Regression baseline, provided that inference latency remains compatible with real-time service requirements. Second, the detection pipeline can be extended toward multimodal analysis by incorporating webpage content inspection, visual similarity measures, and linguistic or interaction patterns, thereby improving robustness against attacks that rely on design mimicry or compromised legitimate domains. Third, adversarial robustness can be strengthened by adopting adversarial training strategies, model diversification, and continuous learning mechanisms informed by feedback from the threat repository and user interactions.

Future work should address explainability, scalability, and user-centric evaluation more systematically. Integrating explainable AI techniques would provide clearer rationales for detection decisions to security analysts and end users, supporting transparency and trust, especially in high-stakes service environments. Large-scale deployment studies in collaboration with e-commerce platforms, financial institutions, and logistics providers are needed to validate performance, availability, and scalability under realistic traffic volumes and operational heterogeneity. In parallel, field studies should assess how alerting mechanisms and repository access influence user decision-making, threat awareness, and behavioral adaptation over time, including the management of false positives to mitigate alert fatigue. Finally, future extensions will need to engage more explicitly with data protection, ethical considerations, and regulatory compliance—particularly in cross-jurisdictional deployments that must align with frameworks such as GDPR and sector-specific standards.

Within these boundaries, the proposed AI-powered phishing URL detection system with real-time alerting and a threat repository establishes a substantive foundation for phishing mitigation in service ecosystems. Continued refinement through rigorous empirical validation and iterative enhancement along the outlined dimensions will be essential to evolve this prototype into a production-ready security platform capable of delivering measurable improvements in the security and trustworthiness of digital service environments.

References

Aburrous, N., Hossain, M. A., & Al-Khatib, A. B. (2019). A survey of URL-based phishing detection. *The Database Engineering and Information Management Conference (DEIM)* (pp. 1–12).

Afriyie, J. K., Boateng, M., & Danso, A. (2023). A supervised machine learning algorithm for detecting fraudulent credit card transactions. *Results in Engineering*, 18, Article 101342. https://doi.org/10.1016/j.rineng.2023.101342

Akhtar, A., Shah, M. A., Hameed, S., Khan, M. A., & Tariq, M. U. (2025). Mitigating cyber threats: Machine learning and explainable AI for phishing detection. *VFAST Transactions on Software Engineering*, 13(2), 170–195. https://doi.org/10.21015/vtse.v13i2.2129

Kurniawan, A. K., Sri Widodo, & Maulindar, J. M. (2025). Fraud detection in health insurance claims based on artificial intelligence (AI). Engineering and Technology Journal, 10(2), 3735–3739. https://doi.org/10.47191/etj/v10i02.01.

Awadallah, A., Eledlebi, K., Zemerly, M. J., Puthal, D., Damiani, E., Taha, K., & Al Hammadi, Y. (2024). Artificial intelligence-based cybersecurity for the metaverse: Research challenges and opportunities. *IEEE Communications Surveys & Tutorials, 27*(2), 1008–1052. https://doi.org/10.1109/COMST.2024.3442475.

Banerjee, P., & Das, A. (2024). Detection of phished URLs using machine learning. Journal of Web Engineering Technology, 15(3), 101–115.

Barik, K., Misra, S., & Mohan, R. (2025). Web-based phishing URL detection model using deep learning optimization techniques. *International Journal of Data Science and Analytics*, 20, 4449–4471.

Bilot, T., Geis, G., & Hammi, B. (2022). PhishGNN: A phishing website detection framework using graph neural networks. *In Proceedings of SECRYPT* (pp. 221–229).

Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. *In Proceedings of the IEEE Symposium on Security and Privacy* (pp. 39–57).

Charishma, T. N. S., Koushik, A. S., Reddy, G. S. A., & HimaBindu, M. (2024). Employing machine learning algorithms to detect phishing URL websites. *In Proceedings of ICICNIS* (pp. 1553–1558).

Chen, L., & Kumar, S. (2025). Blockchain-enabled fraud prevention in sustainable service supply chains. *Journal of Service Innovation and Sustainable Development*, 8(2), 112–128.

Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Systems with Applications*, 41(10), 4915–4928.

Deshmukh, P. G., & Waghmare, V. R. (2021). A survey on credit card fraud detection techniques. *International Journal of Scientific & Technology Research*, 10(6), 45–51.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *In Proceedings of NAACL-HLT* (pp. 4171–4186).

Federal Trade Commission. (2025). *Consumer Sentinel Network data book* 2024.

Garera, S., Provos, N., Chew, M., & Rubin, A. D. (2007). A framework for detection and measurement of phishing attacks. *In Proceedings of WORM* (pp. 1–8).

Harshitha, S., Vidhyashree, N. P., & Shetty, S. S. (2023). Malicious URL detection using logistic regression. *In Proceedings of CONECCT* (pp. 1–6).

Hu, Y., Zhang, Z., Luo, X., & Wang, J. (2023). Ethereum phishing account detection based on graph neural networks. *IET Blockchain*, 3(2), 87–97.

INTERPOL. (2022). 2022 INTERPOL *global crime trend summary report*. INTERPOL. https://www.interpol.int/en/content/download/18350/file/Global%20Crime%20Trend%20Summary%20Report%20EN.pdf

INTERPOL. (2026). INTERPOL *global financial fraud threat assessment. INTERPOL. https://www.interpol.int/en/Crimes/Financial-crime/Financial-crime-initiatives*

Kalal, T., & Shriram, S. S. (2024). E2Phish: Explainable ensemble machine learning model for enhanced phishing URL detection. *In Proceedings of CICT* (pp. 1–6).

Luo, C., Lin, H., Zhou, X., & Wang, Y. (2024). Fraud detection via context encoding and adaptive aggregation graph neural networks. *Expert Systems with Applications*, 247, Article 123648.

Mienye, I. D., & Jere, N. (2024). Deep learning for credit card fraud detection: A review of algorithms, challenges, and solutions. *IEEE Access*, 12, 96893–96910.

Murti, Y. S., & Naveen, P. (2023). Machine learning algorithms for phishing email detection. *Journal of Logistics, Informatics and Service Science*, 10(2), 249–261.

Mushayakarara, R. T. (2024). Evaluating machine learning models for effective phishing URL detection (*Master's thesis, National College of Ireland*).

Nguyen, T. H., Gregar, S., & Tran, M. H. (2019). Detecting financial statement fraud using machine learning techniques. *Journal of Risk and Financial Management*, 12(3), Article 109.

Pathmanaban, J., James, P. G., Ashok, P., Ragesh, B., Aakash, S., & Kaushik, N. (2024). Phishing website detection using machine learning. In *Proceedings of the 2024 2nd International Conference on Networking and Communications (ICNWC)*. IEEE.

Pratama, N. A., & Putra, D. S. (2024). Comparison of logistic regression and random forest using feature selection for phishing website detection. *SISTEMASI: Jurnal Sistem Informasi*, 13(4), 987–994.

Rahman, H. A., Sahoo, S. K., & Khan, M. I. (2025). An efficient phishing website detection plugin service based on website trustworthiness. *International Journal of Cybersecurity Privacy*, 9(8), 3569–3580.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? Explaining the predictions of any classifier. *In Proceedings of KDD* (pp. 1135–1144).

Kulkarn, A., Balachandran, V., Divakaran, D. M., & Das, T. (2025). From ML to LLM: Evaluating the robustness of phishing web page detection models against adversarial attacks. Digital Threats: Research and Practice, 6(2), 1–25. https://doi.org/10.1145/3737295

Roy, A. A., & Sunitha, J. (2021). Fraud detection in credit card transactions using machine learning algorithms. Journal of Physics: *Conference Series*, 1916(1), Article 012129.

Safi, A., Alqahtani, H., & Aljohani, A. (2023). Phishing website detection: A systematic review. *Ain Shams Engineering Journal*, 14(3), Article 101938.

Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, 117, 345–357.

Shafin, S. S., Khan, F. M., & Islam, A. (2024). An explainable feature-selection framework for phishing site detection. *Discover Internet of Things*, 4, Article 17.

Sharma, H., & Gupta, R. (2023). Malicious URL detection using logistic regression. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 11(7), 1234–1242.

Shendkar, B. D., & Chincholkar, A. M. (2024). Enhancing phishing attack detection using explainable AI. *ASEAN Journal of Science and Technology for Development*, 41(1), 23–33.

Shumailov, A., Anderson, R., & Papernot, N. (2022). The evasion-space of adversarial attacks against phishing website detectors. *In Proceedings of CCS* (pp. 2201–2214).

Songailaitė, M., & Laurinavičius, J. (2024). BERT-based models for phishing detection. *CEUR Workshop Proceedings*, 3779, 87–96.

Sudar, K. M., Gopal, R., & Rao, S. N. (2025). Detection of adversarial phishing attacks using machine learning. *Sādhanā,* 50(3), Article 167.

Tiwari, T. (2020). *Phishing site URLs* [Dataset]. Kaggle.

https://www.kaggle.com/datasets/taruntiwarihp/phishing-site-urls

Vajrobol, V., Gupta, B. B., & Gaurav, A. (2024). Mutual information based logistic regression for phishing URL detection. *Cyber Security and Applications*, 2, Article 100044.

Vinayakumar, R., Alazab, M., Soman, K., Poornachandran, P., & Venkatraman, S. (2018). Robust intelligent phishing website classification using deep learning and SVM. *Future Generation Computer Systems*, 87, 172–184.

World Economic Forum. (2026). *Global cybersecurity outlook 2026*. World Economic Forum. https://www.weforum.org/publications/global-cybersecurity-outlook-2026

Xuan, C. D., Nguyen, H. D., & Nikolaevich, T. V. (2020). Malicious URL detection based on machine learning. *International Journal of Advanced Computer Science and Applications*, 11(1), 148–157.

Yuan, Y., Zhang, Z., Li, J., & Wang, X. (2024). Are adversarial phishing webpages a threat in reality? *In Proceedings of the ACM Web Conference* (pp. 2955–2965).

Priya, S., Singh, R., Kumar, A., Kumar, K., & Sharma, V. (2025). Evolution of artificial intelligence in phishing detection: A comprehensive study. International Journal of Scientific Development and Research, 10(10), 20–23. ISSN 2455-2631.

Zhang, Z., Li, P., Zhou, J., & Sun, Y. (2023). Graph neural networks for fraud detection: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(5), 5123–5136.