JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

PhishDec: A Modular Multi-Agent AI Architecture for Autonomous Phishing Detection and Incident Triage

[1]Fardeen Jamshed Shaikh, [2]Nirbhay Singh
[1]M.Sc. CS Specialization in Cybersecurity, [2]Assistant Professor
[1,2]Department of Advanced Computing
[1,2]Nagindas Khandwala College,
[1]University of Mumbai, Maharashtra, India

ABSTRACT: Phishing attacks have become increasingly elusive and polymorphic, targeting blind spots in legacy defensive measures and techniques of cloaking and infrastructure-level obfuscation [4, 17, 18]. In practice, this means that many state-of-the-art approaches operate with either single agent-single modality analysis (i.e., URL lexical features or URL/DOM inspection) [8, 9] or shallow ensemble or deep learning classifiers with poor interpretability or insufficient contextualization for Security Operations Center (SOC) triage [10, 21]. In this paper, we present PhishDec: a modular multi-agents artificial intelligence (AI) architecture for the purposes of phishing detection, multi-vector evidence analysis, and security incident response [2, 3, 5].

PhishDec operates by splitting the task of analyzing potential phishing resources along several evidentiary domains (URL mechanics, DOM/HTML, brand visual identity, and content semantics), with each domain being explored by a dedicated specialist AI agent providing structured findings [1, 2, 3]. An adaptive consensus mechanism then evaluates and combines the results from these domains, with its operation differing fundamentally from prior multi-agent methods in that it enables coordinated debate-style reasoning between dissimilar and potentially conflicting evidence domains [3, 6].

In addition, PhishDec integrates model-level feature attribution and Large Language Model (LLM)-aided reasoning to produce both human-explained findings and structured mitigation and response recommendation, including but not limited to the inference of multi-vector attack graph from a single seed URL for the SOC to prioritize response [2, 4, 5]. We find that the proposed architecture is capable of operating on both phishing-related datasets and unknown campaigns, demonstrating its utility in an operational SOC environment. In this work, we focus on the performance of the evidentiary and architectural aspects of the system, evaluating the architecture's performance on detection, false positives, adaptability to emerging attacks, and the quality of generated triage artifacts [1, 12, 19].

Keywords: Phishing Detection, Multi-Agent AI, Explainable AI, Large Language Models, Cybersecurity, Threat Intelligence, Multi-Vector Analysis, Feature Engineering, Adaptive Consensus, Incident Response, SOC Triage, Infrastructure Analysis, Zero-Day Detection, Cloaking Detection.

1. INTRODUCTION

Phishing remains a critical cybersecurity threat and attackers continue to improve phishing kit technology and leverage deceptive URLs, dynamic web content, brand impersonation, infrastructure reuse, and cloaking to bypass automated detection measures [4], [17], [18], [20]. The state-of-the-art phishing detectors focus on detecting attacks based on a single source of evidence such as URL lexical features, HTML/DOM structure, or content, which leads to significantly reduced efficacy when attackers use multiple sources of attack data simultaneously [8], [9], [14]. Although recent works showed the potential of jointly analyzing multiple data sources using hybrid and machine learning-based approaches to improve raw detection rates [1], [16], [21], language model (LLM) based and multi-agent reasoning systems offer more flexible and explainable decision-making by taking into account multiple lines of evidence through distinct reasoning tracks and agent debate [2], [3], [6]. However, these systems still pose challenges associated with analyzing multi-source data from both infrastructure and client perspectives, extracting actionable insight across heterogeneous evidence domains, providing justification that incorporates relevant cybersecurity concepts, and operating efficiently in the face of rapidly changing phishing infrastructure [2], [4], [5], [13].

In this paper, we present PhishDec, a modular and explainable multi-agent system for multi-vector phishing detection and triage. PhishDec is designed to operate across multiple attack vectors by partitioning the analysis into seven distinct evidence domains such as URL/Lexical, HTML/DOM Structure, Visual Brand Identity, SSL/TLS, Passive DNS/Network, Cyber Threat Intelligence (CTI), and Semantic Content. Each domain is analyzed by a modular agent that extracts relevant features and produces findings which are then combined by an arbitrator agent to produce final detection results. The system uses an arbitration mechanism inspired by recent advances in multi-agent debate and large language modelbased reasoning to enable agents to negotiate and find consensus based on evidence rather than making decisions based on majority voting [2], [3], [5]. By design, the system enables analysts to understand and interpret results at different stages of the detection pipeline. When combined with infrastructure-based detection, the multi-agent architecture provides improved coverage, particularly in cases when the content of the underlying webpages is unavailable or manipulated (e.g., cloaked or hidden), a scenario when infrastructure-only approaches demonstrated consistently improved results [4]. Lastly, the system provides model-level explanations through feature attribution analysis and includes reasoning tracks that generate natural language explanations and remediation recommendations, improving operational relevance by aligning findings and suggestions with analyst expectations and operational procedures [2], [13], [15]. Overall, the proposed system provides a flexible detection architecture that can ingest and analyze data from multiple attack vectors, presents findings in a modular fashion to support human-in-the-loop analysis, and generate actionable takeaways for security analysts.

2. LITERATURE REVIEW

2.1 Existing Tools and Methodologies

Phishing detection methods have evolved from basic black- and heuristic-lists to machine learning, deep learning, visual, and infrastructure-based techniques. While list-based scanners offer fast detection of previously reported phishing domains, they are reactive and provide little protection against newly created or rapidly rotated phishing URLs [8], [17], [18]. Machine- and deep learning algorithms offer better performance on URL, HTML, lexical, and structural phishing features, though model accuracy remains vulnerable to poor feature engineering, distribution shift, and low interpretability [7], [13], [21]. Visual and content-based detection approaches provide strong brand impersonation and design similarity analysis, but require access to the webpage content which attackers often prevent using cloaking mechanisms or selective bot detection [4], [17]. These single-domain approaches have motivated the recent rise of hybrid and multi-source detection techniques that combine different evidence types rather than relying on any single observation channel [1], [16].

2.2 Related Research Papers

Recent works have explored different aspects of multi-domain, infrastructure-aware, and large language model (LLM)-based phishing detection. Jadhav and Chandre [1] proposed a multi-modal hybrid approach that combines transport layer (TLS), URL, and email content features in an OSI-layer aligned multi-cluster deep learning architecture. Ariyadasa et al. [7] used LRCN and GCN deep learning models to analyze both sequential (URL) and graphical (HTML) phishing content. Koide et al. [6] demonstrated that LLM-based approaches can extract useful features from rendered HTML, URL, and even TLS messages to detect potential phishing sites. Yadav and Masum [2] developed a multi-agent LLM system for phishing email detection, where different agents process linguistic, manipulative, and sender authentication features in parallel to achieve higher model interpretability. Li et al. [3] took a similar multi-agent approach, but applied it to different URL-based phishing detection features, letting multiple LLM-based agents argue (with the help of a moderator) about the URL's authenticity before arriving at a final judgment. Chiba et al. [4] applied the multi-agent LLM concept to infrastructure-based phishing detection, using a "cloaking" approach to associate potentially cloaked domains with known attack infrastructure through shared hosting, certificates, and network scans to create new mitigation rules. Finally, Hmimou et al. [5] used multi-agent LLM reasoning to associate different cyber threats beyond phishing, finding that rule- and signature-based systems were insufficient for detecting spear-phishing and multi-stage cyberattacks. Farzaan et al. [11] took a broader look at AI-assisted cybersecurity operations, covering phishing detection and response as a part of wider cyber threat hunting operations.

Overall, the prior research suggests that there is considerable room for improvement in all three areas: multi-domain feature fusion, multi-agent LLM processing, and infrastructure correlation. At the same time, none of the prior works attempt to combine all three approaches into a single system, despite the clear benefits of such an architecture. The PhishDec system described in this paper aims to fill this void by incorporating seven different types of evidence, from traditional URL features to visual design analysis, certificate inspection, DNS monitoring, and CTI intelligence, and using both debate-based LLM reasoning and feature-level explainability to arrive at its conclusions [2], [3], [4], [5], [13], [15].

3. PROBLEM STATEMENT

Modern phishing attacks use polymorphic URLs, dynamically rendered or cloaked content, brand impersonation and rapidly flipping infrastructure to evade conventional detection mechanisms [8], [17], [18], [20].Although significant advances are being made, the literature has three concrete gaps that we address here:

Fragmented evidence analysis: Current detection literature operates on a single evidence source at a time - URL lexical structure, HTML/DOM content, visual

similarity, or content semantics - rather than combining observations from multiple attack layers. This creates attack surface for adversaries who can distribute indicators of compromise across multiple domains at once, a problem acknowledged directly by recent hybrid-framework work that advocates multi-source correlation over single-source analysis [7], [13], [16].

Limited evasion and non-arbitrated reasoning: While recent LLM-based and multi-agent approaches have improved contextual understanding of phishing content [2], [3], [6], and infrastructure-pivoting techniques have shown that detection can be effective even when page content is withheld or tampered with [4], there are still open questions around how to reconcile conflicting agent-level predictions through principled arbitration, and around how to extend multi-agent reasoning from individual detection tasks to domain-wide threat correlation [5].

Insufficient operational actionability: Even when detections are accurate, they may be of limited value to the analyst if they are not interpretable or traceable to specific evidence. Multi-agent and LLM-based approaches have begun addressing this - PhishDebate's structured debate creates interpretable rationales that reduce analyst cognitive load [3], and PhishLumos produces reusable mitigation artifacts to enable hunting, blocking, and takedown [4] - but these capabilities are currently limited to the evidence domain of the individual approach, rather than being combined in a single multi-vector analysis [2], [4].

These gaps motivate our approach to create an autonomous phishing detection framework that can (i) integrate heterogeneous telemetry across independent evidence domains [1], [16], (ii) adaptively reconcile conflicting or partial evidence rather than using static voting approaches [3], [5], and (iii) convert detection results into actionable SOC-focused findings [2], [4]. We address this problem in this work with PhishDec, which models a phishing candidate using URL, HTML/DOM, visual, SSL/TLS, passive DNS, cyber threat intelligence (CTI), and semantic evidence, and produces a consolidated detection decision as well as supporting explanations and structured response information.

4. OBJECTIVES OF THE RESEARCH

The main objective of this research is to formulate, implement, and evaluate PhishDec, an explainable, modular multi-agent AI architecture for multi-vector phishing detection and SOC-oriented incident triage. In pursuit of this goal, this work aims to develop a multi-domain telemetry pipeline that covers URL, HTML/DOM, visual, SSL/TLS, passive DNS, cyber threat intelligence (CTI), and semantic evidence, building on prior multi-domain fusion work that combined URL, network, and email-layer telemetry into a single detection architecture [1]. A related objective is to design independent feature-extraction and role-specialized analysis agents for heterogeneous phishing indicators, leveraging demonstrated agent-specialization patterns like role-decomposed linguistic, psychological, and sender-identity analysis [2] and structure-specific agents targeting URL, HTML, semantic content, and brand impersonation [3]. Building from this, the research will construct an adaptive consensus mechanism for evidence fusion and conflict resolution across specialist agents informed by debate-style arbitration [3] and multi-agent correlation of heterogeneous, potentially-conflicting security evidence [2], [4], [5]. Furthermore, the work will build statistical feature attribution and LLM-assisted analysis to deliver interpretable detection evidence and structured mitigation recommendations in line with explainability-focused multi-agent design principles [2] and campaign-level mitigation-artifact generation for analyst support [4], [15]. Finally, the research will conduct an empirical evaluation of the proposed architecture against appropriate baseline methods on the basis of detection performance, false-positive behavior, inference latency, robustness against previously-unseen phishing samples and quality of SOC triage outputs as criteria - informed by prior evaluation methodologies for multi-domain hybrid detection [1], emerging and zero-day threat classification [12], hybrid-framework applicability testing [16] and phishing taxonomy and evaluation frameworks in the broader detection literature [6], [19].

5. RESEARCH GAPS ADDRESSED BY PHISHDEC

Gap 1: Single-Modality and Feature-Siloed Detection

Current phishing detection research frequently focuses an analysis on a single evidence source – URL, HTML, visuals, or content – without attempting to combine complementary hints across attack layers. This pattern is visible in the literature: feature-engineering surveys observe consistent emphasis on URL/lexical features in the face of growing evidence that phishing detection benefits from multi-modal input [7], [8], [13], [17]. PhishDec addresses this gap by operating across seven telemetry domains – URL/Lexical, HTML/DOM, Visual Brand Identity, SSL/TLS, Passive DNS/Network, CTI, and Semantic Content – with analysis carried out by independent agents rather than a monolithic feature space.

Gap 2: Content-Inaccessible and Evasive Phishing

Content-based detection is challenged by situations where cloaking or similar obfuscation techniques make it impossible for an automated agent to obtain a representative view of the page's contents [4], [6], [17]. PhishDec addresses this gap by separating the analysis of webpage content from infrastructure-level evidence, using DNS, SSL/TLS, and CTI as supplements or alternatives to traditional content scans and achieving similar performance when the target page's content is unavailable, cloaked, or otherwise inaccessible, reflecting both the infrastructure pivoting technique for cloaked target domains [4] and multi-agent correlation of cross-domain security-relevant evidence [5].

Gap 3: Limited Explainability and SOC Actionability

While machine- and deep-learning classifiers can achieve strong performance on phishing detection tasks, they frequently lack the capacity to explain or justify an individual decision, and LLM-based reasoning requires appropriate prompting to be effective in a security context [6], [15]. PhishDec addresses this gap by complementing statistical features importance with LLM assisted analysis, which adheres to schema-guided prompts, yielding not only the final classification but also justification and mitigation advice in a manner useful to a Security Operations Center, reflecting an extension of explainable AI and multi-agent analysis literature focussing on actionable outputs [2] and generative AI techniques applied to security domains [5], [15].

Gap 4: Static Evidence Fusion and Conflict Resolution

Ensemble learning techniques are often limited to static weights or simple voting logic, but recent advances in multi-agent machine learning indicate that there may be more flexible ways to combine agents to improve results. This includes using multiple agents to analyze the same input, structured debate between agents, and using LLM based reasoning to achieve fusion of multiple heterogeneous cybersecurity relevant signals [3], [5]. PhishDec tackles this gap by implementing an ad hoc consensus mechanism that combines the outcomes of the individual analyses and resolves conflicts based on contextual information in order to arrive at a final classification, reflecting related work in multi-agent cybersecurity applications [2] and adversarial question-answering strategies [3].

Gap 5: Adaptation to Evolving and Previously Unseen Threats

Classical blocklist and signature-based approaches, are only effective against known threats, while machine-learning approaches are sensitive to shifts in the input distributions and require continuous retraining in the face of evolving phishing techniques [8], [17], [18], [20]. PhishDec bridges this gap by using modular analysis combined with CTI and infrastructure level features, in line with recent advancements in identification of emerging threats that utilizes topic modeling and pattern recognition to identify inputs that differ from known classes, and allowing the system to evaluate an unknown sample's potential impact without requiring extensive prior training data [12].

6. PROPOSED PHISHDEC ARCHITECTURE

PhishDec is an orchestrator framework consisting of heterogeneous multi-agent architecture that consumes and processes various kinds of phishing evidences such as URL/redirection, HTML/DOM content, visual artifacts, SSL/TLS certificate information, passive DNS records, CTI indicators, and semantic content through the sequential steps of collection, feature extraction, specialized analysis, fusion, explanation, and SOC-ready outputs, extending the recent approaches that combine hybrid and infrastructure-aware evidence for phishing detection [1], [4], [5], [16]. Particularly, the system consists of an API-facing orchestrator that takes a target URL and initiates asynchronous collection of URL/redirection evidence, HTML/DOM content, visual artifacts, SSL/TLS certificate information, passive DNS records, CTI indicators, and semantic content. Then, a feature extraction layer transforms the collected evidences into standard representations such as lexical/entropy-based URL features, DOM structure, vision/OCR features, certificates' attributes, DNS metadata, CTI indicators, and semantic features [7], [13], [17] that are fed into seven role-specific agents (URL/Lexical, HTML/DOM, Visual Brand Identity, SSL/TLS, Passive DNS/Network, CTI, Semantic Content). Each agent produces a claim, confidence level, and supporting evidence in accordance with the multi-agent phishing research emphasizing the necessity of role specialization [2], [3], [5] and goes beyond the prior work that only performed content-only analysis by large language models [6]. A decision fusion component conducts adaptive evidence-weighting and aggregation, resolving conflicting inferences through context-based arbitration instead of using fixed weights or simple majority voting [2], [3], [5]. The fused results are delivered to an explainability and SOC-triage component that utilizes feature-attrubution techniques to identify the most influential features and applies an LLM-assisted process to produce natural-language explanations and remediations [2], [15].

Finally, the system provides a triage-ready detection result through the REST API, SOC dashboard, or report for the downstream usage [11], [12]. As the first step in the architecture, we integrated both client-side and infrastructure-level evidences that can be effective in detecting phishing sites even when the page content is not available or deceptive [4].

The proposed architecture can support the future works such as the improvement of detection accuracy, robustness, explainability, and SOC triage usefulness.

7. DATA COLLECTION AND FEATURE EXTRACTION

PhishDec ingests and normalizes heterogeneous telemetry from seven dimensions related to the security of a website to create an evidence representation (see Table I). The URL/Lexical group consists of features that represent the characteristics of the URL such as length, entropy, delimiter frequency, subdomain depth, pattern in characters, and features of the top-level domain [7], [13], [17]. The HTML/DOM group contains the features related to form destination, presence of hidden elements, iframe, script, link features, and hyperlink analysis [7], [13], [17]. The Visual/Brand group involves features extracted from the website using computer vision and screenshot analysis of the page to detect presence of logos, patterns, layout, brand impersonation [3], [6]. The SSL/TLS and Passive DNS/Network groups have elements such as certificate information, network infrastructure, and DNS information of the domain [4], [16]. The CTI features consist of reputation, blacklist, and malware association features of the domain in the system [2]. The

Semantic/Content group analyzes the contents of the website for features related to the text of the page such as presence of specific words related to phishing [6]. The various feature groups are normalized and used as input to the different agents of PhishDec to perform complementary analysis on the content of the page and the infrastructure [1], [4], [5].

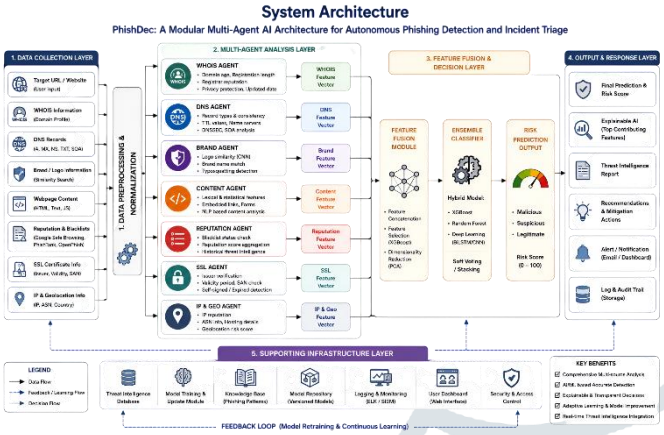

Fig 6.1: PhishDec System Architecture

8. MULTI-AGENT DECISION AND CONSENSUS MECHANISM

PhishDec compromises seven distinct agents (., URL/Lexical, HTML/DOM, Visual/Brand, SSL/TLS, Passive DNS/Network, CTI, and Semantic/Content) that analyze different evidences [2], [3], [5]. Each agent generates a schema-valid output that contains its verdict, risk score, confidence, and supporting indicators for downstream evidence aggregation. The decision engine applies domain-specific weights to fusion their outputs as depicted below, and computes a risk score S_fused,

$$S_{\text{fused}} = \frac{\sum_{i=1}^{7} w_i r_i}{\sum_{i=1}^{7} w_i}$$

which is then converted to actionable risk categories for the SOC to triage. In the event of significant disagreements - e.g., high-risk indication by one agent but others report benign - instead of simple averaging mechanism, the system performs contextual conflict arbitration, extending prior multi-agent approaches [2], [3], [5] for phishing and cybersecurity evidence analysis with specialized agents and reasoning coordination. The final decision is thus based on multi-vector evidence, but the individual agent confidence and supporting indicators are retained to enable explainability and mitigation-specific triage [2], [4].

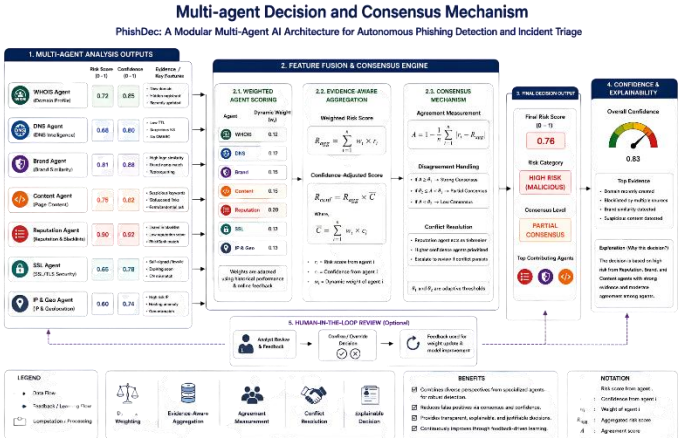

Fig 8.1: Decision Fusion and Consensus Mechanism

9. EXPLAINABILITY AND SOC TRIAGE

PhishDec implements a two-layered explainability methodology which combines statistical feature attribution with LLM-based evidence synthesis to enhance transparency and usability [2], [15]. In particular, SHAP/LIME pinpoint influential features which impacted the model's decision, while the LLM-based component consumes the distilled agent findings and transforms them into a human-readable explanation which is then presented to the SOC analyst. The triage record then contains the resulting risk assessment (confidence), supporting indicators (evidence), and the evidence provenance.

In addition, the triage layer transforms validated findings into first-class security artifacts (URLs, IPs, domains, fingerprints, and other IoCs with STIX-compatible representations, if available) to be used across the wider security stack [4], [5]. Finally, depending on the risk profile and the site's policies, the system can recommend mitigation actions, such as blocking domains/IPs, monitoring related infrastructure, and taking protective measures against credentials, thus transforming phishing detection into actionable incident triage [11], [12].

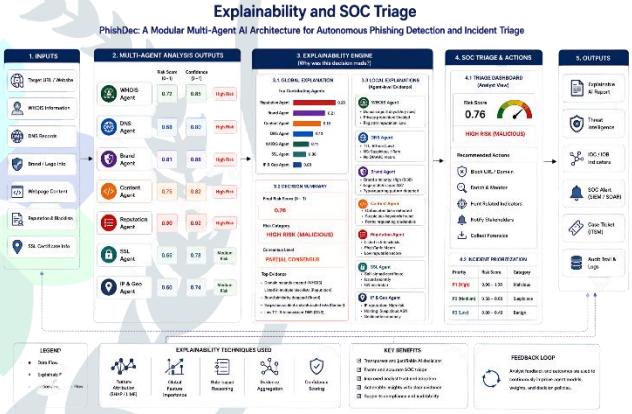

Fig 9.1: EXPLAINABILITY AND SOC TRIAGE

10. EXPERIMENTAL METHODOLOGY

The PhishDec experiments are conducted on heterogeneous phishing data and baselines, covering detection, robustness, explainability, and efficiency.

A. Research Environment: The system is implemented in Python 3.10 on Linux with deep learning framework TensorFlow/Keras, classical ML and evaluation scikit-learn, HTML/DOM parsing BeautifulSoup4, semantics Transformers/spacy, and explanation SHAP/LIME features, and GPU-accelerated deep learning and vision.

B. Datasets: The system is evaluated on heterogeneous phishing and legitimate samples with URL, HTML/DOM, visual, semantics, SSL/TLS, passive-DNS, and CTI features, including PhishTank, OpenPhish, benchmark, legitimate-domain, and email corpora samples for semantic-agent phishing analysis. Multiple datasets are employed to enable system evaluation independent of any specific benchmark, which is common in phishing detection [8], [13], [17], [18].

C. Preprocessing: The URL lexical and structural features are extracted with tokenization, HTML/DOM using parser DOM traversal for forms, hyperlinks, scripts, and iframes, text is tokenized, lemmatized and embedded, and the pages are rendered on browser for visual inspection [7], [13], [16].

D. Train/Test Strategy: An 80/10/10 stratified sampling is employed to preserve class distribution, with duplicates or cross-fold contamination removed. A temporal or independent unseen sample subset is used as a robustness test set [13], [16], [17].

E. Baseline Comparisons: PhishDec is compared against Random Forest, SVM, Logistic Regression, XGBoost tree-based and CNN/LSTM autoencoder deep learning models, GCN graph neural networks, static ensemble voting, and LLM-based standalone detection, representing classical ML, deep learning, ensembles, graph-based, and LLM approaches [6], [7], [13], [16].

F. Experimental Scenarios: The experiments are conducted across three scenarios: (1) standard in-distribution detection, (2) content inaccessible scenario where infrastructure evidence is used as surrogate for unavailable or restricted content [4], and (3) distribution shift test with unseen or temporally separated samples as a robustness probe [17], [20].

G. Evaluation Metrics: Accuracy, precision, recall, F1-score, false positive rate, ROC-AUC, latency, and statistical significance are reported, as individual metrics can be misleading, particularly with class imbalance [8], [13], [18]:

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall}$$

11. RESULTS AND EVALUATION

The evaluation of the PhishDec centers on the detection performance at agent level, decision fusion, and operational properties: class imbalance is a common problem in phishing datasets and accuracy is an inadequate metric for detection performance [8], [13], [17], [18]. The current evaluation of the trained XGBoost classifier works with 106,677 held-out test samples (17 features: 73,792 legitimate samples, 32,885 phishing samples) and reports 82.28% accuracy, 81.99% precision, 54.49% phishing recall, 65.47% phishing F1-score, and 88.45% ROC-AUC: the latter indicates excellent discrimination and precision, but current model requires tuning to raise phishing recall and lower false negatives.

A. Overall Detection Performance: The XGBoost baseline yields 69,855 TN, 3,937 FP, 14,967 FN, and 17,918 TP on the test set of 106,677 samples: ~5.34% FP rate and ~45.51% FN rate. While the model detects legitimate samples effectively, greater detection of phishing samples is needed and class imbalance problem motivates selection of alternate performance metrics beyond accuracy [13], [17], [18]. A separate Postman integration test demonstrates decision fusion from two active agents: the URL AI Agent returned a 93/100 risk score (93.01% phishing confidence), while the HTML AI Agent returned an 8/100 risk score (90% legitimate confidence). The resulting 1-1 vote conflict produced a 50/100 suspicious final risk score, which is consistent with the justification for multi-agent phishing analysis: specialized agents can produce complementary and conflicting evidence that must be resolved to obtain a final decision [2], [3], [5].

B. Error Analysis: The 3,937 FP (false positive) and 14,967 FN (false negative) errors most likely reflect, respectively, legitimate samples that exhibit phishing-like lexical/structural/infrastructure/visual properties, and phishing samples that have few or inconclusive observable evidence — this pattern is consistent with past work on detection effectiveness as a function of feature representation, dataset properties, and available evidence [4], [7], [13], [16], [17]. The 54.49% phishing recall is cited as the main limitation of the current implementation.

C. Agent-Level Ablation Analysis: A complete seven-agent ablation study (removing one domain at a time — URL/Lexical, HTML/DOM, Visual/Brand, SSL/TLS, Passive DNS/Network, CTI, Semantic/Content — and comparing recall, F1, FPR) has not yet been completed; no ablation values are reported at this time. This evaluation follows the modular evaluation principle established in recent multi-agent, multi-domain phishing research [2], [3], [5].

D. Consensus and Decision Fusion: The integration test (URL risk = 93/100, HTML risk = 8/100) produced one phishing and one legitimate vote, resulting in conflict = true and a 50/100 suspicious final decision — this demonstrates that conflicting agent decisions are passed to the fusion layer instead of reduced to the strongest vote, consistent with specialized evidence decomposition approaches [2], [3], [5]. This remains a functional system test rather than a statistical benchmark; quantitative comparison of adaptive consensus against majority voting and weighted fusion will wait on the complete multi-agent implementation, as individual agent or single-case results should not be taken as system-wide accuracy [2], [3].

E. Latency and Operational Performance: End-to-end inference latency (telemetry collection → feature extraction → agent execution → fusion → SOC-oriented output) is a topic for future measurement; efficient orchestration of heterogeneous analysis components is a design priority for automated detection and response [5], [11], [12]. Current records do not support a statistically reliable report of mean, standard deviation, or percentile latency, and no unsupported value is provided.

F. Robustness to Evasive and Unseen Phishing: Current results report baseline performance, but do not yet include quantitative evidence of robustness against content-inaccessible, evasive, or previously unseen phishing. Since phishing pages may use cloaking or selective access to prevent conventional retrieval, infrastructure-level pivoting (domains, IPs, certificates, historical observations) provides a documented complementary approach when webpage content is unavailable [4]; PhishDec's architecture includes both evidence types for this reason. Future evaluation will use temporally separated and distribution-shifted samples to test generalization, without making any

unsupported universal "zero-day" claims [17], [19], [20].

G. Statistical Evaluation: The 106,677-sample held-out evaluation provides point estimates of accuracy, precision, recall, F1-score, ROC-AUC, and confusion-matrix-derived error rates, but confidence intervals and paired significance testing ($\alpha = 0.05$) have not yet been reported and will follow repeated/paired baseline experiments. Multiple complementary metrics and statistically supported comparisons are preferable to reliance on accuracy alone [8], [13], [17], [18].

12. RESULTS AND EVALUATION

The PhishDec baseline of XGBoost was tested on 106677 instances with 17 features, resulting in 82.28% of accuracy, 81.99% of precision, 54.49% of recall, 65.47% of F1-score, and 88.45% of ROC-AUC. The confusion matrix had 69855 TN, 3937 FP, 14967 FN, and 17918 TP, which produced approximately 5.34% of false positive rate (FPR) and approximately 45.51% of false negative rate (FNR). Although the majority of instances were classified correctly, there is a room for improvement in terms of phishing recall. The issue of class imbalance in phishing data is well known and accuracy as a performance metric is considered misleading in such scenarios [8], [13], [17], [18]. Another experiment, which utilized two Postman agents in a single request-response transaction with an intentionally introduced conflict, resulted in 93/100 URL agent score and 8/100 HTML agent score, which produced a conflict of 1-1 votes and generated a suspicious 50/100 fused decision. This outcome is consistent with a body of research, which emphasizes the importance of evidence produced by multiple agents, which should be combined with care, as they may represent different aspects of the same problem and not necessarily in agreement [2], [3], [5]. It is important to note that this is still a far cry from definitive quantitative conclusions, as extensive model ablation, latency analysis, and distribution shift tests would be needed to confirm the benefits of the entire seven-agent architecture in practice [4], [19], [20].

13. DISCUSSION

A. Interpretation of Results: PhishDec showed potential to leverage heterogeneous evidence through its modular structure with different decision fusion levels. The base XGBoost model achieved 82.28% accuracy, 81.99% precision, 54.49% recall, 65.47% F1-score, and 88.45% ROC-AUC, suggesting adequate discrimination but the need to substantially improve phishing recall. This aligns with our multi-agent and infrastructure-focused prior work that highlighted the value of combining heterogeneous evidence and infrastructure indicators when webpage contents are inaccessible [2], [3], [4], [5].

B. Practical SOC Implications: We designed PhishDec to empower SOC analysts to correlate heterogeneous evidence with reduced manual effort and achieve interpretable results [5], [12]. The CTI output of PhishDec can be expressed with STIX 2.1 objects to enable SOC operationalization and interoperability, similar to infrastructure correlation and mitigation report generation in recent phishing detection systems [4]. The reduction in false positives and the time saved for SOC analysts would require further operational validation to quantify.

C. Strengths, Trade-Offs, and Limitations: The main strengths of PhishDec are the seven-vector evidence architecture, multi-agent modular pipeline, adaptive fusion, and two-layered explainability enabled by our prior work on multi-agent approaches and explainable detection [2], [3], [5]. These design decisions have allowed us to achieve broader evidence coverage than prior work. The main trade-offs involve complexity and greater computational demands compared to non-modular methods. This comes from the -agent pipeline, reliance on external APIs and the need for more infrastructure support. The main limitations of this work include the 54.49% phishing recall achieved by the performing baseline along with a lack of thorough ablation studies, latency measurements and testing, by third-party security operations centers.

14. CONCLUSION AND FUTURE WORK

A. Summary of Findings and Main Contributions: This research introduced PhishDec, a modular and explainable multi-agent architecture for phishing detection and SOC-oriented incident triage. The framework integrates seven evidence domains, role-specialized analysis agents, adaptive consensus, and explainability mechanisms to support heterogeneous phishing analysis, extending recent multi-agent and infrastructure-aware detection research [2], [3], [4], [5]. The current experiment baseline resulted in 82.28% of accuracy, 81.99% of precision, 54.49% of recall, 65.47% F1-score, and 88.45% of ROC-AUC.

B. Research Limitations and Future Improvements: The current evaluation is limited by the baseline model's relatively low phishing recall and the absence of complete ablation, latency, distribution-shift, and human-SOC studies. Future work will focus on improving recall through adaptive learning and multimodal evidence fusion, evaluating open-weight LLMs to reduce external API dependency, investigating online or federated adaptation to evolving phishing campaigns [19], [20], and conducting human-in-the-loop SOC studies to assess explainability and operational effectiveness. Infrastructure-aware analysis remains particularly promising for content-inaccessible phishing, as demonstrated by recent research on infrastructure pivoting under cloaking [4].

References

[1] Jadhav, A., & Chandre, P. R. (2026). Optimization-driven multi-domain hybrid framework for phishing detection across URL, network, and email layers using deep learning and heuristic intelligence. Research Square.
Link: https://doi.org/10.21203/rs.3.rs-9519176/v1

[2] Yadav, T., & Masum, M. (2026). Explainable multi-agent LLM framework for phishing email detection via role-specialized evidence decomposition. Electronics, 15(12), 2606.
Link: https://doi.org/10.3390/electronics15122606

[3] Li, W., Manickam, S., Chong, Y., & Karuppayah, S. (2026). PhishDebate: An LLM-based multi-agent framework for phishing website detection.
Link: https://arxiv.org/abs/2506.15656

[4] Chiba, D., Nakano, H., & Koide, T. (2026). PhishLumos: From a single URL to campaign-level phishing mitigation. IEEE Access, 14, 79716–79739.
Link: https://doi.org/10.1109/ACCESS.2026.3696597

[5] Hmimou, Y., Tabaa, M., Khiat, A., & Hidila, Z. (2025). A multi-agent system for cybersecurity threat detection and correlation using large language models. IEEE Access, 13, 150202–150218.
Link: https://ieeexplore.ieee.org/search/searchresult.jsp?newsearch=true&queryText=A%20multi-agent%20system%20for%20cybersecurity%20threat%20detection%20and%20correlation%20using%20large%20language%20models

[6] Koide, T., Nakano, H., & Chiba, D. (2024). ChatPhishDetector: Detecting phishing sites using large language models. IEEE Access, 12, 154381–154400.
Link: https://doi.org/10.1109/ACCESS.2024.3483905

[7] Ariyadasa, S., Fernando, S., & Fernando, S. (2022). Combining long-term recurrent convolutional and graph convolutional networks to detect phishing sites using URL and HTML. IEEE Access, 10, 82355–82375.
Link: https://doi.org/10.1109/ACCESS.2022.3196018

[8] Asiri, S., Xiao, Y., Alzahrani, S., Li, S., & Li, T. (2023). A survey of intelligent detection designs of HTML URL phishing attacks. IEEE Access, 11, 6421–6443.
Link: https://doi.org/10.1109/ACCESS.2023.3237798

[9] Wood, T. M., Basto-Fernandes, V., Boiten, E. A., & Yevseyeva, I. (2022). Systematic literature review: Anti-phishing defences and their application to before-the-click phishing email detection.
Link: https://arxiv.org/abs/2204.13054

[10] Kaushik, P., & Rathore, S. P. S. (2023). Deep learning multi-agent model for phishing cyber-attack detection. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9s), 680–686.
Link: https://ijritcc.org/index.php/ijritcc/article/view/7674

[11] Farzaan, M. A. M., Ghanem, M. C., El-Hajjar, A., & Ratnayake, D. N. (2025). AI-powered system for an efficient and effective cyber incidents detection and response in cloud environments. IEEE Transactions on Machine Learning in Communications and Networking, 3, 631–644.
Link: https://ieeexplore.ieee.org/search/searchresult.jsp?newsearch=true&queryText=AI-powered%20system%20for%20an%20efficient%20and%20effective%20cyber%20incidents%20detection%20and%20response%20in%20cloud%20environments

[12] Haile, A. T., Abebe, S. L., & Melaku, H. M. (2025). Real-time automated cyber threat classification and emerging threat detection framework. IEEE Open Journal of the Computer Society, 6, 921–930.
Link: https://doi.org/10.1109/OJCS.2025.3580235

[13] Kustiawan, Y. A., & Ghauth, K. I. (2025). Feature engineering for phishing website detection using machine learning: A systematic review. IEEE Access, 13, 192080–192104.
Link: https://doi.org/10.1109/ACCESS.2025.3630334

[14] Dumitras, A., Mocan, C. M., & Oprişa, C. (2024). A feature engineering approach for detecting phishing

emails. In 2024 IEEE 20th International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 125–132.
Link:
https://doi.org/10.1109/ICCP63557.2024.10793001

[15] Saidi, S., Yashvardhan, U., Chamola, V., & Sikdar, B. (2024). Generative AI for cyber security: Analyzing the potential of ChatGPT, DALL-E, and other models for enhancing the security space. IEEE Access, 12, 53502–53520.
Link: https://doi.org/10.1109/ACCESS.2024.3385107

[16] van Geest, R. J., Cascavilla, G., Hulstijn, J., & Zannone, N. (2024). The applicability of a hybrid framework for automated phishing detection. Computers & Security, 139, 103736.
Link: https://doi.org/10.1016/j.cose.2024.103736

[17] Zieni, R., Massari, L., & Calzarossa, M. C. (2023). Phishing or not phishing? A survey on the detection of phishing websites. IEEE Access, 11, 18499–18519.
Link: https://doi.org/10.1109/ACCESS.2023.3247135

[18] Safi, A., & Singh, S. (2023). A systematic literature review on phishing website detection techniques. Journal of King Saud University - Computer and Information Sciences, 35(2), 590–611.
Link: https://doi.org/10.1016/j.jksuci.2023.01.004

[19] Daoud, E., Garcia-Blas, J., Alawadi, S., & Carretero, J. (2026). Phishing in the age of distributed intelligence: Taxonomies, detection strategies, and the emerging role of federated learning. Progress in Artificial Intelligence.
Link: https://doi.org/10.1007/s13748-026-00448-6

[20] Kavya, S., & Sumathi, D. (2025). Staying ahead of phishers: A review of recent advances and emerging methodologies in phishing detection. Artificial Intelligence Review, 58, Article 50.
Link: https://doi.org/10.1007/s10462-024-11055-z

[21] Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Abu Elsoud, E. (2022). An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), 3819–3828.
Link: https://doi.org/10.1007/s10586-022-03604-4