Prompt Engineering for Detecting Phishing

Darrell L. Young[a], Eric C. Larson[a], and Mitchell A. Thornton[a]

[a]Darwin Deason Institute of Cybersecurity, Southern Methodist University, Dallas, TX 75205, USA

ABSTRACT

This paper introduces an **adversarial framework** that leverages two **Large Language Models (LLMs)** via **prompt engineering** to enhance phishing detection. One LLM functions as a **generator**, producing sophisticated phishing emails that mimic legitimate communications, while the other serves as a **discriminator**, detecting and classifying these emails and providing detailed reasoning for its decisions. By dynamically refining prompts based on adversarial interactions, this framework not only improves detection accuracy but also educates users on phishing indicators—helping reduce cognitive biases. Our results demonstrate a robust, adaptive defense against increasingly complex cyber threats.

Keywords: Phishing detection, adversarial framework, large language models, prompt engineering, user education, cybersecurity

1. INTRODUCTION

Phishing attacks have become a pervasive and rapidly escalating threat, compromising individuals, organizations, and governments worldwide. According to the *APWG Phishing Activity Trends Report for Q3 2024*,[1] there were over 932,923 reported phishing attacks in one quarter alone. This increase is compounded by the misuse of **Large Language Models (LLMs)** that enable attackers to generate highly convincing, personalized phishing emails at scale.[2] Such capabilities outpace traditional defenses and necessitate new adaptive detection methodologies.

The evolution of phishing has reached a critical tipping point. High-profile incidents demonstrate the catastrophic consequences of phishing attacks. Modern attackers employ LLMs to generate emails that appear authentic, making it challenging for the reactive network techniques shown in Table 1 to keep pace. AI-enabled zero-day attacks can filter through network defenses. The last defense is a combination of a text-based classifier and an educated user with the discernment to recognize possible phishing attacks and other forms of online deception.

1.1 Outcome Engineering for Enhanced User Education

A central long-term goal of this research is to improve user outcomes by enhancing their ability to recognize and appropriately respond to phishing attacks. We propose the concept of *Outcome Engineering* as a strategic framework for using AI to achieve tangible, measurable improvements in organizational and personal decision-making.[3] In our context, the desired outcome is a better-educated user—one who is equipped to discern phishing attempts and other forms of online deception with greater accuracy and confidence.

Outcome Engineering, as introduced in our concurrent work for the IEEE Dallas Circuits and Systems Conference, emphasizes the use of adaptive AI systems to drive improvements in human performance. By coupling the incremental classifier's rapid adaptation with the Detector's detailed, confidence-weighted explanations, our framework not only detects phishing attacks but also provides users with insights into the underlying red flags. This dual approach supports user education by offering clear, interpretable guidance on why an email is flagged as suspicious. Although the present work does not yet fully integrate continuous adaptive feedback into the Generator, the underlying principles of Outcome Engineering are evident in our efforts to create a system that not only achieves high detection accuracy but also directly contributes to enhanced user vigilance and informed decision-making.

Further author information: (Send correspondence to Darrell L. Young)
Darrell L. Young: E-mail: dlyoung@smu.edu

Together, these components demonstrate how combining adaptive machine learning with interpretable, explainable AI can lead to improved security outcomes. In future research, we aim to refine this approach further by deepening the integration between incremental learning and LLM-based reasoning, thereby advancing the broader vision of Outcome Engineering.

Table 1. Security Approaches and Descriptions

Security Approach	Description
DNS-Based Defenses:	Using advanced DNS-layer security to block malicious sites before a connection is established.
Secure Access Service Edge (SASE):	Securing network access for users regardless of location.
Endpoint Protection:	Employing machine learning and behavioral analysis to detect threats at the endpoint level.
Email Security:	Focusing on protecting against email-based threats with advanced threat detection and threat intelligence.
Zero Trust Security:	Enforcing continuous authentication and authorization to validate user access.
Extended Detection and Response (XDR):	Integrating data from multiple security products for coordinated threat response.

Text-based phishing detection often relies on extensive labeled datasets and model fine-tuning. However, fine-tuning is time-consuming, computationally expensive, and static in nature. Labeled datasets are expensive to produce and curate. In contrast, **prompt engineering** leverages in-context learning to dynamically adapt to new phishing tactics. This approach reduces computational costs and enables rapid integration of new threat intelligence—making it particularly suited for addressing the dynamic nature of phishing.

Our adversarial generator-detector LLM framework detects phishing through dynamic prompt engineering. The Detector not only classifies suspicious emails but also educates users by explaining the reasoning behind each classification. The detection prompt is optimized to achieve these twin goals of detection and education, using a custom metric in the DSPy Multiprompt Instruction Proposal Optimizer (MIPRO) v2 optimizer. The Declarative Self-improving Python (DSPy) framework transitions from traditional prompting to programming language models, enabling rapid iteration in building modular AI systems. It provides algorithms for optimizing prompts and weights across various applications, from simple classifiers to complex RAG pipelines and Agent loops.[4]

2. AGILITY OF PROMPT ENGINEERING WITH IN-CONTEXT LEARNING

Our approach harnesses a curated URL dataset as a surrogate for real-time threat intelligence. By leveraging detailed URL and HTML features collected in this dataset, we can emulate the dynamics of evolving cyber threats without the need for extensive retraining. This enables rapid adaptation to emerging phishing tactics and supports agile decision-making within our dual-LLM framework.

Real-time threat intelligence is critical for dynamic phishing detection. By using our URL dataset as a threat intelligence surrogate, the discriminator LLM can incorporate emerging phishing strategies—such as deceptive bank notifications or social media scams—directly into its analysis. This seamless integration allows the system to adjust its focus and improve detection performance while avoiding the overhead associated with continuous retraining.

The **PhiUSIIL Phishing URL Dataset** is a substantial dataset comprising 134,850 legitimate and 100,945 phishing URLs.[5] Most of the URLs analyzed in constructing the dataset are recent. Features are extracted from both the source code of the webpage and the URL; derived features such as *CharContinuationRate*, *URLTitleMatchScore*, *URLCharProb*, and *TLDLegitimateProb* enhance threat identification. We iterate through the

dataset in blocks. For each block, the prompt-optimized LLM phishing detector is evaluated; if the F1 Score falls below a predefined threshold, a prompt optimization cycle is initiated. Concurrently, the performance of the incremental learning–based URL phishing detector is computed. The strong performance of this incremental learning approach suggests that it can serve as an effective training aid for further LLM prompt optimization.

2.1 Incremental Learning

At the core of our system lies the principle of incremental learning, which enables the model to evolve and adapt through continuous data intake and learning, without the need for reinitialization. This approach is pivotal for:

- **Updating Detection Capabilities:** As new phishing strategies emerge, the system incrementally integrates this new information, enhancing its detection algorithms.
- **Reducing Model Staleness:** By continuously learning from new data, the model remains up-to-date and effective against the latest phishing tactics, preventing the degradation of its predictive accuracy over time.

2.2 Incremental Learning and LLM Optimization Integration

Our approach leverages an incremental learning algorithm that continuously adapts to new phishing data without requiring full retraining. Incremental learning works by updating model parameters with each new block of data, allowing for rapid adaptation in dynamic threat environments. In our framework, the incremental classifier directly processes engineered URL features—such as URL length, domain characteristics, and heuristic measures—to yield high-accuracy predictions in near real time. This method ensures that the system remains current with evolving phishing tactics, as new data are assimilated on a continual basis.

In contrast, the LLM-based detector is optimized via a prompt optimization cycle using DSPy's MIPROv2 optimizer. Although the Detector does not implement incremental learning in the traditional sense, it is designed to incorporate contextual cues from the incremental classifier. In practice, the Detector receives an "ML Context" that summarizes the incremental classifier's prediction and confidence, along with extracted URL indicators. This blended input allows the Detector to generate explainable classifications and provide confidence scores along with detailed reasoning. While the current implementation focuses primarily on optimizing the Detector's performance, our framework is conceptually inspired by incremental learning; future work may further integrate continuous, incremental updates into the LLM's prompt, thereby fully merging the benefits of both approaches.

2.3 Integration in the Adversarial Framework

In our adversarial architecture, these components are crucial. The Generator LLM leverages the URL indicators to create emails that closely mimic actual phishing attempts, testing the limits of the detection capabilities. Concurrently, the Detector LLM applies incremental learning to assess these emails, refining its classification accuracy by learning from each interaction. Discrepancies between the generated emails and the Detector's assessments prompt optimizations that refine detection strategies, leading to a cyclic enhancement of both the generation and detection processes.

This strategic interplay between continuous learning and dynamic response forms the backbone of our adversarial system, ensuring it remains ahead of sophisticated phishing threats. The following section on **System Architecture** explores how these elements are orchestrated to create a resilient and adaptive phishing detection framework.

2.4 Method of URL Indicator Extraction

The method utilized for extracting URL indicators involves the use of a virtual machine sandbox, as depicted in the following figure. This method, used by the researchers of the referenced PDF and proposed for this paper, ensures the safe opening of suspect links and the secure extraction of indicators.

3. SYSTEM ARCHITECTURE

Our system employs an **Agentic Architecture** that seamlessly integrates cloud-based and local resources to generate and analyze synthetic email templates. At its core, the system is driven by two large language models (LLMs) operating in an adversarial loop. The **Generator LLM** (powered by Mistral) is responsible for crafting realistic phishing (or legitimate) emails, leveraging a comprehensive set of URL indicators extracted from input data. These indicators encompass a wide range of features, such as URL length, domain characteristics, TLD properties, character composition, similarity indices, and other heuristic measures designed to capture subtle anomalies.

Concurrently, the **Detector LLM** (using phi4) scrutinizes the generated emails. It evaluates the content—namely, the subject and body—along with a provided glossary of URL indicator definitions. This glossary explains key features (e.g., *URLSimilarityIndex*, *NoOfSubDomain*, *IsHTTPS*, etc.), ensuring that the detector bases its classification solely on the visible cues in the email text, without access to the generator's internal analysis. The detector outputs a classification along with detailed reasoning for its decision.

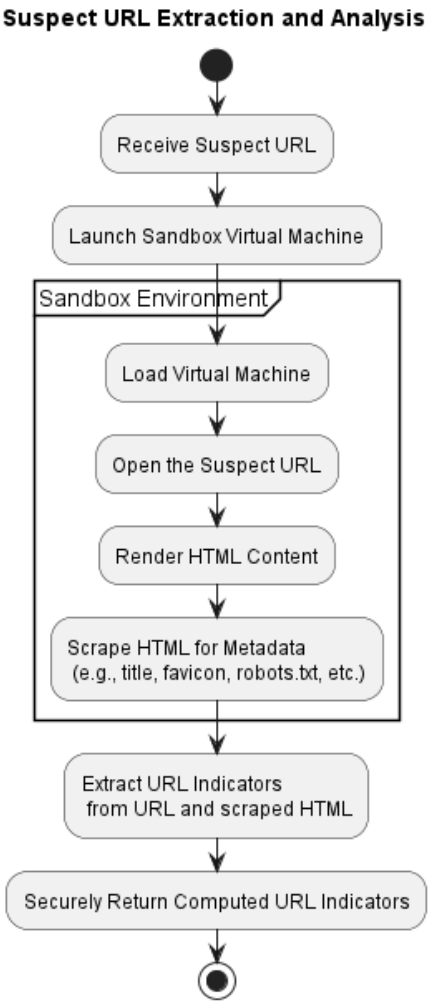

Figure 1. Flowchart illustrating the process of opening suspect links in a virtual machine sandbox and the safe extraction of URL indicators.

The adversarial interaction is central to our design. In our system, if the Detector's output (classification and reasoning) fails to align with the ground truth established by a classical incremental classifier (which directly analyzes the URL and indicator data), or if the Detector's explanation lacks sufficient detail (i.e., falls below a preset threshold of reasoning items), these discrepancies trigger a DSPy prompt optimization cycle. During this cycle, the Detector's prompt is refined via DSPy's MIPROv2 optimizer to enhance its ability to detect evolving phishing tactics and to provide more robust, confidence-weighted explanations. Although our design envisions that improved Detector performance will eventually inform and influence the Generator to adapt its phishing email crafting strategies (for example, by modifying URL indicators such as lowering the URLSimilarityIndex or enforcing an IsHTTPS flag), the current implementation is focused exclusively on optimizing the Detector. In other words, while the adversarial framework is designed to support an escalating arms race between phishing simulation and detection, our present work emphasizes the dynamic refinement of the Detector's prompt as a critical step toward achieving this long-term goal.

The detection LLM prompt optimization metric is shown in Algorithm 1, the phishing metric is computed by checking the label equality and adding a bonus based on the length of the reasoning.

Algorithm 1 Phishing Metric

function PHISHINGMETRIC(predicted, actual)
  if $predicted.label = actual.label$ **then**
    $base \leftarrow 1$
  else
    $base \leftarrow 0$
  end if
  $reasoning \leftarrow predicted.\_store.\text{get}(\text{"reasoning"}, \emptyset)$
  if $|reasoning| > \tau$ **then**
    $bonus \leftarrow (|reasoning| - \tau) \times 0.1$
  else
    $bonus \leftarrow 0$
  end if
  return $base + bonus$
end function

The system also logs detailed performance metrics at every step. These logs capture block-level incremental learning performance, test set evaluations (accuracy and reasoning quality) both before and after optimization cycles, and key events (such as blacklist updates and prompt optimization completions). This structured logging facilitates the extraction of data for generating performance tables and supporting the claims in our paper regarding the adaptability and robustness of the adversarial framework.

3.1 Implementation Resources: Mistral 7B, Microsoft Phi-4, and SMU SuperPOD

Our experimental pipeline leverages the advanced capabilities of large language models for both email generation and evaluation. In our framework, the **Generator LLM** utilizes the **Mistral 7B model**[6] to craft realistic phishing (or legitimate) emails based on detailed URL indicators. The Mistral 7B model is recognized for its high performance in text generation tasks and provides creative synthesis of threat scenarios that serve as input to the detection component.

For the detection side, we employ the **phi-4 model**[7] as the Detector LLM. This 14-billion parameter model, which incorporates synthetic data throughout training and is partially distilled from GPT-4, excels on reasoning-centric tasks due to its enhanced data generation and post-training techniques.

Figure 2 provides a summary flowchart of the adversarial interaction:

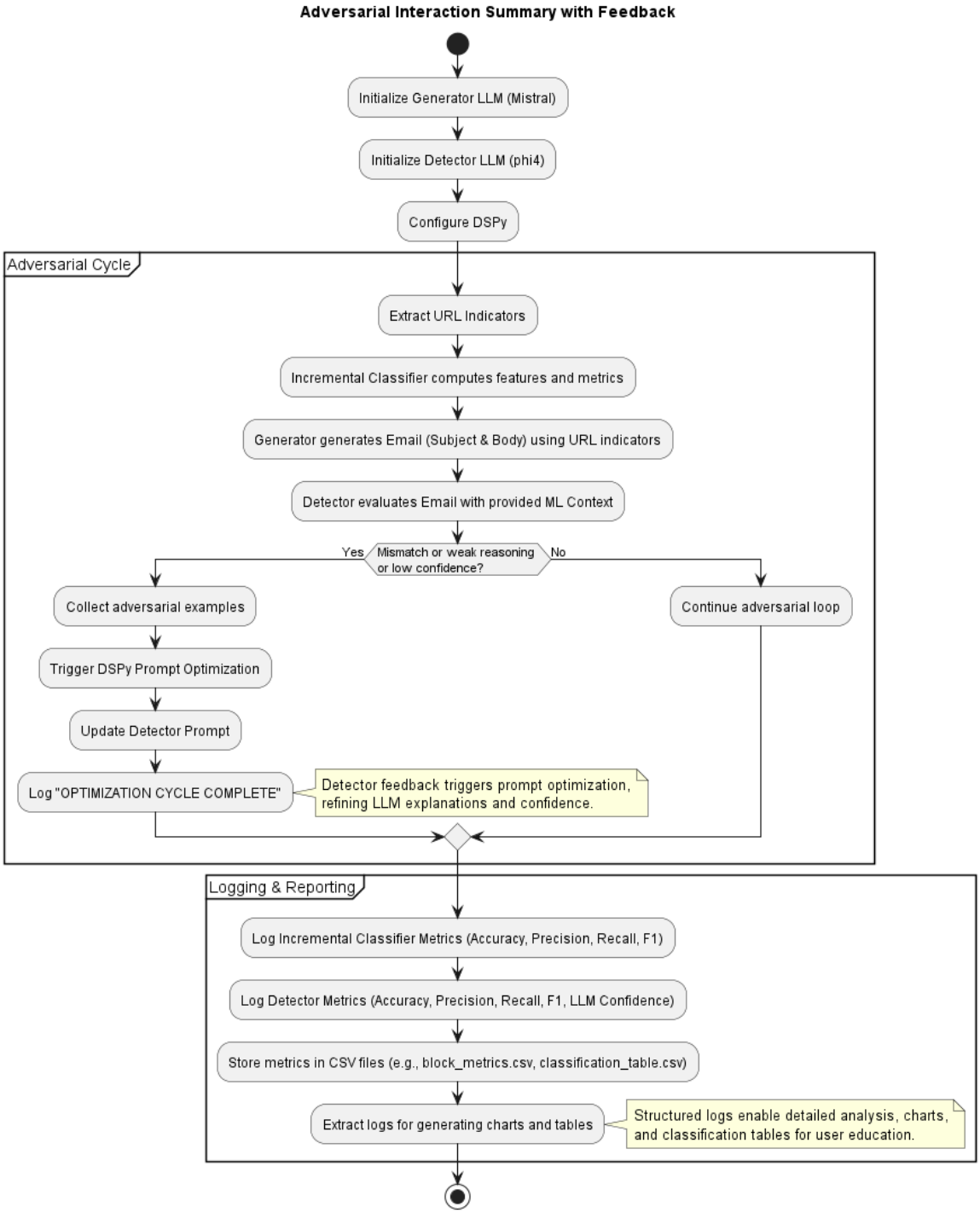

Figure 2. Adversarial Agentic Phishing System Architecture. The Generator crafts realistic phishing emails using URL indicators, while the Detector analyzes the email content and associated indicators to classify the email and provide detailed reasoning along with a confidence score. When discrepancies or insufficient explanations are detected, prompt optimization is triggered to refine the Detector's prompt. Although the adversarial framework is designed to eventually enable the Generator to adapt based on Detector feedback, the current implementation focuses on optimizing the Detector's performance.

All optimization experiments—including parameter tuning with DSPy's MIPROv2 optimizer—were conducted on the **SMU NVIDIA DGX SuperPOD Advantage**.[8] This high-performance computing cluster consists of 20 DGX A100 nodes, each equipped with 8 GPUs, enabling large-scale parallel evaluations of different prompt strategies and instruction sets. This computational environment allowed us to rapidly iterate and converge on an optimal prompt strategy for our Detector LLM, significantly reducing iteration time and facilitating robust performance evaluations.

The optimized prompt instructs the discriminator to:

> "Analyze the provided email content and classify it as either `phishing` or `legitimate`. Provide your response as a JSON object with exactly two keys: `label` and `reasoning`. The `reasoning` should be an array of strings that detail your step-by-step evaluation. Do not include any markdown or additional formatting."

4. ILLUSTRATIVE EXAMPLE

Email Phishing Detection Classification Output

Subject line: Unexpected Win Notification from Southland Casino - Verify Your Account Now!
Body:
Dear valued customer,
We are writing to inform you about an important update regarding your account at Southland Casino. To ensure the security and integrity of your personal information, we have recently made some changes to our website's address.
Please be advised that our new web address is slightly different from the one you may have previously used. It now includes a few more letters and fewer numbers than before. This change is part of our ongoing efforts to enhance the safety and efficiency of our online services.
We strongly recommend that you visit our updated website at southlandcasino.com as soon as possible to ensure uninterrupted access to your account and all its features. If you encounter any difficulties or have questions, please do not hesitate to contact our customer support team for assistance.
Thank you for choosing Southland Casino. We appreciate your patience and understanding as we continue to improve our services for your convenience.
Best regards,
The Southland Casino Team
Classification: phishing
Reason: The email claims an urgent need to verify the account due to a change in the website address, which is a common phishing tactic to create urgency and prompt immediate action. The mention of changes to the website's address without prior notification or context raises suspicion. Legitimate companies typically inform customers through official channels about such significant updates. The email lacks specific details about the user's account, such as their name or any personalized information, which is unusual for legitimate communications from a service provider. There is no direct link provided in the body of the email to verify its authenticity. Instead, it instructs users to visit a website by typing the address manually, which can be misleading if the user does not recognize the domain as legitimate. The language used in the email is somewhat generic and lacks the professional tone expected from a reputable company like Southland Casino. Phishing emails often create a sense of urgency or fear about account security to prompt users into taking immediate action without verifying the legitimacy of the request.

While traditional models would require retraining to capture these nuances, prompt-engineered LLMs can immediately integrate such patterns into their reasoning by updating the prompt with contextual information.

5. EXPERIMENTS AND RESULTS

Our experiments show that:

- The Generator LLM produces realistic phishing emails incorporating deceptive tactics.

Table 2. Detector Performance Metrics Over Time

Elapsed Time (min)	Block	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
0.00	1	60.00	64.77	86.36	74.03
27.92	2	65.00	70.00	88.73	78.26
Additional experiments in progress					

- The Discriminator LLM, following iterative prompt refinements and post-processing, reliably outputs clean JSON and correctly classifies phishing emails.
- The adversarial feedback loop enhances system robustness and adaptability.

6. FUTURE WORK

While the current framework demonstrates promising results in dynamically detecting and explaining phishing attempts, an important avenue for future research is the development of personalized cognitive models to monitor and enhance individual user learning. Specifically, we propose to investigate methods for modeling the cognitive processes underlying user decision-making in response to phishing alerts. By capturing metrics such as response accuracy, reaction time, and error patterns, our goal is to build a user-centric model that tracks progress toward improved decision-making.

In parallel, a key concept for future work is *Outcome Engineering*—the idea that AI-enabled humans can achieve desired personal or organizational outcomes by leveraging adaptive, interpretable technology.[3] In our context, the desired outcome is a better-educated user, one who can reliably detect and respond to phishing attempts and other deceptive schemes. By integrating data collected from user interactions with psychometric and behavioral indicators, the system can construct individualized learning profiles. These profiles would enable the system to provide tailored feedback and adaptive training interventions that not only enhance threat detection accuracy but also foster long-term improvements in decision-making.

Leveraging theories from cognitive psychology—such as metacognition, cognitive load, and transfer of learning—alongside advanced machine learning techniques, the proposed research would contribute to a deeper understanding of how users assimilate and apply threat intelligence. This line of inquiry promises to improve cybersecurity outcomes by not only enhancing the technical detection of phishing attacks but also by empowering users to achieve a heightened state of vigilance. Such an approach is expected to have a transformative impact on human-in-the-loop security, ultimately leading to more resilient organizations and better-informed individuals.

Future studies will explore the feasibility of real-time cognitive modeling, the design of effective adaptive feedback mechanisms, and the long-term impact of these interventions on user behavior and overall system resilience. In doing so, this research will advance the broader goal of Outcome Engineering—demonstrating that AI-enabled systems can drive significant improvements in both personal and organizational outcomes through enhanced, adaptive learning.

7. CONCLUSION

This paper presents a novel adversarial framework that leverages programmatic prompt optimization to dynamically detect and explain suspicious emails. Our approach integrates two distinct yet complementary components: an incremental URL indicator classifier that continuously adapts to evolving phishing tactics, and an LLM-based phishing detector that provides detailed, confidence-weighted explanations of its classifications. The incremental classifier offers rapid, high-accuracy predictions based on engineered URL features, while the LLM detector supplements these predictions with human-readable reasoning that can be used to educate end users about the subtle red flags of phishing attempts.

By combining these methodologies, our framework achieves robust detection performance and also addresses a critical need for transparency and user education in cybersecurity. The LLM's ability to articulate its decision-making process, including a confidence score and a step-by-step explanation, equips users with actionable insights—enabling them to understand why an email was flagged as suspicious and how to recognize similar threats

in the future. This interpretability is essential for fostering trust in automated security systems, particularly in environments where user awareness is a key line of defense.

Moreover, the adversarial feedback loop—where discrepancies between the incremental classifier's output and the LLM detector's explanation trigger prompt optimization cycles—ensures continuous refinement of the detection strategy. This dynamic adjustment, implemented via DSPy's MIPROv2 optimizer, creates an adaptive system capable of responding to new phishing strategies in near real-time. The detailed, structured logging of block-level metrics and per-sample classification outcomes further supports rigorous analysis; it enables the generation of performance charts and classification tables that can be used to evaluate and improve the system over time.

In summary, our work demonstrates that merging rapid, adaptive machine learning with the rich interpretability of LLM-based reasoning can significantly enhance phishing detection systems. This dual approach not only improves detection accuracy but also provides a valuable educational component, helping users develop a better understanding of phishing indicators. Future work will focus on deepening the integration between these components and further refining the prompt optimization process, paving the way for even more resilient and user-empowering cybersecurity solutions.

REFERENCES

[1] APWG, "Phishing activity trends report for q3 2024," *APWG Reports* (2024). Accessed: [Insert Date].

[2] Review, H. B., "The threat of ai-driven phishing," *Harvard Business Review* (2024). Accessed: [Insert Date].

[3] D. L. Young, E. C. L. and Thornton, M. A., "Ai-assisted outcome engineering for mapping natural language to radar configuration files," in [*Proceedings of the IEEE Dallas Circuits and Systems Conference (DCAS)*], (2025). Under review.

[4] Khattab, O., Singhvi, A., Maheshwari, P., Zhang, Z., Santhanam, K., Vardhamanan, S., Haq, S., Sharma, A., Joshi, T. T., Moazam, H., Miller, H., Zaharia, M., and Potts, C., "Dspy: Compiling declarative language model calls into self-improving pipelines," *The Twelfth International Conference on Learning Representations* (2024).

[5] Prasad, A. and Chandra, S., "Phiusiil: A diverse security profile empowered phishing url detection framework based on similarity index and incremental learning," *Computers & Security* **136**, 103545 (2024).

[6] Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. D. L., Sayed, W. E., et al., "Mistral 7b," (2023). arXiv preprint.

[7] Research, M., "Phi-4: A 14-Billion Parameter Language Model." https://www.microsoft.com/en-us/research/uploads/prod/2024/12/P4TechReport.pdf (2024). Accessed: 2025-01-23.

[8] Southern Methodist University, OIT Services.