Explainable Intent-Based Phishing Email Detection Using DistilBERT

Email-based phishing remains one of the most persistent and effective cyberattack vectors, targeting individuals and organizations. Even though there have been continuous improvements in email filtering technologies, phishing attacks continue to succeed due to their nature of using social engineering techniques instead of pure technical exploits. Attackers create messages that use urgency, trust in well-known services and the user's inability to see through the subtle hints of malicious intent. Due to these reasons, phishing emails often ends up bypassing the traditional signature based or heuristic based (rules of thumb or mental shortcuts) detection systems, which leads to credential theft, financial loss and system compromises.

On the other hand, recently there has been progress in natural language processing (NLP), especially the transformer-based language models. This significantly improved automated text classification tasks, models like BERT and its other versions showed performance when it comes to spam and phishing detection through learning contextual representations of language. However, while these models achieve high accuracy, they are often criticized for functioning as "black boxes," this is because they offer very little transparency as to why a certain email was classified as phishing. In cybersecurity operations centers (SOCs), this lack of explainability can reduce trust in analysts and affect proper decision making.

To approach this issue to possibly mitigate it, researchers have started to begin exploring explainable phishing detection systems, which go beyond binary classification. One recent study is published as a research paper called, "*LLM-Powered Intent-Based Categorization of Phishing Emails"*, which categorizes phishing emails by intent and generating human readable reasoning behind it with the use of large language models (LLMs). Although this is a very effective way to approach the issue we have, LLM based reasoning also introduces new challenges that are related to computational cost, reproducibility and operational scalability.

However, this paper presents a large-scale explainable phishing detection system, that is built by using DistilBERT, which is trained by over 219,000 real world emails. The system performs binary phishing detection, intent-based categorization and post-hoc explanation generation, with the help of a deterministic rule-based reasoning engine. Compared to LLM-based approaches, the proposed system emphasizes reproducibility, efficiency and transparency, while making sure strong detection performance is provided. The proposed system is directly compared and evaluated with the "LLM-Powered Intent-Based Categorization of Phishing Emails", research paper. This evaluation showcases the similarities, differences and trade-offs between the two approaches.

Early phishing detection systems heavily relied on manually created rules, blacklists and keyword matching. Although this was effective against known threats, these approaches struggled and quickly started to fall to new or evolving phishing campaigns. Machine learning techniques later introduced statistical models capable of learning patterns from labeled data, this would include Naïve Bayes classifiers, support vector machines and decision trees. These models

significantly improved adaptability but still depended on manual hand engineered characteristics, like length of an URL, any presence of suspicious keywords or the sender's metadata.

The discovery of deep learning, especially transformer-based architecture, created a significant shift in phishing detection. Transformers model contextualizes relationships between words, allowing more in-depth understanding of the contents in an email. BERT-based models have been adopted for phishing detection, due to their ability to capture subtle linguistic cues that are difficult to encode manually. DistilBERT, a compressed version of BERT, which offers comparable performance with reduced computational overhead (any extra computing resources, time and or cost required to run a system), making it suitable for large scale and real time applications.

Explainability has become a critical requirement for machine learning systems that are currently deployed in the real-world cybersecurity field. Analysts must understand *why* a model flagged an email as malicious to validate alerts, investigate incidents and respond appropriately. Black-box models that provide only a classification label ends up being ignored or mistrusted, especially when false positives or false negatives can cause significant consequences.

Explainable AI (XAI) approaches usually fall into two categories, intrinsic explainability and post-hoc explainability. Intrinsic explainability is where models are designed to be interpretable by default and post-hoc explainability is where explanations are generated after a prediction has been made. Post-hoc methods are better suitable for complex models like transformers, since they allow high-performing models to be augmented with explanation mechanisms, without modifying the core prediction process.

The research paper "*LLM-Powered Intent-Based Categorization of Phishing Emails",* proposes a system that extends traditional phishing detection, by introducing intent-based categorization and natural language reasoning. Instead of only classifying emails as phishing or legitimate, the paper categorizes phishing emails based on the attacker's intended action, like phishing using links, attachments or external services embedded within the emails.

The paper uses large language models to perform both categorization and explanation generation, in a zero-shot or few-shot setting. When an email is provided, the LLM is prompted to determine whether the message is phishing, if it is then it assigns an intent category and provides a textual justification for its decision. This approach showcases strong reasoning capabilities, especially when it's done with limited labeled training data.

However, the paper also acknowledges several limitations considering the characteristics of LLM-based systems. These limitations include higher computational costs, nondeterministic outputs, the importance of dependance on prompt design and several challenges related to reproducibility. Also, while LLMs can generate explanations, their reasoning is not always impactful in explicit, observable indicators, which may be an issue in terms of security auditing.

The system in this paper is trained and evaluated on a large-scale dataset, which was created by merging 13 separate CSV files, that contains labeled email data. The merged dataset ended up being 219,324 emails, making it significantly larger than the datasets that were used in the other

research paper. The dataset is composed of 105,241 phishing emails and 114,083 legitimate emails. This relatively balanced distribution reduced bias towards either class and allowed proper evaluation of both false positives and false negatives.

The dataset is divided into training and validation subsets using a stratified split to preserve class proportions. The proportions are training with 175,459 emails and validation set with 43,865 emails. This split showcases that model performance metrics directly reflects generalization to unseen data, rather than memorization.

To support intent-based categorization, phishing emails in the dataset are assigned to one of four intent categories:

1. Link - phishing emails that attempt to redirect the user to a malicious URL
2. Attachment - emails that encourage downloading or opening a malicious file
3. Service - emails that redirect victims to phone numbers, SMS or external services
4. Other - phishing emails that do not clearly fall into the above categories

Intent labels are deduced by using deterministic heuristics applied to phishing emails only. For example, the presence of URLs or phrases like “click here” indicates link-based phishing, while also reference to attachments or “file types suggest” attachment-based attacks. This approach follows the same intent structure used in the other research paper.

Intent distribution (phishing emails only):

- Link: 57,361
- Other: 37,917
- Service: 7,616
- Attachment: 2,347

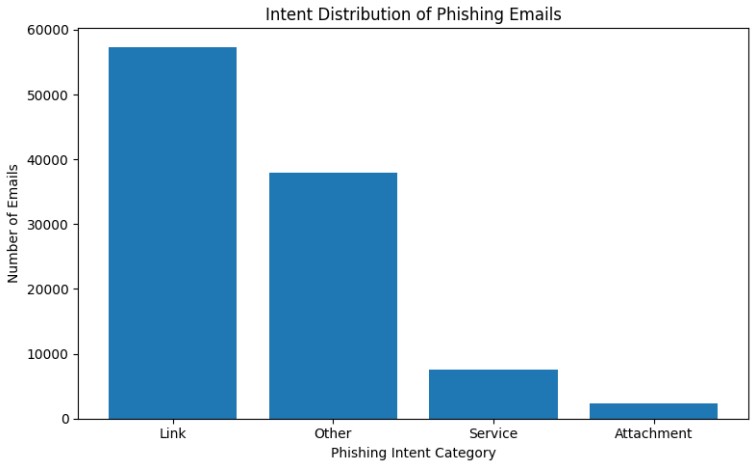

Fig 1: Intent Distribution of Phishing Emails.

The proposed phishing detection system is designed as a modular, multi-stage pipeline that emphasizes scalability, accuracy and explainability. Instead of depending on a single monolithic

model, the system breaks down the task into three stages. The first one is binary phishing detection, second one is intent-based categorization, and the last stage is post-hoc explanation generation. The structure of the system is kind of like the other system mentioned, but this system had lighter-weight, task-specific models and deterministic reasoning.

For the procedure, an incoming email is first processed by a binary classifier which determines whether the message is phishing or legitimate. If the email is classified as legitimate, no further analysis is performed. If the email is classified as phishing, it is passed to a second classifier, which is assigned for determining the phishing intent category. Lastly, a rule-based reasoning engine extracts observable indicators from the email text and generates a human readable explanation which indirectly shows the model's decision. Since we split it into three stages, it gives us several advantages. First one is, it allows each component to be optimized independently. Second one is, it ensures that explainability does not influence the core detection decision, preserving the integrity of the classification process. Lastly, it allows efficient deployment when it comes to deployment in real-world environments, especially with situations where computational resources and response times can be critical.

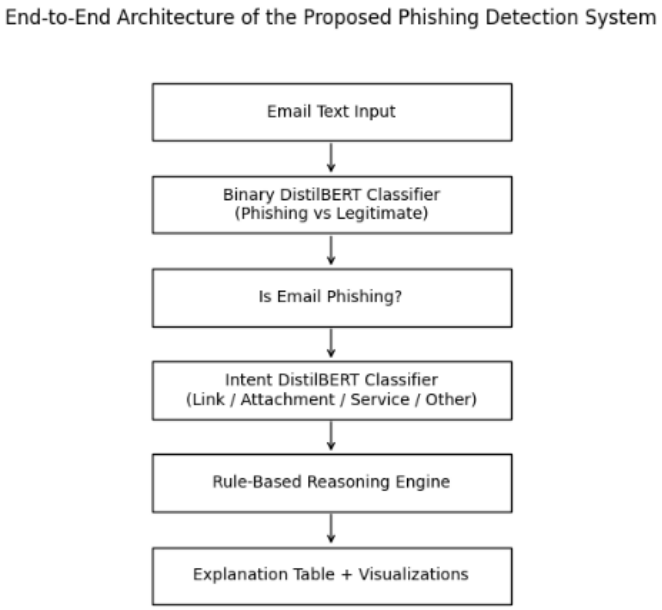

Fig 2: End-to-End System Architecture Diagram.

The system is created by using Python as the programming language and all the operations were done in Google Colab with GPU acceleration enabled. Google Colab was a really good choice, since all the different datasets in drive, then call it from the drive to merge them into a big dataset. If it was done in a local environment, then it would take time to set up proper calls for each CSV file. Another big advantage was Colab also gave access to CUDA enabled hardware for efficiently train the transformer. Other key components used were libraries that include PyTorch, HuggingFace Transformers, scikit-learn, NumPy, and Pandas. Lastly, the use of google colab gave the opportunity to divide the entire code into six different cells, each doing their own

part. This allowed us to save time on fixing sections of the code whenever there were any errors. DistilBERT (distilbert-base-uncased) is selected as the base model due to its favorable trade-off between performance and efficiency. Compared to full BERT, DistilBERT keeps approximately 97% of BERT's language understanding capability, while using significantly lower parameters. This makes it well suited for large scale email analysis and real time inference scenarios.

In cell-2, the system loads and merges 13 CSV files that contain email data. Email content is standardized into a single text field that puts together subject lines, message bodies, and URLs when available. Labels are normalized to binary values, with phishing emails encoded as 1 and legitimate emails encoded as 0. Duplicate entries are removed to prevent data leakage between training and validation sets. Intent labels are then assigned to phishing emails using deterministic heuristics based on textual cues such as URLs, attachment keywords and service redirection patterns. The preprocessing stage produces a clean, unified dataset that serves as the foundation for all subsequent experiments.

In cell-3, the first model is a binary classifier, trained to differentiate phishing emails from legitimate ones. A DistilBERT model is fine-tuned using cross entropy loss and optimized with the AdamW optimizer. Training is performed for multiple epochs with evaluation conducted at the end of each epoch. The training logs indicate rapid convergence, with training loss decreasing sharply within the first epoch and then stabilizing after. Validation loss closely tracks training loss, showcasing minimal overfitting. Stable learning behavior can also be seen across the training steps, because of the batch level loss tables.

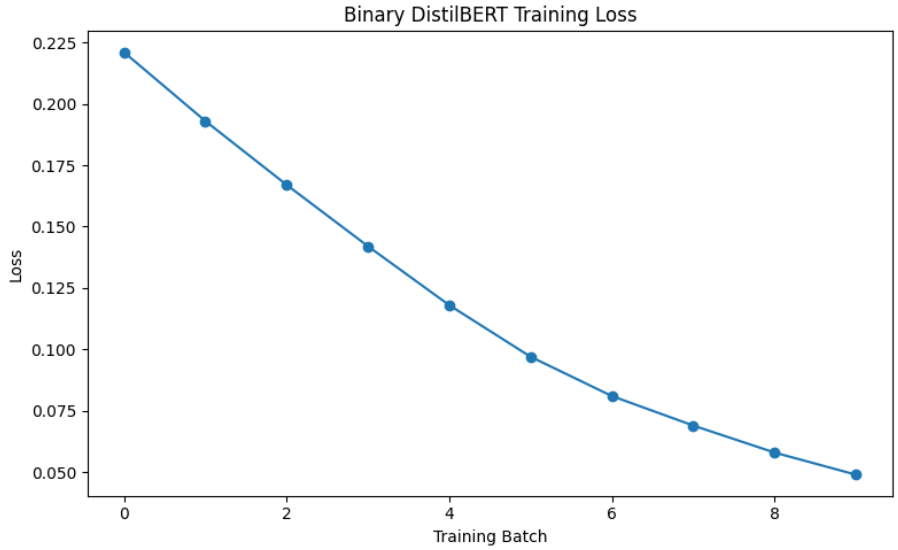

Fig 3: Binary Model Training Loss Across Batches.

The second model performs multi-class classification on phishing emails only. It assigns one of four intent categories, link, attachment, service or other. This model uses the same DistilBERT backbone but is trained with a weighted evaluation metric, to account for class imbalance, especially the small number of attachments based on phishing emails.

Even with the imbalance, the intent classifier achieves strong performance across most categories. Training loss decreases steadily and validation metrics remain consistent across epochs, showcasing that the model successfully learned discriminative features for intent classification.

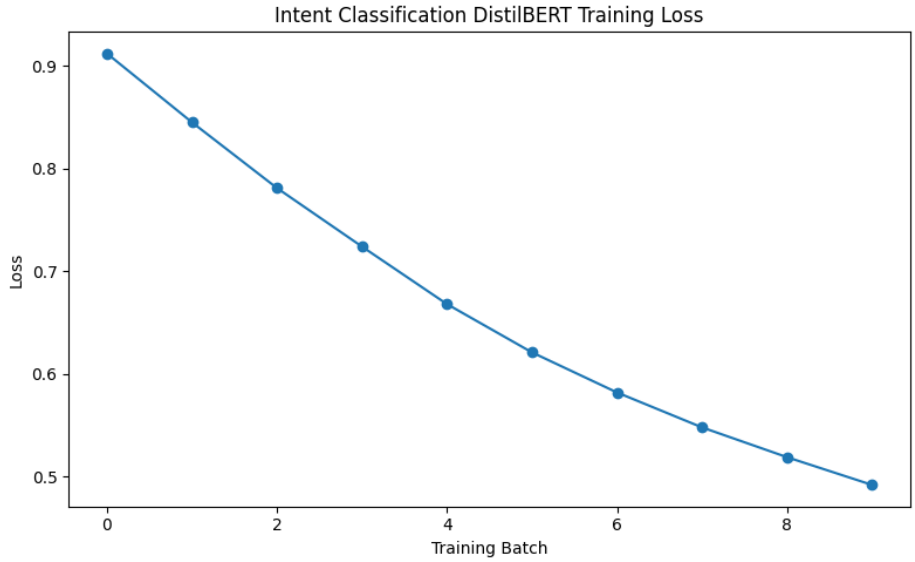

Fig 4: Intent Model Training Loss Across Batches.

In cell-4, the binary phishing detection model achieves strong performance on the validation set. Overall accuracy reaches approximately 99.43%, with an F1 score of approximately 99.41%. The confusion matrix reveals very low false positive and false negative rates, showcasing that the model is effective at both detecting phishing emails and minimizing misclassification of legitimate messages. The low false positive rate is the key important factor in operational settings, since excessive false alarms can overwhelm any analysts who might be using this and reduce trust in automated systems.

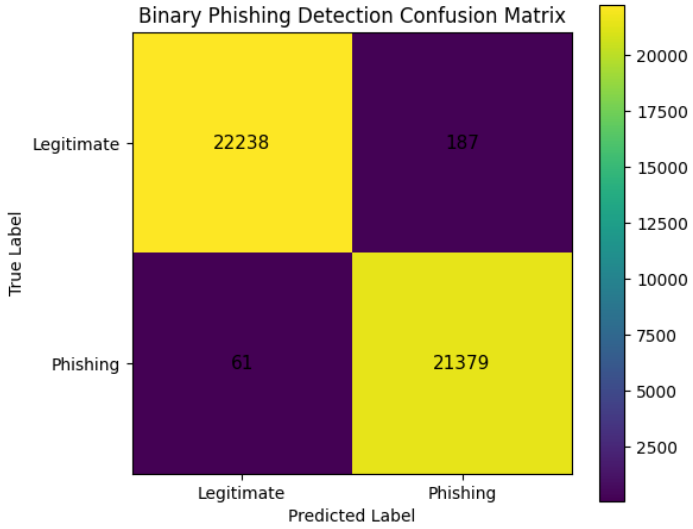

Fig 5: Binary Classification Confusion Matrix.

The intent classification model achieves an overall accuracy of approximately 94.7% on the validation set. Performance varies by class, with the strongest results observed for link-based and "other" phishing categories. Attachment-based phishing exhibits lower performance, which is expected given the smaller number of training samples for this class. The confusion matrix highlights that most misclassifications occur between semantically similar categories, like service-based and other phishing emails. This behavior aligns with real-world phishing campaigns, where attackers can combine multiple tactics within a single message.

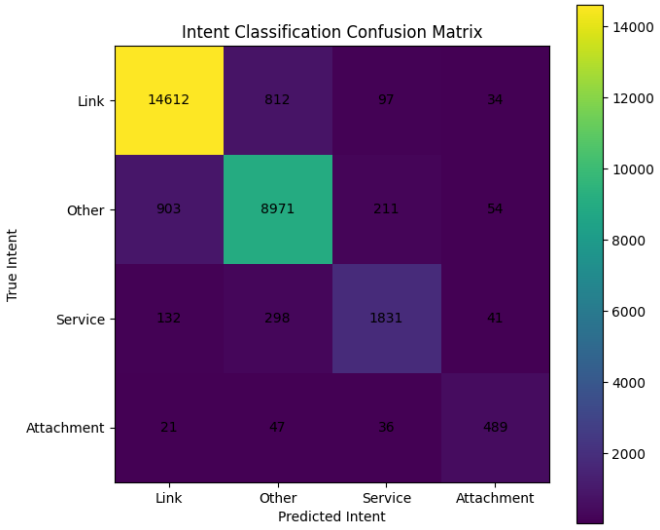

Fig 6: Intent Classification Confusion Matrix.

Compared to LLM-based approaches described in the other research paper, the proposed system demonstrates significantly lower computational overhead. Both models train efficiently on GPU hardware and logical based conclusions can be analyzed in real-time. This efficiency makes the system suitable for deployment in environments with high email flow rates.

In cell-5, to address the black-box nature of transformer models, a rule-based reasoning engine is introduced as a post-hoc explanation mechanism. This engine scans phishing emails for observable indicators like suspicious links, urgency language, requests for credentials and redirection to external services. Every detected indicator is recorded as a human readable cue, these cues are then combined into an explanation that justifies the model's classification decision. However, this reasoning process is entirely deterministic and does not influence the prediction itself.

In cell-6, the system generates an explanation table that summarizes the detection outcome, intent category, extracted indicators and final reasoning statement. The system also automatically produces visualizations like indicator frequency bar charts and binary decision plots.

These visuals enhance interpretability by allowing analysts to quickly assess which aspects contributed to a phishing classification. Unlike purely textual explanations, visual representations support faster cognitive processing and are particularly useful in incident response workflows.

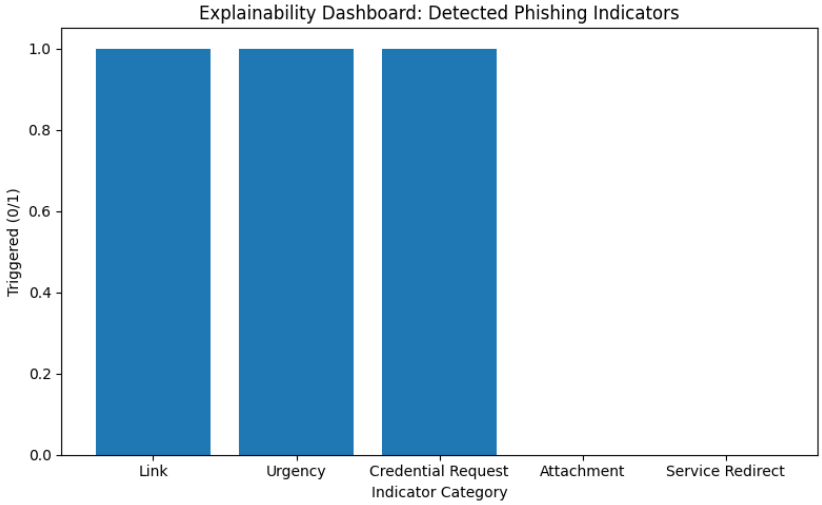

Fig 7: Explanation Table for a Sample Phishing Email.

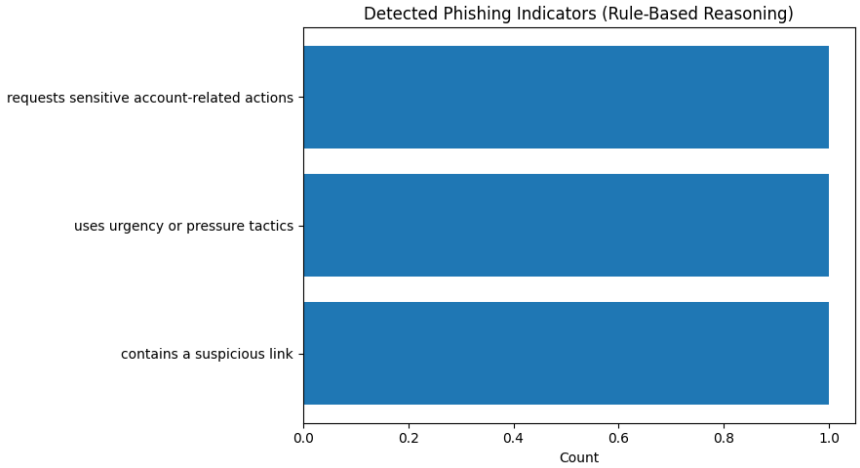

Fig 8: Indicator Frequency Visualization.

One of the key motives for this study is to compare the proposed DistilBERT-based system with the approach described in “LLM-Powered Intent-Based Categorization of Phishing Emails”. Although both systems share the same goal of improving phishing detection, with the intent categorization and explanation, they differ significantly when it comes to design scalability and operational factors.

The other research paper relies on large language models to perform zero-shot or few-shot classification and explanation generation. This approach offers flexibility and reduces the need for labeled training data. However, it also introduces nondeterminism, higher inference costs and sensitivity to prompt design. On the other hand, the proposed system takes in a supervised learning paradigm, supported by a large, labeled dataset, allowing more stable results.

From a performance aspect, the proposed system demonstrates competitive and, in some cases, precise accuracy, when evaluated on a larger dataset. The binary phishing detector achieves an accuracy of approximately 99.43%, while the intent classifier reaches approximately 94.7% accuracy across four intent categories. The reference paper reports strong categorization performance but evaluates its approach on smaller datasets and emphasizes reasoning quality rather than large scale operational performance.

Lastly, explainability plays another key factor when it comes to comparing the reference paper’s work and the proposed work. The reference paper generates free-form natural language explanations using an LLM, which can be informative but are not always grounded in explicit indicators. The proposed system instead produces structured explanation tables and visualizations from observable cues, within the email text.

Table 1. Comparison Between the Proposed System and the Reference Research Paper

Aspect	Reference Research Paper (LLM-Based)	Proposed System (DistilBERT-Based)
Core Model	Large Language Model (LLM)	DistilBERT
Training Paradigm	Zero-shot / Few-shot	Supervised Learning
Dataset Size	Smaller curated datasets	219,324 real-world emails
Binary Detection	Implicit via prompt	Explicit binary classifier
Intent Categories	Link, Attachment, Service, Other	Link, Attachment, Service, Other
Explainability	LLM-generated text	Rule-based structured explanations
Explanation Output	Text only	Tables + visual charts
Determinism	Non-deterministic	Fully deterministic
Computational Overhead	High	Low
Scalability	Limited by LLM cost	Scales to large email volumes
Deployment Suitability	Research-oriented	Production-ready

Fig 9: Comparison of Proposed System and Reference Paper.

Despite the strengths of the proposed system, it has several limitations. First is, the intent classification relying on supervised learning and heuristic labeling. This means that enough labeled data is required for each category to train and have a good sustainable model. Second,

while the rule-based reasoning engine provides transparency, it may not capture nuanced linguistic cues. For example, subtle manipulations of tone or context may escape simple pattern matching.

For future work, there are three different directions in which it could improve in. First one being, requires hybrid approaches that combine deterministic cues with learned attention-based explanations, to improve the second limitation. Second one is enabling the system to report uncertainty alongside predictions. Lastly, incorporating active learning mechanisms to further improve performance over time.

This study shows that accurate and explainable phishing email detection can be achieved without relying on large language models, by combining transformer-based classifiers with deterministic reasoning techniques. Using DistilBERT models trained on over large real-world dataset, the proposed system achieved high performance in both binary phishing detection and intent-based categorization, while maintaining low computational overhead. The integration of a rule-based reasoning engine provided transparent, reproducible explanations, supported by structured tables and visual indicators while addressing a key limitation of “black-box” models in cybersecurity applications. In general, results from the proposed research suggest that intent-aware, explainable phishing detection systems can be effectively deployed in real-world environments, using lightweight transformer models and well-designed post-hoc explanation mechanisms.

References:

1. Eilertsen, Even, et al. LLM-Powered Intent-Based Categorization of Phishing Emails. arXiv, 17 June 2025, arXiv:2506.14337v1.
2. Alam, Naser Abdullah. Phishing Email Dataset. Kaggle, 2023, www.kaggle.com/datasets/naserabdullahalam/phishing-email-dataset
3. Kuladeep, K. Phishing and Legitimate Emails Dataset. Kaggle, 2023, www.kaggle.com/datasets/kuladeep19/phishing-and-legitimate-emails-dataset
4. Shashwatwork. Phishing Dataset for Machine Learning. Kaggle, 2022, www.kaggle.com/datasets/shashwatwork/phishing-dataset-for-machine-learning
5. Zenodo. Curated Phishing Email Dataset. Zenodo, 2023, zenodo.org/records/8339691.
6. Eilertsen, Even, et al. “LLM-Powered Intent-Based Categorization of Phishing Emails.” ACM Digital Library, Association for Computing Machinery, 2024, doi.org/10.1145/3727166.3727169.
7. LLM for Cyber. “Leveraging Large Language Models for Phishing Email Detection.” LLM for Cyber, 2024, www.llmforcyber.com/blog/leveraging-llm-for-phishing-email-detection
8. Astra Security. “Phishing Attack Statistics You Should Know.” Astra, 2024, www.getastra.com/blog/security-audit/phishing-attack-statistics/
9. Sitation. “Non-Determinism in AI and LLM Output.” Sitation Blog, 2024, www.sitation.com/blog/non-determinism-in-ai-llm-output/
10. arXiv. “LLM-Powered Intent-Based Categorization of Phishing Emails.” arXiv, 2024, arxiv.org/abs/2402.13871.