© 2026 Arabian International Conference on Science,Technology and Innovation. All rights reserved.

EVALUATING MODEL TRANSPARENCY IN PHISHING DETECTION USING LIME EXPLAINABILITY FRAMEWORKS

Mirtakhirova Nazokat
ML engineer at Artificial Intelligence Research Institute

ABSTRACT

Blackbox machine learning models are widely deployed in cybersecurity for mostly automating threats but due to their lack of interpretability it has become tough for security analysts to determine why specific web link or email content is flagged as a phishing. In modern enterprise environments, security operations centers (SOCs) and incident response teams usually handle the massive volumes of incoming digital traffic daily but they mostly rely on complex classification algorithms and pipelines to filter out malicious URLS and misleading payloads. For example, when transformer models flag specific email as malicious with high confidence , LIME reveals this decision was made due to entirely one typo. This research investigates the practical implementation of the LIME (Local Interpretable Model-agnostic Explanations) framework to suggest visualizing complex decision making ways of modern phishing detection. Along with that LIME highlights the exact sub-components – such as unusual strings, suspicious domains or parameters that trigger specific model results. After that security analysts can observe specific predictions closely, helping them to differentiate between legitimate and malicious content.

Furthermore, this study explores the stability and effectiveness of explanation models when they are applied to hybrid architectures that combines transformer based models with traditional metadata features. Our evaluation included some critical insights into model behavior, identifying the cases where model's high confidence in some predictions rely on unstable features making the model vulnerable to adversarial threats. When local explanations are visualized security teams can easily identify system errors, noisy outputs eventually enhance the training dataset. It shows that, explainable AI frameworks do not just enhance operational trust but it also provides structural framework for diagnosing performance. As the result it can improve the overall system resilience to real world malicious phishing threats.

Keywords: phishing, Explainable AI, LIME, model transparency, cybersecurity, AI frameworks, URLs, Security Operation Centers

Introduction

Cyber security threats continue to rise in scale and sophistication resulting serious harm in individuals, organizations and critical infrastructures. The threat on cloud computing, devices and digital communication platforms has created new attack surface, making traditional approaches insufficient to handle. Among these threats phishing attack remains one of the most common, damaging sensitive information and gaining unauthorized access to various systems.

Recent advances in natural language processing has significantly enhanced the ability of machine learning techniques to understand and classify textual data. The Transformer-based architectures particularly BERT have been widely popular because of its ability to capture contextual among each word in particular sentences. Despite these factors transformer models like BERT is relatively heavyweight and costly to implement

Despite the rising challenges current cybersecurity systems rely heavily on rule-based filters and conventional machine learning classifiers. While these approaches can detect known threats, they often fail on new, unsophisticated phishing threats and struggle with real-time processing and resource-constrained devices

© 2026 Arabian International Conference on Science,Technology and Innovation. All rights reserved.

Phishing detection in email environments require rapid processing of large volumes of incoming messages, being under strict latency constraints. In most of the practical applications computational resources such as memory, processing and energy is limited

Related work

Phishing detection has evolved significantly from earliest rule-based techniques and manual engineering features to advanced deep learning and machine learning pipelines. While traditional classifiers and modern architectures achieve high level accuracy , they often lack interpretability and heavy to implement in real life.

Recent literature covers issues like balance between high statistical accuracy and achieving operational deployment in environments especially where resources are limited.

Prior studies mostly focus on integrating deep learning architectures with metadata or HTML structures yet fail to explain how these complex architectures make decisions under challenging real world conditions. Furthermore, while explainable AI (XAI) frameworks like LIME have been widely used in such natural language processing tasks, but their application to hybrid , multi framework phishing detection systems remains unexplored , especially in terms of explanation stability and robustness to adversarial inputs.

Research questions

To better evaluate how transparent and reliable these phishing detection systems are this study explores following key questions.

1. To what extend these local surrogate explainability frameworks such as LIME, can accurately expose specific lexical mistakes, structural typos or domain substrings that drive high confidence in most phishing detection and classification architectures?

2. Do local explanation models reveal structural instabilities and vulnerabilities to adversarial manipulation when hybrid architectures like multi framework transformer models with metadata are deployed in production environments?

3. How does integration of visual interpretability impact the ability of Security Operation Center (SOC) analysts filter incorrect alerts and identify model errors?

Methodology & Implementation

To investigate model transparancy and decision making process, this research implements three component lightweight phishing detection framework that combines URL analysis, HTML structure and full email content powered with LIME interpratibility framework.

Component 1. DistilBERT lightweight transformer model for URL detection.

This framework uses fine-tuned DistilBERT model that is highly optimized to capture semantic patterns from URL strings. It trained on LegitPhish dataset with approximately 100000 samples after careful data cleaning and preprocessing as well as URL normalization. Furthermore it configured with maximum token length of 128, batch size of 32, learning rate of 2 x 10^-5 and weighted cross-entropy loss over 3 epochs to handle class distributions.

Component 2.Random Forest HTML Structural Classifier.

This framework employs Random Forest ensemble using 100 decision trees on 13 pre-extracted HTMLfeatures from CIC-Trap4Phish dataset. Along with that architecture is designed to capture visual deceptions , hidden links and malicious button structures that mostly text-only models frequently overlook. To ensure the model accuracy and consistancy the model was eveluated using 5-fold cross validation across the dataset.

Component 3.DistilBERT email Content Classifier.

Next component inside the architecture incorporates second fine-tuned DistilBERT lightweight model and trained on zefang-liu phishing email dataset including over 18k samples to enhance models

© 2026 Arabian International Conference on Science,Technology and Innovation. All rights reserved.

ability to capture deep linguistic patters , social engineering tone and persuasive language. Model also trained on maximum length of 256 tokens and uses regularization techniques such as early stopping and dropout to prevent overfitting especially in smaller datasets like this.

Explainability integration via LIME

The architecture integrates Local Interpretable Model-agnostic Explanation directly into DistelBERT to generate surrogate explanations. The model operates by applying controlled perturbations to input strings and observes how prediction shifts by isolating exact feature weight and tokens. The computational environment and experiments were stanardized across all pipeline dtages to ensure accurate performance tracking.

Hardware and cloud Infrastructure

It was executed within cloud based Google colaboratory Jupyter environment and utilized an NVIDIA Tesla T4 GPU hardware accelerator to significantly optimized fine-tuning interval and integrated to google drive storage to save model's important checkpoints and session artifacts.

Software libraries and tooling

Hugging Face Transformers & Datasets. Utilized for loading pre-trained DistilBERT weights, executing automated tokenization and streaming dataset corpora.

DistilBERT

DistilBERT is the pre-trained transformer model used for the core two components in the architecture of this research.DisilBBERT with 66 million parametrs enough provide required speed, memory and efficiency. The two separate DistilBERT instances were trained through the traing process which is URL strings from LegitPhish(2025) and email body contect from zefang-liu(2024).

Results and evaluation

URL Component: Achieved an accuracy of 99.87\% (0.0008), demonstrating high precision in detecting malicious link structures. **HTML Component**: Achieved an accuracy of 93.81 (0.0027) on structural feature sets.

Email Content Component: Achieved an accuracy of 96.79\% 0.0126}) and an F1-score of 0.958 in capturing complex social engineering indicators. **LIME Explanation Fidelity and Stability Findings:**When applied to high-confidence phishing predictions, LIME successfully isolated primary malicious tokens.

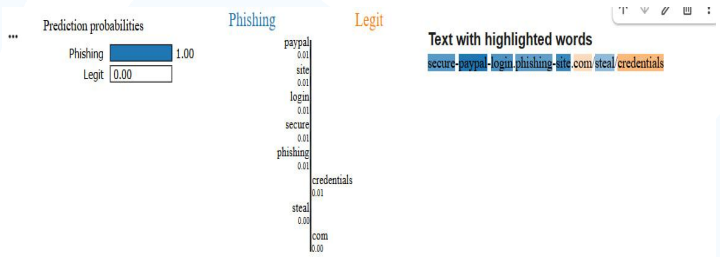


Fig 1 Lime explainability output for specific URL

References

1. Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014). Phishing detection based on associative classification data mining. Expert Systems with Applications, 41(13), 5948–5959. https://doi.org/10.1016/j.eswa.2014.03.019

2. Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J. P. (2022). An effective phishing detection model based on character-level convolutional neural network from URL. Electronics, 11(1), 60. https://doi.org/10.3390/electronics11010060

3. Anti-Phishing Working Group. (2024a). Phishing activity trends report, Q4 2024. APWG. https://apwg.org/trendsreports/

© 2026 Arabian International Conference on Science,Technology and Innovation. All rights reserved.

4. Bahnsen, A. C., Torroledo, I., Camacho, L. D., & Villegas, S. (2017). DeepPhish: Simulating malicious AI. Proceedings of the APWG Symposium on Electronic Crime Research (eCrime), 1–8. https://doi.org/10.1109/ECRIME.2017.7945461
5. Basnet, R. B., Mukkamala, S., & Sung, A. H. (2014). Detection of phishing attacks: A machine learning approach. Soft Computing Applications in Industry. https://doi.org/10.1007/978-3-540-77465-5_19
6. Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324
7. Canadian Institute for Cybersecurity. (2025). CIC-Trap4Phish dataset [Dataset]. University of New Brunswick. https://www.unb.ca/cic/datasets/
8. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of NAACL-HLT 2019, 4171–4186.