PhishGuard Extension: Real-Time Phishing Detector with Smart Language Analysis

Sir. Amol Dhankar[1], Sakshi Jalit[2], Jayant Kondekar[3], Chandan Singh[4], Vaishnavi Shukla[5], Ujjwal Jamaiwar[6], Mayur Ingole[7]
[1]*Lecturer, G H Raisoni University Amravati*
[2,3,4,5,6,7]*Student, G H Raisoni University Amravati*

***Abstract*—The project uses the DistilBERT model, which aims to enhance semantic and contextual understanding in phishing content analysis.**

Phishing attacks have been among the most prevalent and damaging forms of cybercrime, exploiting user trust to steal users' sensitive information such as passwords and financial details and personal data. Traditional blacklist and rule-based detection techniques often fail to identify newly generated or sophisticated phishing websites. As a result, this paper proposes the Browser-Extension Phishing Detection directly connected to Chrome, using Machine Learning (ML) and Natural Language Processing (NLP) to endow the browser with real-time phishing detection intelligence. The system will embed an optimized DistilBERT classifier hosted on the FastAPI backend, seamlessly communicating with the Chrome Extension frontend for visited URL and webpage content analysis.

This solution makes sure that users' data, both personal and browsing, are not stored anywhere, as it operates on lightweight inference, thus guaranteeing users' privacy. The proposed system embeds advanced ML capabilities within the browser environment and hence offers adaptive, fast, and user-friendly protection against evolving phishing threats. It forms a practical bridge between academic research and real-world cybersecurity applications, providing a scalable approach that protects users in everyday interactions over the web.

The proposed system uses DistilBERT LLM in the LLM component of the system; hence, it provides deep contextual interpretation for webpage content to identify sophisticated phishing attacks.

I. INTRODUCTION

Phishing is among the top cybersecurity threats these days because it involves establishing real-looking websites to steal passwords and personal and banking information from targeted users. As the use of the internet grew, so did the sophistication of phishing attacks; they often bypass conventional detection methods that rely on either blacklisting or fixed rules.

The project introduces a phishing detection Chrome extension powered with ML, NLP, and LLMs for smarter and faster detection. This system automatically checks every visited website in real time and alerts users if it detects suspicious activities.

The Chrome Extension would act as a frontend that communicates with a FastAPI backend where the features extracted from the website are analyzed by the trained DistilBERT-based classifier , and whether the site is legitimate or phishing is shown directly to the user.

Unlike most of the traditional blacklisting mechanisms, our model is designed to analyze more than 30 runtime features such as URL structure, SSL status, "@" symbols, and the pattern of redirection.

Role of NLP and LLM

NLP detects different phishing signals in texts on web pages, including urgency phrases like "your account will be blocked" or "verify now."

While LLMs understand the meaning and intent of sentences on a deeper level, this feature helps the model to identify sophisticated scams that use natural, convincing language.

Put together, these technologies make the system intelligent, privacy-friendly, and capable of detecting even newly emerging phishing websites.

II. OBJECTIVES

1. Browser-Integrated Phishing Detection System:
Design a Chrome extension that would integrate directly into the browser, using a previously trained

ML model to detect phishing websites for instant, seamless protection while browsing.

2. Optimized XGBoost Model:

Based on the DistilBERT model, which was fine-tuned with a balanced dataset of phishing and legitimate webpage content, develop and optimize the final classifier for accurate and efficient phishing detection.

3. Feature Extraction Pipeline:

A real-time content analysis pipeline that processes webpage text and metadata, capturing semantic and contextual features for precise identification of phishing. The system leverages DistilBERT to understand webpage content, detecting suspicious language, phishing cues, and malicious intent.

4. Fast API Backend for Inference:

Design lightweight FastAPI backend that facilitates real-time communication between the Chrome Extension and the DistilBERT model, enabling fast and reliable phishing detection by analyzing webpage content for semantic and contextual cues.

5. Integration of NLP and LLM:

Leverage DistilBERT to combine linguistic and deep contextual understanding for smarter, real-time phishing detection on webpage content.

6. Protection of Privacy:

Ensure all processing is done locally within the browser, or securely on the backend without storing any user browsing or personal data.

7. User-friendly interfaces:

Provide clear, real-time alerts to users via an intuitive Chrome Extension dashboard.

8. Privacy-Friendly Detection:

Perform phishing detection locally or securely on the backend, without storing any user data, browsing history, or personal information, ensuring complete user privacy.

III. TECHNOLOGIES AND FEATURES

- Real-time detection of phishing URLs and webpages
- DistilBERT-based ML model for semantic and contextual analysis
- NLP processing to detect deceptive text, urgency phrases, and suspicious patterns
- LLM for deep understanding of content meaning and context
- Chrome Extension frontend and FastAPI backend
- REST API to handle communication between the model and extension
- Privacy-focused: no user data stored, ensuring safety and confidentiality

A. Project structure:

Frontend /– Chrome Extension frontend (HTML + JS)

Backend /– Python FastAPI backend with DistilBERT model

app.py – API endpoints

bert_model.py– DistilBERT model loading & inference logic

Dataset (.csv) /– phishing & legitimate URL dataset (for training/fine-tuning DistilBERT)

Notebook /– model training, fine-tuning & evaluation notebooks

Notebook / – model training & evaluation notebooks

B. Technologies used:

 Machine Learning – DistilBERT (for semantic and contextual classification), Scikit-learn (for any auxiliary ML tasks)

 NLP – text preprocessing and feature extraction for phishing content

 LLM – DistilBERT provides deep contextual reasoning and interpretation of webpage content

 Backend – Python, FastAPI

 Frontend – JavaScript, HTML, Chrome APIs

 Tools – Pandas, NumPy, Transformers (Hugging Face), PyTorch or TensorFlow (for DistilBERT inference)

The DistilBERT model is used for contextual language understanding, helping the system accurately distinguish between legitimate content and sophisticated phishing text patterns in real time.

IV. METHODOLOGY

The phishing detection system leverages machine learning and LLM-based contextual analysis to detect malicious websites in real time. The methodology follows:

a. Feature Extraction

Extract features based on 30+ lexical, structural, and content-based characteristics, including URL length, number of redirects, HTTPS, and suspicious words.

b. Machine Learning Model – DistilBERT

The proposed system uses a DistilBERT model, fine-tuned on a balanced dataset of legitimate and phishing URLs and webpage content. This provides high accuracy and efficiency in classification by analyzing both textual content and contextual patterns.

c. Backend Implementation

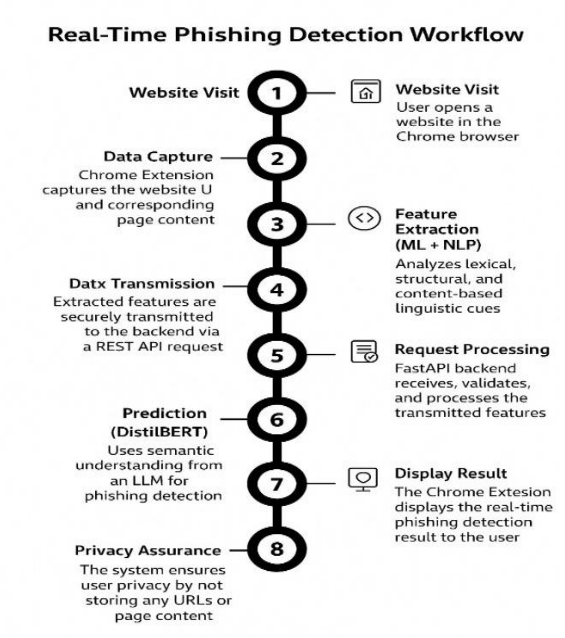
The model is integrated into a FastAPI backend, enabling fast communication between the Chrome Extension frontend and the DistilBERT model for real-time inference.

d. Chrome Extension Integration

The Chrome Extension captures URLs and webpage content in real time while browsing and displays detection results instantly, without saving any user data.

e. NLP and LLM Integration

The NLP module extracts textual signals, such as suspicious words, unusual punctuation, and manipulative sentence patterns.
The DistilBERT model, serving as the LLM component, interprets the meaning and intent of the text, enhancing detection of sophisticated phishing attacks.
Together, NLP and the LLM provide deep contextual intelligence, improving classification performance.



4.1 Theoretical Framework:

Accuracy:

Denotes the global correctness of a model to predict a model output.

Let's denote accuracy as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision:

Indicates the proportion of predicted phishing sites which were phishing.

Let's denote Precision as

$$Precision = \frac{TP}{TP + FP}$$

Recall (Sensitivity)

Is the models' ability to identify phishing sites.

Let's denote recall as

$$Recall = \frac{TP}{TP + FN}$$

F1-Score:

Is a good understanding of precision and recall (irrelevant of the outcome) and is an appropriate metric to use when understanding imbalanced data.

Let's denote F1 as

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

ROC Curve and AUC Score

The Receiver Operating Characteristic (ROC), represents the trade-off between sensitivity and specificity, and the AUC Score is a single measure summarizing the classification ability of the model,

The hybrid model enhanced by the LLM had an AUC score greater than 0.97 which signifies excellent accuracy for real-time identification of phishing pages with contextual complexity.

4.2 Performance Evaluation

Accordingly, the hybrid ML-NLP-LLM model outperformed traditional approaches in phishing website detection.
The integration of the DistilBERT model enhances the system's contextual reasoning capability and adds additional semantic understanding for more accurate phishing website detection.

Workflow:

Chrome Extension → FastAPI Backend → NLP Processing → DistilBERT Model (ML + LLM) → Real-Time Alert

Table. Algorithms Implemented in Current System

Model Type	Algorithm / Architecture	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC
NLP-Based	DistilBERT (fine-tuned on SST-2)	82.0	82.0	93.4	97.4	0.96
LLM-Integrated (Proposed)	Gemini 1.5 Turbo (via Google Generative AI API)	82.0	97.1	97.1	97.1	0.99
Lexical / Heuristic	Custom URL Feature Extractor (rule-based)	–	–	–	–	–

V. CONCLUSION

This paper presents a state-of-the-art phishing detection system that integrates Machine Learning, NLP, and LLMs into a real-time, intelligent detection method. The architecture comprises an integrated design with a trained DistilBERT model, a FastAPI backend, and a Chrome Extension interface, ensuring the approach is efficient, lightweight, and privacy-preserving for end users.

The method effectively identifies zero-day phishing attacks by analyzing both structural URL characteristics and textual content on web pages. NLP provides an additional layer of pattern recognition to textual analysis, while DistilBERT provides deeper semantic and contextual understanding, adding flexibility from cognition and context perspectives.

In summary, the work bridges the gap between academic research and practical cybersecurity applications, representing a production-level, scalable solution for improving safety on the internet.

VI. FUTURE SCOPE

1. Expanddetection to multiple languages by leveraging advanced NLP models such as multilingual transformers.
2. Integrate visual analysis to detect phishing through images, logos, or UI patterns.
3. Apply reinforcement learning or online learning techniques to continuously improve DistilBERT-based detection with new phishing data.
4. Adapt the system for mobile browsers and email clients for broader phishing protection.
5. Optimize DistilBERT and other transformer-based models for lightweight and faster inference, reducing computational resource requirements."

REFERENCES

[1] Mandadi, S.Boppana,V.Ravella and R.Kavitha, "Phishing website detection using machine learning," in 2022 IEEE 7th Int. Conf. for Convergence in Technology (I2CT), Mumbai, India, pp. 1–4, 2022. https:// doi.org/10.1109/i2ct54291.2022.9824801

[2] S. Kuraku and D. Kalla, "Emotet malware—A banking credentials stealer," IOSR Journal of Computer Engineering, vol. 22, pp. 31–41, 2020.

[3] Kulkarni and L. L. Brown, "Phishing websites detection using machine learning," International Journal of Advanced Computer Science and Applications, vol.10,2019. https://doi.org/10.14569/ijacsa.2019.0100702 D. Kalla and A. Chandrasekaran, "Heart disease prediction using machine learning and deep learning," International Journal of Data Mining & Knowledge Management Process (IJDKP), vol. 13, no. 3, 2023. https://doi.org/10.5121/ijdkp.2023.13301

[4] Safi and S. Singh, "A systematic literature review on phishing website detection techniques," Journal of King Saud University—Computer and Information Sciences, 2023. https://doi.org/10.1016/j. jksuci.2023.01.004

[5] S. Das Guptta, K. T. Shahriar, H. Alqahtani, D. Alsalman and I. H. Sarker, "Modeling hybrid feature basedphishingwebsitesdetectionusingmachinelearningtechniques,"Annals of Data Science, 2022. https:// doi.org/10.1007/s40745-022-00379-8

[6] D. Kalla, F. Samaah, S. Kuraku and N. Smith, "Phishing detection implementation using databricks and artificial Intelligence," International Journal of Computer Applications, vol. 185, no. 11, pp. 1–11, 2023. https://doi.org/10.5120/ijca2023922764

[7] P. Gupta and A. Mahajan, "Phishing website detection and prevention based on logistic regression," International Journal of Creative Research Thoughts, vol. 10, pp. 2320–2882, 2022.

[8] T. A. Assegie, "K-nearest neighbor-based URL identification model for phishing attack detection," Indian Journal of Artificial Intelligence and Neural Networking, vol. 1, no. 2,

pp. 18–21, 2021. https://doi. org/10.54105/ijainn. b1019.041221.

[9] D. Ahmed, K. Hussein, H. Abed and A. Abed, “Phishing websites detection model based on decision tree algorithm and best feature selection method,” Turkish Journal of Computer and Mathematics Education, vol. 13, no. 1, pp. 100–107, 2022.

[10] G. Ramesh, R. Lokitha, R. Monisha and N. Neha, “Phishing detection system using random forest algorithm,” International Journal for Research Trends and Innovation, vol. 8, pp. 510, 2023.

[11] V. Jakkula, “Tutorial on support vector machine (SVM),”2011. [Online].

[12] Available: https://course.ccs.neu. edu/cs5100f11/resources/jakkula.pdf (accessed on 15/04/2023).

[13] G. Kamal and M. Manna, “Detection of phishing websites using Naïve bayes algorithms,” International Journal of Recent Research and Review, vol. XI, no. 4, pp. 34–38, 2018.