Phishing Detection System using State-of-the-Art ML AND AI

Submitted by, Nikhil A Mathew

TABLE OF CONTENTS

INTRODUCTION

- This project introduces a Phishing Detection System utilizing advanced ML and AI techniques.
- It involves the implementation of tailored deep learning models for websites and URLs, integrating NLP, behavioral analysis, and anomaly detection.
- Mainly aims for increased detection accuracy, adaptability to emerging threats, and a refined user experience.

EXISTING SYSTEM

Static Rule-Based Approach

Static rules and signature-based methods for phishing detection.

Limited ML Integration

Basic ML for feature extraction but lack advanced techniques.

Less Emphasis on User Experience

Lack of real-time notifications and interactive features for users.

Absence of Client-Side Security Measures

Server-side detection without incorporating client-side security measures.

PROPOSED SYSTEM

Advanced ML and AI Techniques

Implement NLP, behavioral analysis, anomaly detection, and ensemble learning for improved accuracy

Client-Side Security Measures

Integrate client-side security measures through a browser extension, providing real-time alerts and page analysis

Static Analysis and Behavioral Detection

Conducted static analysis to identify common vulnerabilities such as buffer overflows and SQL injection flaws

TWO MODULES

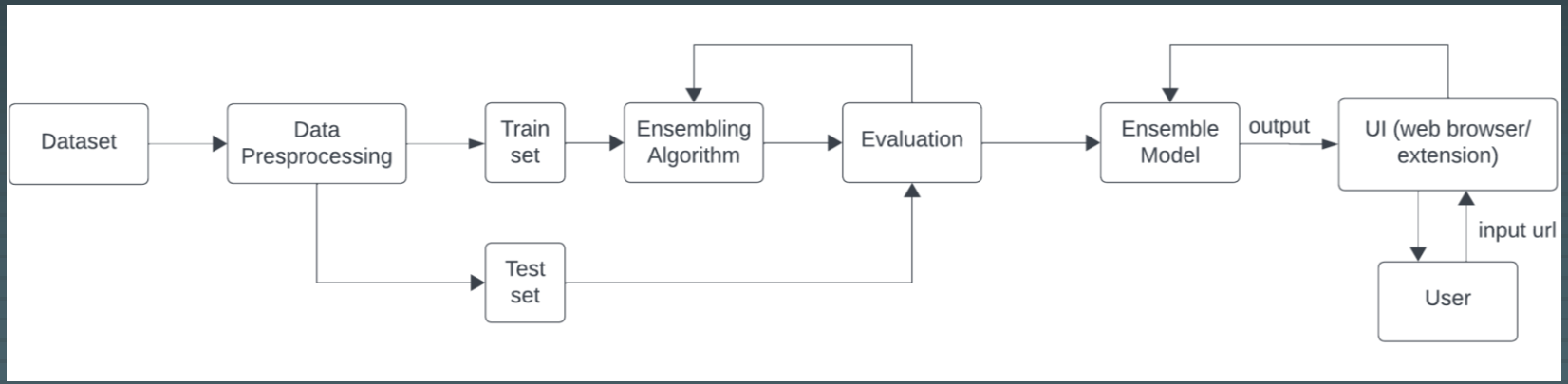
USER

- ✓ The extension fetches the current tab URL and sends it to API and predicts the result
- ✓ Can input the URL/IP to get the detailed results.
- ✓ Can report the phishing or suspicious URLs.

ADMIN

- ✓ Can manage URL/IP report details.
- ✓ Can respond to feedbacks and queries of users.

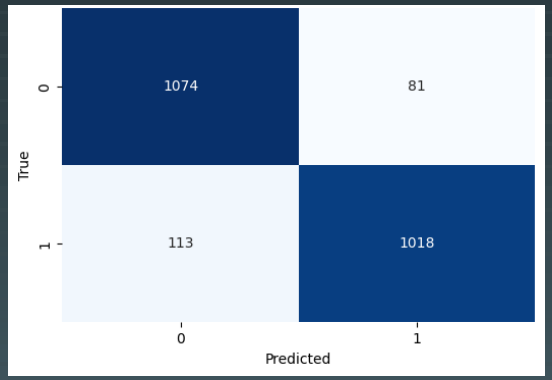
PROCESS FLOW



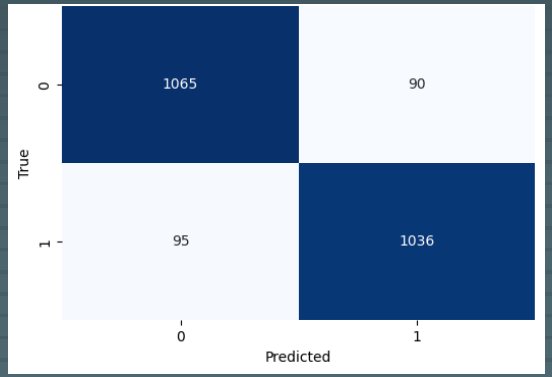
IMPLEMENTATION

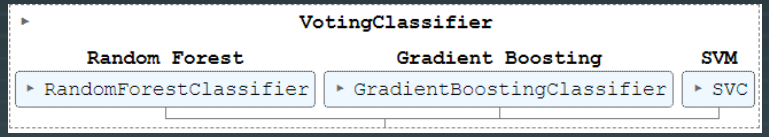
This project implements BERT (Bidirectional Encoder Representations from Transformers) framework for extracting the features from URL and predict using ensemble model to get the result.

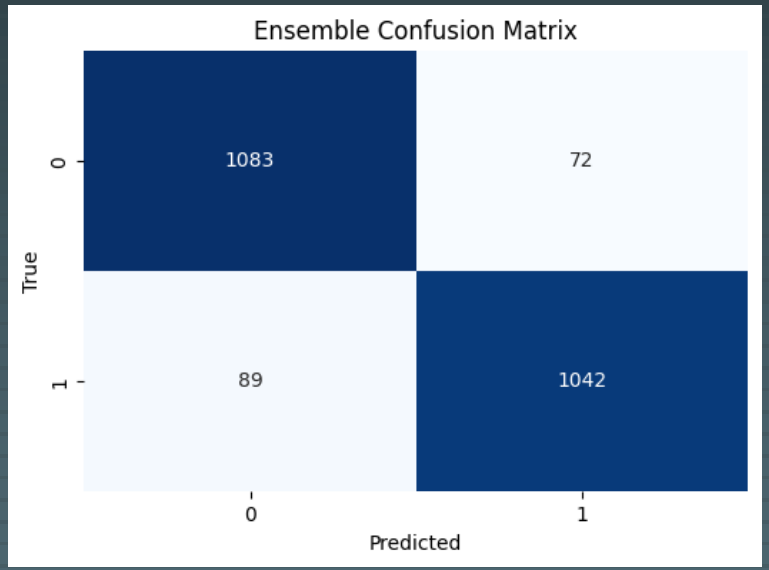
SVM



Random Forest







FEATURE EXTRACTION

```
| 💡 Click here to ask Blackbox to help you code faster
from transformers import BertModel, BertTokenizer
import torch

model = BertModel.from_pretrained('bert-base-uncased', output_hidden_states=True)
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
```

```
def extract_features(text):
    input_ids = torch.tensor([tokenizer.encode(text, max_length=512, truncation=True, add_special_tokens=True)])
    with torch.no_grad():
        outputs = model(input_ids)
        hidden_states = outputs[2]
    token_vecs = []
    for layer in range(-4, 0):
        token_vecs.append(hidden_states[layer][0])
    features = []
    for token in token_vecs:
        features.append(torch.mean(token, dim=0))
    return torch.stack(features)
```

```
URL :  https://web-facebook.com/email
[[ 0.46388137 -0.197477    0.3489256  ... -0.6999789  -0.04692942
   0.41776296]
 [ 0.535829   -0.2559319   0.38091075 ... -0.8139619  -0.2573377
   0.42401108]
 [ 0.5034204  -0.22673263  0.22911905 ... -0.80252373 -0.4607492
   0.25550106]
 [ 0.53867    -0.17979275  0.033126   ... -0.1576785  -0.48102996
   0.02750963]]
Result :  PHISHING
```

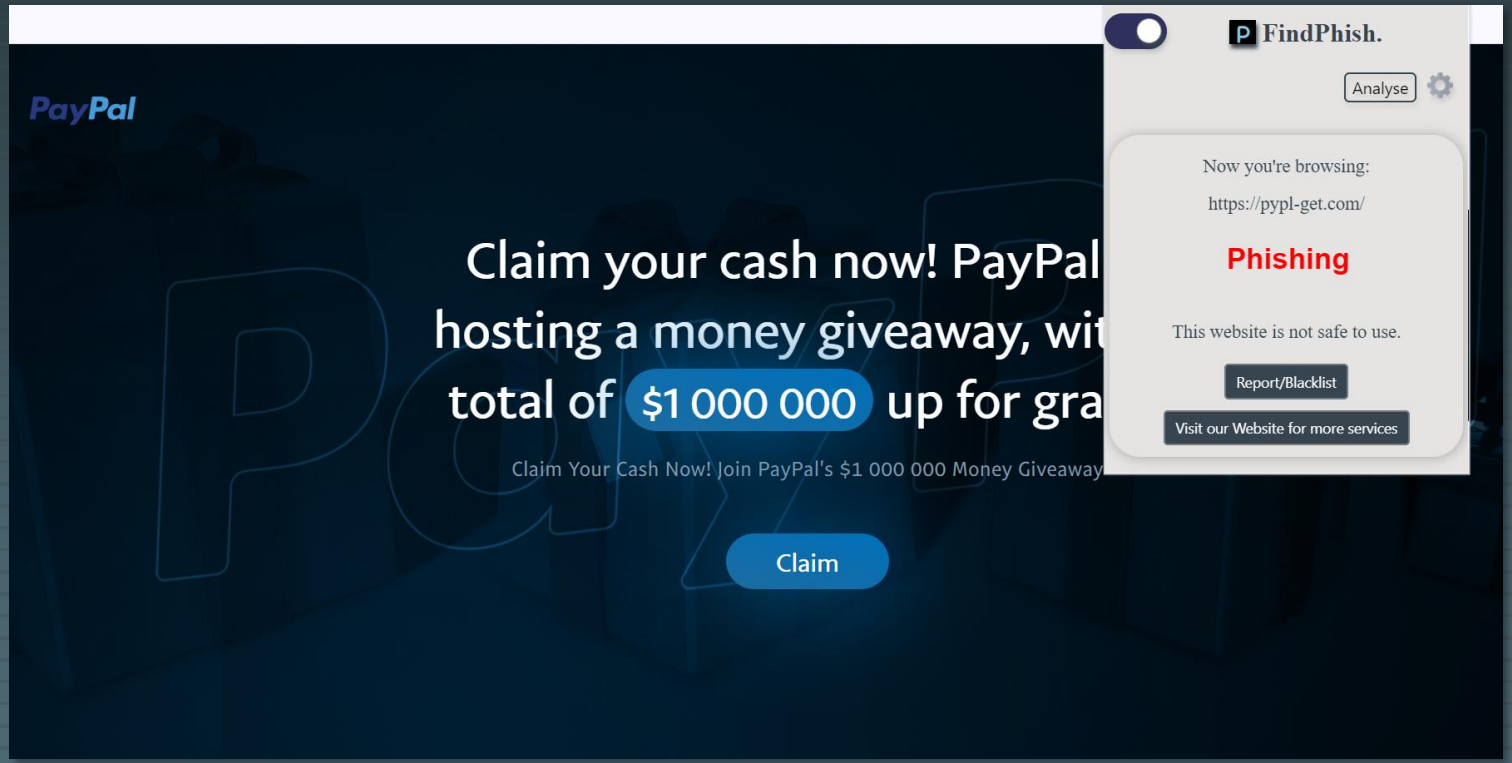
TEST CASE 1

SYSTEM REQUIREMENTS

01	Front-End	Framework: Vanilla JavaScript Languages: HTML5, CSS, JavaScript
02	Back-End	Framework: Flask Languages: Python Database: MongoDB
03	Libraries	Flask, Numpy, PyMongo, LabelEncoder, SMOTE, BertModel, BertTokenizer

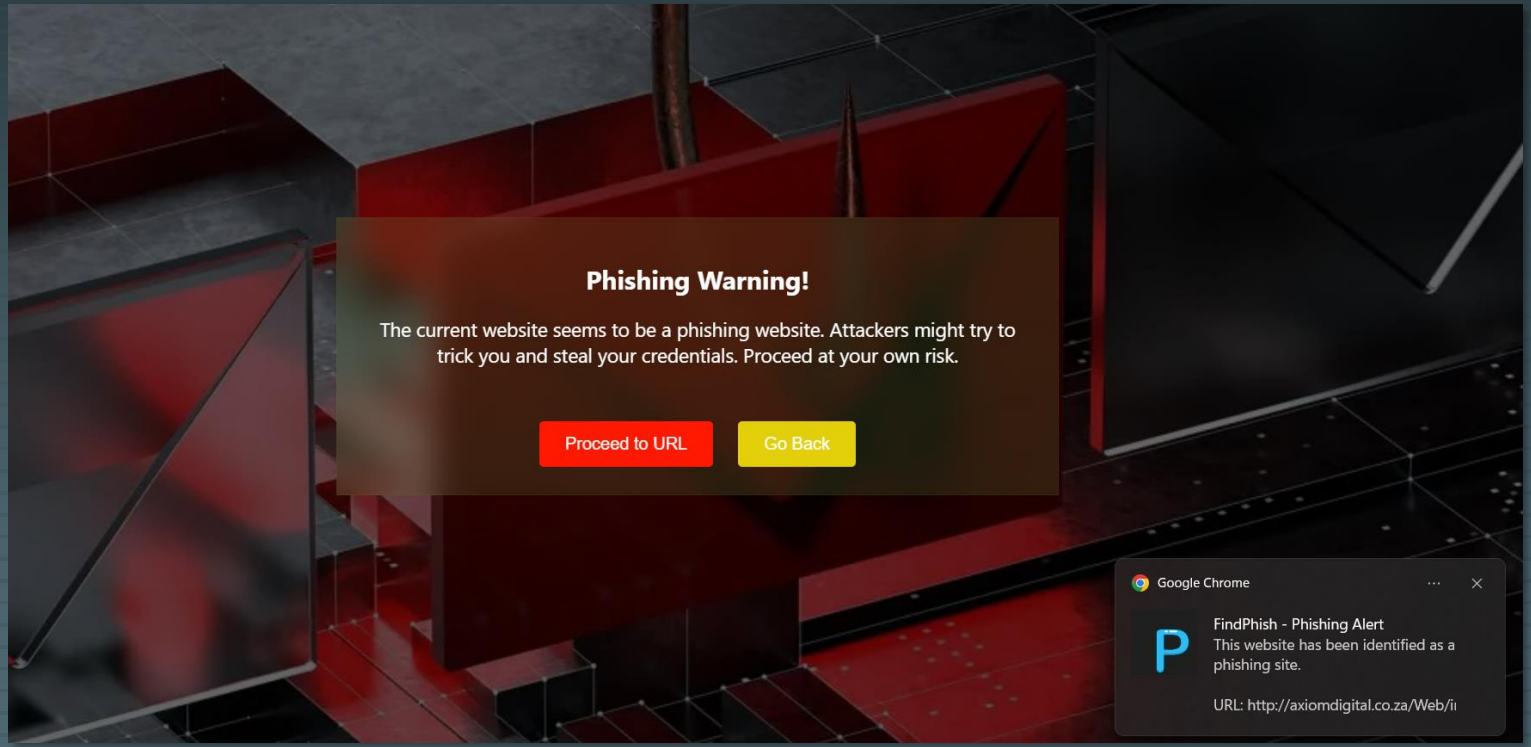
SCREENSHOTS

Feature 1: Real-Time Detection



SCREENSHOTS

Feature 2: Real-Time Notifications



SCREENSHOTS

Feature 3: Search Advisor

Google

bert

arXiv
https://arxiv.org › cs

BERT: Pre-training of Deep Bidirectional Transformers for ...

by J Devlin · 2018 · Cited by 98073 — **BERT** is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, ...

Towards Data Science
https://towardsdatascience.com › bert-explained-state-of...

BERT Explained: State of the art language model for NLP

10 Nov 2018 — **BERT** (**Bidirectional Encoder Representations from Transformers**) is a recent paper published by researchers at Google AI Language.

H2O.ai
https://h2o.ai › wiki › bert

What is BERT and how is it Used in AI?

BERT, short for **Bidirectional Encoder Representations from Transformers**, is a machine learning (ML) framework for natural language processing.

Google Blog
https://blog.google › search-language-understanding-bert

Understanding searches better than ever before

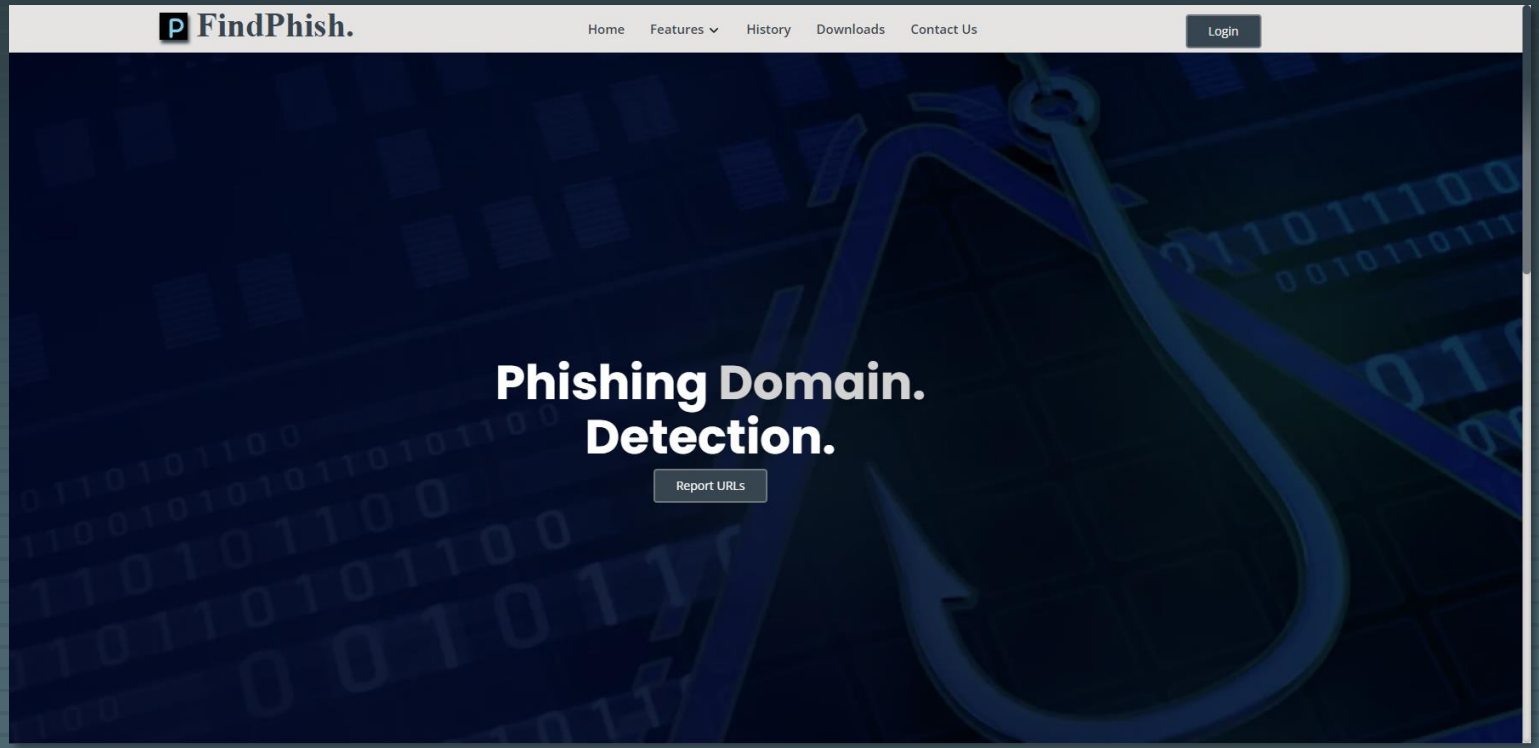
25 Oct 2019 — Here are some of the examples that showed up our evaluation process that demonstrate **BERT's** ability to understand the intent behind your search.

ACL Anthology
https://aclanthology.org › ...

BERT: Pre-training of Deep Bidirectional Transformers for ...

by J Devlin · 2019 · Cited by 98073 — Abstract. We introduce a new language representation model called **BERT**, which stands for **Bidirectional Encoder Representations from**...

SCREENSHOTS



SCREENSHOTS

Protect yourself from **phishing attacks** with the help of **FindPhish**.

Paste yout URL here, and hit the button.

http://axiomdigital.co.za/Web/index.html

CHECK URL

Trust Score : 45 / 100

URL : http://axiomdigital.co.za/Web/index.html

Preview URL within FindPhish. | Show Source Code of URL

(External scripts are disabled for your safety.)

More Information about this URL

Property	Value
Global Rank	10,00,000+
HTTP Status Code	200
Domain Age	7.1 year(s)
Use of URL Shortener	NO
HSTS Support	NO
IP instead of Domain	NO
URL Redirects	NO
IP of Domain	154.0.161.154

Protect yourself from **phishing attacks** with the help of **FindPhish**.

Paste yout IP address here, and hit the button.

192.42.116.203

CHECK IP

Risk Score: 1.11% (1/89)

Security vendors analysis

Engine Name	Category	Result
0xSI_f33d	undetected	unrated
ADMINUSLabs	harmless	clean
AILabs (MONITORAPP)	harmless	clean
Abusix	malicious	malicious
Acronis	harmless	clean
AlienVault	harmless	clean
AlphaSOC	suspicious	suspicious
Antiy-AVL	malicious	malicious
ArcSight Threat Intelligence	suspicious	suspicious
AutoShun	undetected	unrated

Feature 4: Check URL & IP

CONCLUSION

- This project aims to revolutionize phishing detection by seamlessly integrating advanced ML and AI techniques, providing a proactive and user-centric defense against evolving cyber threats.
- User experience is a central focus, with a user-friendly web application dashboard, real-time alerts, and educational resources to empower users in recognizing and reporting potential threats.

REFERENCES

- “A systematic literature review on phishing website detection techniques”, Asadullah Safi, 2023.
- “BERT Against Social Engineering Attack: Phishing Text Detection”, Nafiz Rifat, 2022.
- “https://www.kaggle.com/code/sujithmandala/phishing-domain-detection-bert-log-reg-96-ac

THANK YOU