DETECTION OF PHISHING WEBSITE USING MACHINE LEARNING MODEL BASED ON URL ANALYSIS

FINAL PROJECT PRESENTATION
SPRING 2024

-PRIYANKA PATIL

INTRODUCTION

Phishing is one of the familiar attacks that trick users to access malicious content and gain their information. In terms of website interface and uniform resource locator (URL), most phishing webpages look identical to the actual webpages. However, due to inefficient security technologies, there is an exponential increase in the number of victims. The anonymous and uncontrollable framework of the Internet is more vulnerable to phishing attacks.

In order avoid getting phished,

- Users should have awareness of phishing websites.
- Have a blacklist of phishing websites which requires the knowledge of website being detected as phishing.
- Detect them in their early appearance, using **machine learning** and deep neural network algorithms.

Of the above three, the machine learning based method is proven to be most effective than the other methods.

Even then, online users are still being trapped into revealing sensitive information in phishing websites.

OBJECTIVE

This project aims to develop a machine learning model to effectively detect phishing websites through URL analysis. Utilizing the best model stored, the project will focus on differentiating between legitimate and phishing URLs by analyzing their distinct characteristics. This includes examining the linguistic and aesthetic features of URLs commonly employed in phishing attacks. The project will leverage a dataset comprising both phishing and legitimate URLs to train and test the model, ensuring a robust and accurate classification system.

APPROACH

- **Objective:** Develop a model to detect phishing URLs using machine learning.
- **Data Collection:** Dataset of 11,000+ URLs from Kaggle, enhanced with real-time data scraping.
- **Feature Engineering:** Extracted 30 URL features using BeautifulSoup and Whois for dynamic data extraction.
- **Model Selection:** Tested different models including Logistic Regression, KNN, SVM, Random Forest, Naïve Bayes, MLP, Gradient Boosting and Decision Trees.
- **Validation:** Used k-fold cross-validation to ensure model reliability.
- **Implementation:** Built with Python, Scikit-learn, and Flask for web application deployment.
- **User Interface:** Created a user-friendly web interface for real-time phishing detection.

DATA COLLECTION AND ANALYSIS

The dataset is borrowed from Kaggle https://www.kaggle.com

A collection of website URLs for 11000+ websites. Each sample has 30 website parameters and a class label identifying it as a phishing website or not (-1 or 1).

- **Dataset Observations:**

I. There are 11054 instances and 31 features in dataset.

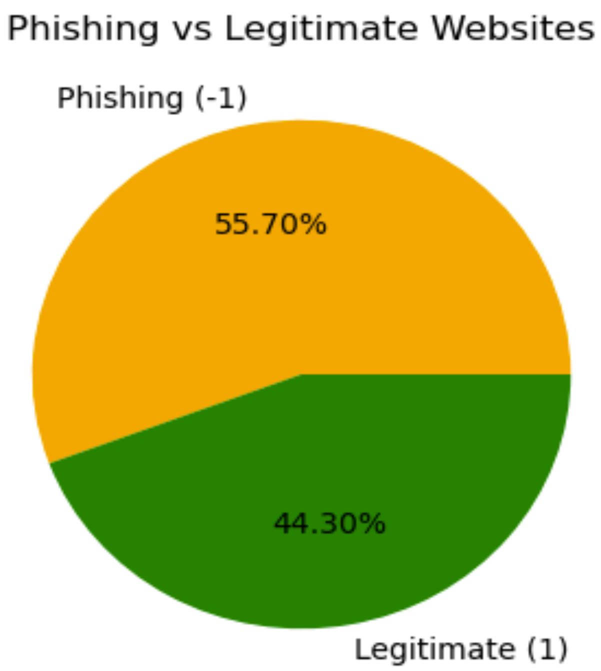
II. Out of which 30 are independent features whereas 1 is dependent feature.

III. Each feature is in int datatype, so there is no need to use Label Encoder.

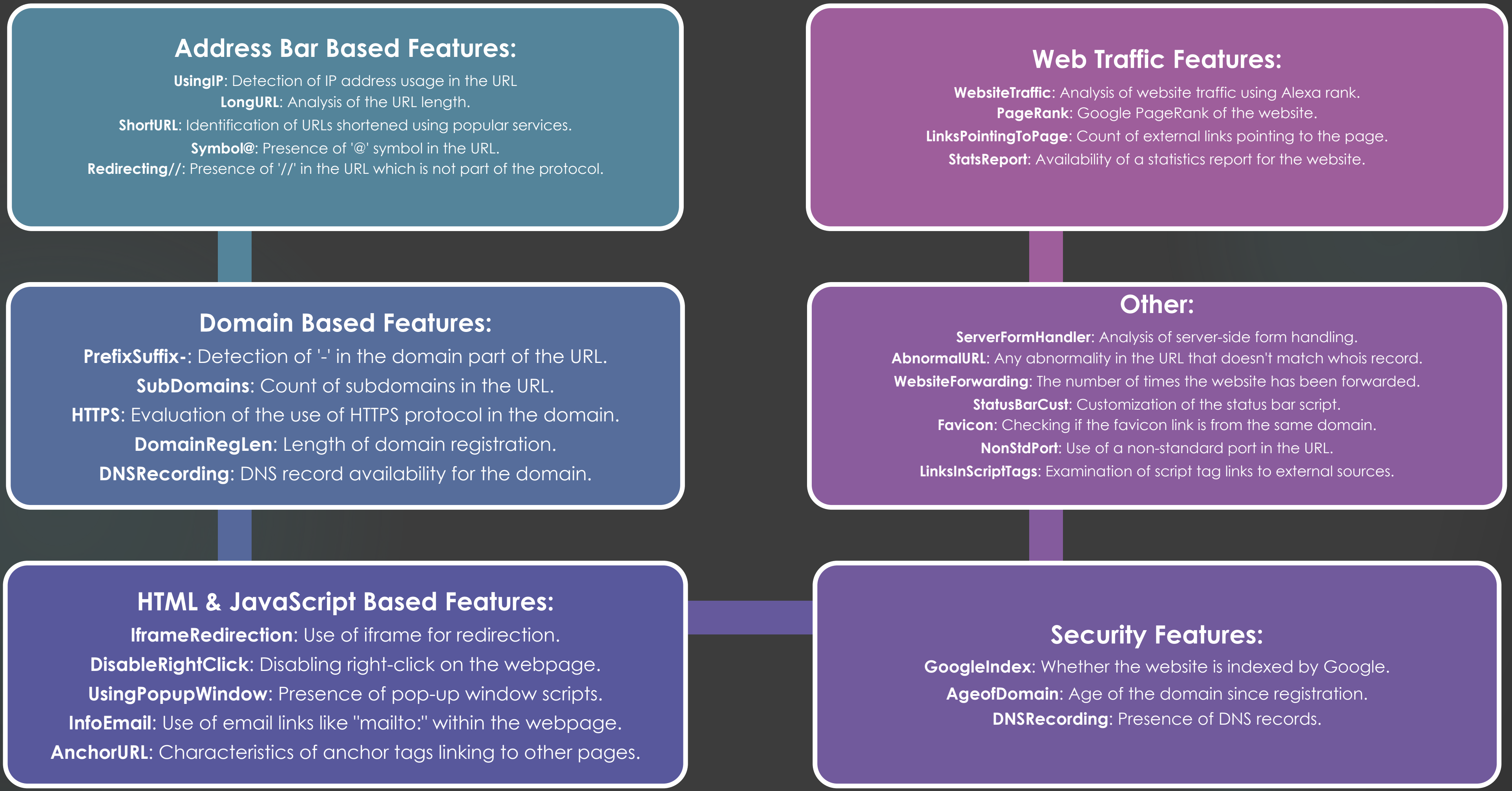
IV. There is no outlier present in dataset.

V. There is no missing value in dataset.

CLASS DISTRIBUTION



FEATURE EXTRACTION



Address Bar Based Features:

UsingIP: Detection of IP address usage in the URL
LongURL: Analysis of the URL length.
ShortURL: Identification of URLs shortened using popular services.
Symbol@: Presence of '@' symbol in the URL.
Redirecting//: Presence of '//' in the URL which is not part of the protocol.

Domain Based Features:

PrefixSuffix-: Detection of '-' in the domain part of the URL.
SubDomains: Count of subdomains in the URL.
HTTPS: Evaluation of the use of HTTPS protocol in the domain.
DomainRegLen: Length of domain registration.
DNSRecording: DNS record availability for the domain.

HTML & JavaScript Based Features:

IframeRedirection: Use of iframe for redirection.
DisableRightClick: Disabling right-click on the webpage.
UsingPopupWindow: Presence of pop-up window scripts.
InfoEmail: Use of email links like "mailto:" within the webpage.
AnchorURL: Characteristics of anchor tags linking to other pages.

Web Traffic Features:

WebsiteTraffic: Analysis of website traffic using Alexa rank.
PageRank: Google PageRank of the website.
LinksPointingToPage: Count of external links pointing to the page.
StatsReport: Availability of a statistics report for the website.

Other:

ServerFormHandler: Analysis of server-side form handling.
AbnormalURL: Any abnormality in the URL that doesn't match whois record.
WebsiteForwarding: The number of times the website has been forwarded.
StatusBarCust: Customization of the status bar script.
Favicon: Checking if the favicon link is from the same domain.
NonStdPort: Use of a non-standard port in the URL.
LinksInScriptTags: Examination of script tag links to external sources.

Security Features:

GoogleIndex: Whether the website is indexed by Google.
AgeofDomain: Age of the domain since registration.
DNSRecording: Presence of DNS records.

MACHINE LEARNING MODELS

- This is a supervised machine learning task. There are two major types of supervised machine learning problems, called classification and regression.

- This data set comes under classification problem, as the input URL is classified as phishing (-1) or legitimate (1). The machine learning models (classification) considered to train the dataset in this project are:

i. Logistic Regression

ii. K-Nearest Neighbors

iii. Support Vector Machine

iv. Naïve Bayes Classifier

v. Decision Tree

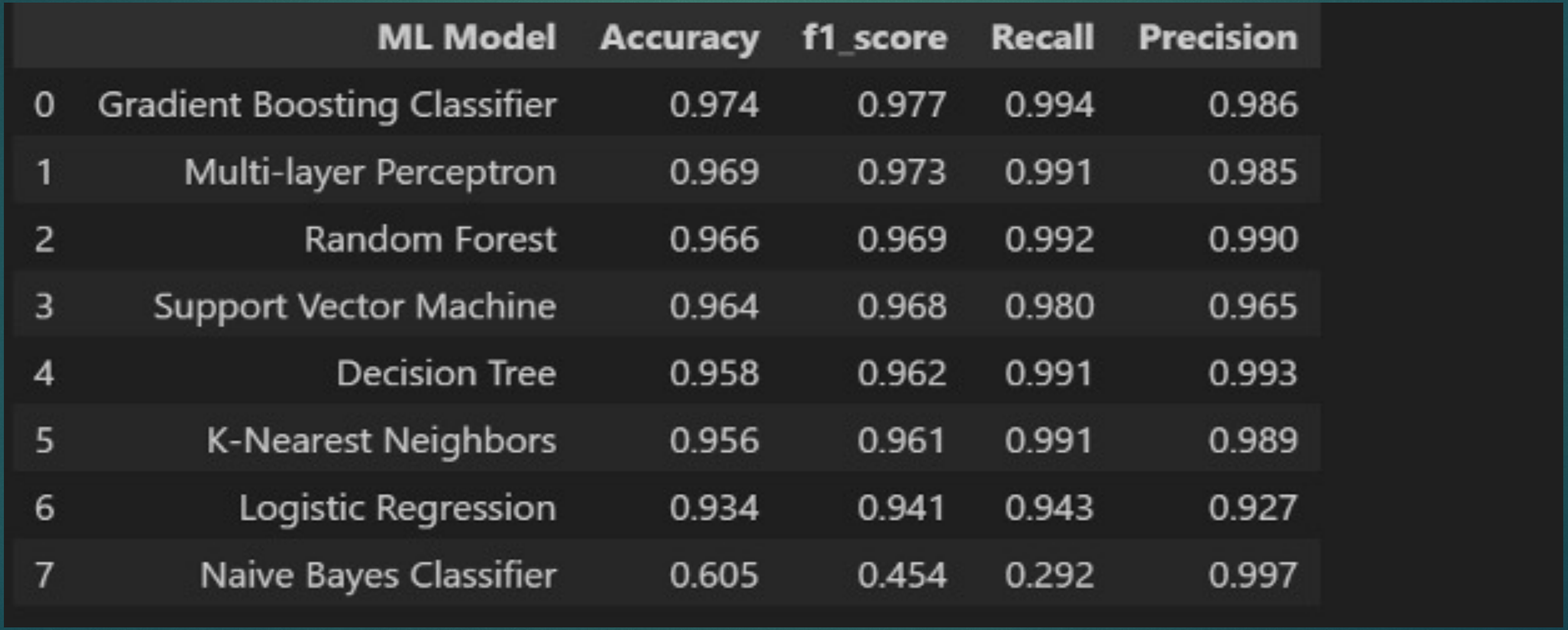
vi. Random Forest

vii. Gradient boosting Classifier

viii. Multi Layer Perceptron

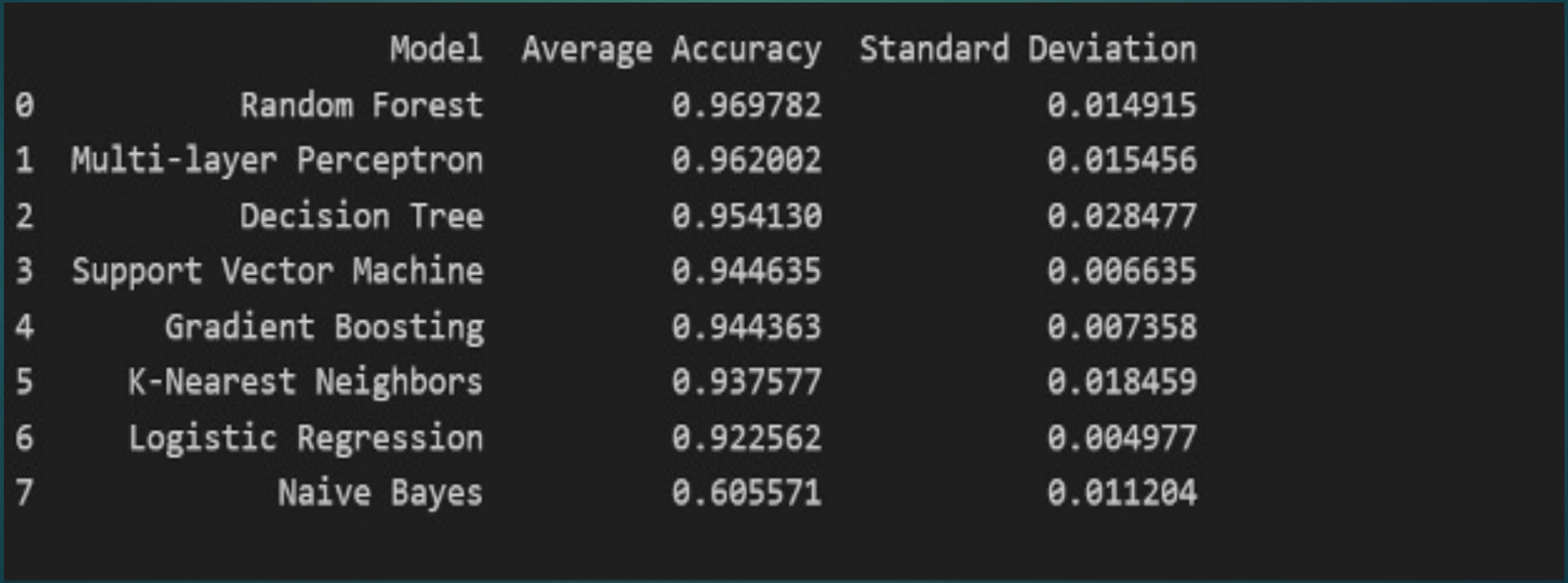
MODEL EVALUATION

- The metrics considered to evaluate the model performance are Accuracy & F1 score.



	ML Model	Accuracy	f1_score	Recall	Precision
0	Gradient Boosting Classifier	0.974	0.977	0.994	0.986
1	Multi-layer Perceptron	0.969	0.973	0.991	0.985
2	Random Forest	0.966	0.969	0.992	0.990
3	Support Vector Machine	0.964	0.968	0.980	0.965
4	Decision Tree	0.958	0.962	0.991	0.993
5	K-Nearest Neighbors	0.956	0.961	0.991	0.989
6	Logistic Regression	0.934	0.941	0.943	0.927
7	Naive Bayes Classifier	0.605	0.454	0.292	0.997

Model Evaluation(K-Fold Cross Validation)



```
                  Model  Average Accuracy  Standard Deviation
0         Random Forest          0.969782            0.014915
1  Multi-layer Perceptron        0.962002            0.015456
2         Decision Tree          0.954130            0.028477
3  Support Vector Machine        0.944635            0.006635
4     Gradient Boosting          0.944363            0.007358
5   K-Nearest Neighbors          0.937577            0.018459
6   Logistic Regression          0.922562            0.004977
7           Naive Bayes          0.605571            0.011204
```

Model Selection:

The Gradient Boosting Classifier was chosen for its balance of high accuracy and performance consistency in both individual and cross-validated tests.

While Random Forest excelled in average cross-validation accuracy, the Gradient Boosting Classifier's superior precision and recall in practical application settings make it particularly effective for our use case of phishing detection, where both identifying as many true phishing instances as possible (recall) and ensuring accurate identifications (precision) are crucial.

Model Deployment:

To make our phishing detection model accessible and user-friendly, we deployed it using a Flask web application. This setup allows users to input URLs and receive real-time assessments of their safety.

User Interface:

The interface is straightforward, enabling users to easily submit URLs for phishing detection. The backend processes the input using our trained Gradient Boosting Classifier to predict and return the likelihood of phishing.

Continuing Challenges:

i. **Adaptive Phishing Techniques:** As cybercriminals evolve their methods, our systems must adapt to detect ever-changing phishing tactics.

ii. **Handling Zero-Day Phishing Attacks:** Identifying and mitigating attacks that exploit previously unknown vulnerabilities remains a significant challenge.

iii. **Data Privacy Concerns:** Ensuring user data is handled securely, especially when scaling the solution to handle more sensitive information.

FUTURE DEVELOPMENT:

i. **Enhancing Detection Algorithms:** Implementing deep learning techniques to improve accuracy and reduce false positives.

ii. **Real-time Data Analysis:** Developing capabilities to analyze and block phishing attempts in real time as they happen.

iii. **User Education Integration:** Creating interactive and engaging educational tools to help users recognize phishing attempts.

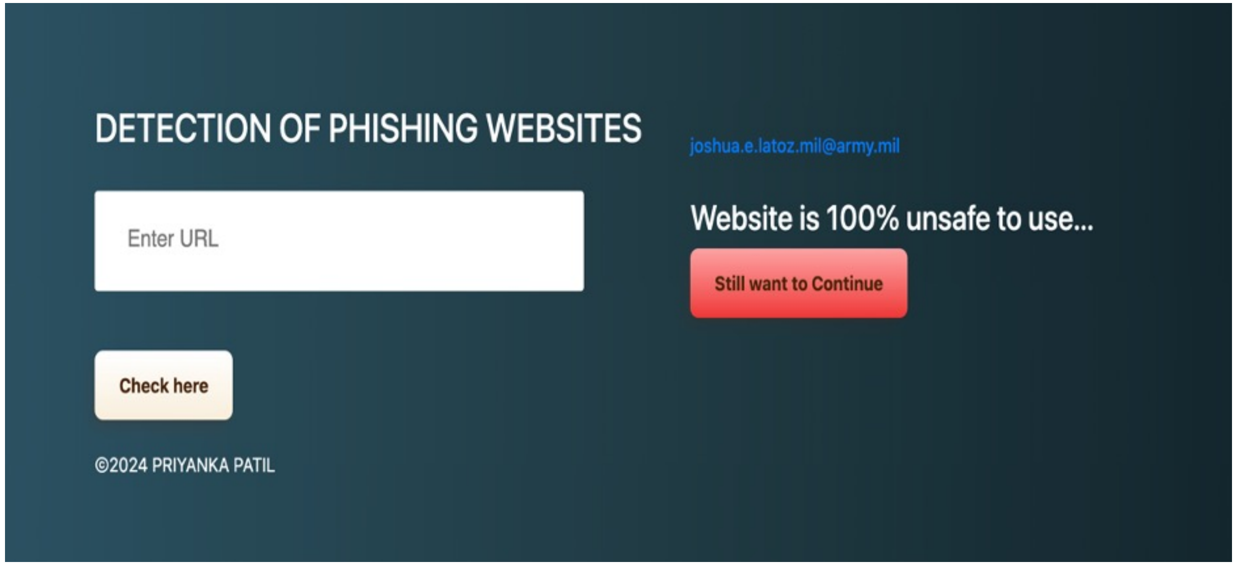
iv. **Collaboration with Cybersecurity Platforms:** Partnering with existing cybersecurity platforms to integrate our solution for a broader reach.

RESULTS VISUALISATION:

LIVE DEMO

During this presentation, we will conduct a live demonstration of our Flask-based web application in action. This will provide a firsthand look at how our phishing detection system evaluates and classifies URLs in real time.

MODEL DEMO RESULTS

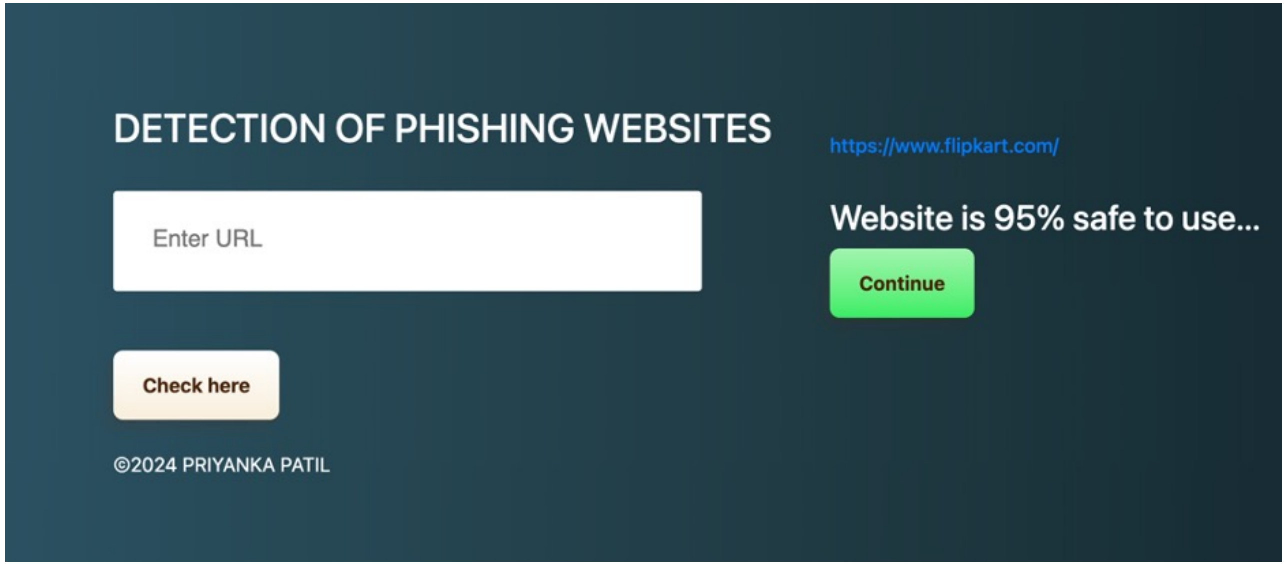


DETECTION OF PHISHING WEBSITES

joshua.e.latoz.mil@army.mil

Website is 100% unsafe to use...

©2024 PRIYANKA PATIL



DETECTION OF PHISHING WEBSITES

https://www.flipkart.com/

Website is 95% safe to use...

©2024 PRIYANKA PATIL

THANK YOU FOR JOINING ☺

Before you go!!

Remember every click counts – make sure it’s a safe one.