CYBER SECURITY INTERNSHIP

Submitted in partial fulfillment for the award of certificate of

BACHELOR OF TECHNOLOGY

IN COMPUTER SCIENCE AND ENGINEERING – ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

By

Navya Sri .V(218T1A4251)

DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY

GANGURU, VIJAYAWADA - 521 139

Affiliated to JNTUK, Kakinada &Approved By AICTE,

New Delhi Certified by ISO 9001-2015, Accredited By NBA

DHANEKULA INSTITUTE OF ENGINEERING&TECHNOLOGY

GANGURU, VIJAYAWADA - 521 139

Affiliated to JNTUK, Kakinada &Approved By AICTE, New Delhi Certified by ISO 9001-2015, Accredited by NBA

Department of Computer Science & Engineering – Artificial Intelligence & Machine Learning

CERTIFICATE

This is to certify that the Summer Internship work entitled **"CHROME EXTENSION FOR DETECTING PHISHING WEBSITES"** is a bonafied record of internship work done by NAVYA SRI.V(218T1A4251) for the award of the Summer Internship in Computer Science and Engineering -Artificial Intelligence & Machine Learning by Jawaharlal Nehru Technological University, Kakinada during the academic year 2024 - 2025.

Head of Department:
Dr. CH. SURESH BABU
Professor, HOD CSE- AI& ML

EXTERNAL EXAMINER

DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY

Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning)

DEPARTMENT VISION & MISSION

DEPARTMENT VISION

To empower students of Computer Science and Engineering (Artificial Intelligence & Machine Learning) Department to be technologically adept, innovative, global citizens possessing human values.

DEPARTMENT MISSION

- Encourage students to become self-motivated and problem-solving individuals.
- Prepare students for professional career with academic excellence and leadership skills.
- Empower the rural youth with computer education.
- Create Centre's of excellence in Computer Science and Engincering (Artificial Intelligence & Machine Learning).

PROGRAM OUTCOMES(PO'S)

1. **Engineering knowledge:** apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem Analysis:** identify, formulate, review research literature, and analyze complex engineering problems reaching sustained conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/Development Of Solutions:** design solutions for complex engineering problems and design system components or process that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct Investigations Of Complex Problems:** use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern Tool Usage:** create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The Engineer And Society:** apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment And Sustainability:** understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for susta -inable development.
8. **Ethics:** apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual And Team Work:** function effectively as an individual, and as a member or a leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project Management And Finance:** demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life- Long Learning:** recognize the need for, and have the preparation and ability to engage in independent and life- long learning in broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES(PSOs)

PSO1: Have comprehensive expertise in machine learning, deep learning algorithms, networking, database management systems and web based applications to design and develop optimized, efficient systems.

PSO2: Qualify in national and international level competitive examinations for successful higher studies and get employment .

Internship Mappings

Project Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CHROME EXTENSION FOR DETECTING PHISHING WEBSITES	3	3	2	2	3	3	2	3	3	2	3	3	3	3

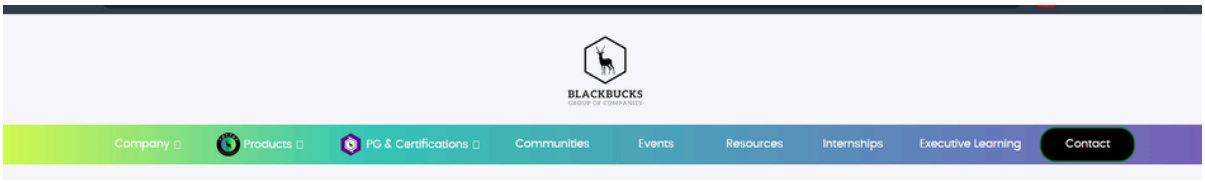
Mapping Level	Mapping Description
1	Low Level Mapping with PO & PSO
2	Moderate Mapping with PO & PSO
3	High Level Mapping with PO & PSO

NAME : Navya Sri .Vangala
ROLLNO : 218T1A4251
CLASS : IV- I

Contents

1. Internship carried out Company/Organiazation.
2. Duration of Internship & Internship Log-Task schedule.
3. Domain/Area of the Internship.
4. Project documentation report carried out during Internship.
5. Internship completion Certificate.

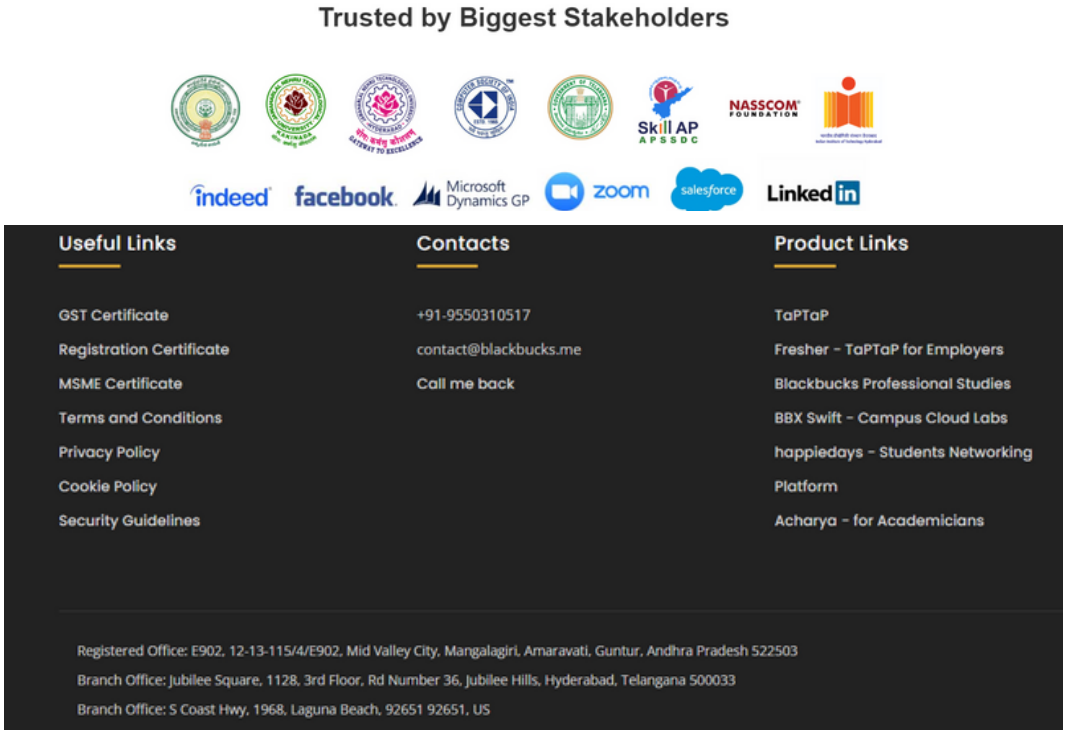
INTERNSHIP CARRIED OUT COMPANY DETAILS :



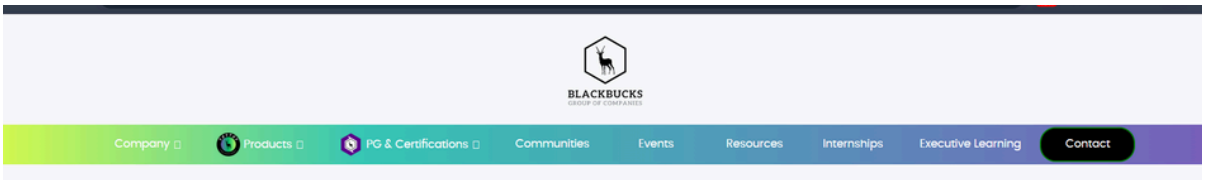
Name: BlackBucks **Place: West Godavari (Dist.), AP,India.** **Established on: 2013**

Blackbucks is an innovative campus success platform designed to enhance student employability through various AI-driven applications and post-graduation programs. The platform offers a comprehensive suite of tools for students and recruiters, focusing on data science, cybersecurity, and software engineering. As a leader in educational technology in India, Blackbucks collaborates with prestigious universities and organizations to provide placement guarantees and extensive internship opportunities for students.

- **Blackbucks provides a unique software suite, TaPTaP, which aids in the preparation, practice, and placement of students using analytics.**
- **The platform features recruitment apps specifically tailored for GenZ, simplifying the hiring process for recruiters.**
- **BBX Swift offers automated grading for BTech labs, aligning with academic standards set by AICTE/JNTU.**
- **Advanced certification programs in areas like AI, ML, and cybersecurity are available, emphasizing industry-led education.**
- **Blackbucks has partnered with APSCHE to help train and assess over 100,000 students in Andhra Pradesh.**
- **The platform has enrolled more than 5,000 students in various internship programs across multiple tech fields.**
- **With over ten years of service, Blackbucks celebrates a decade of transforming student employability through technology and training initiatives.**



INTERNSHIP LOG :



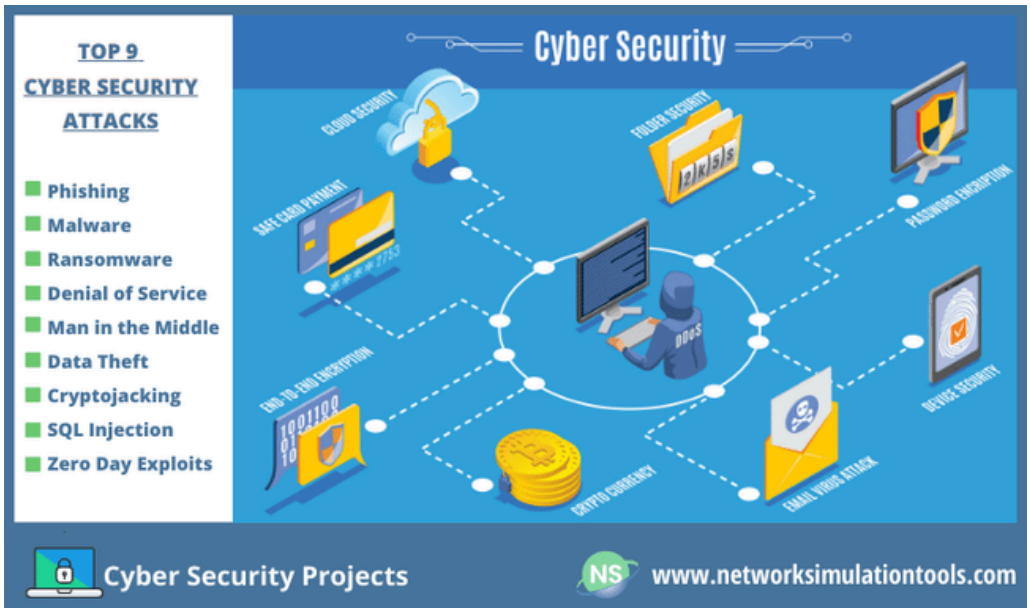
Company Name : Blackbucks
Intern Name : Navya Sri .V
Intern Period : 05 / 06 /2024 - 31 /07 /2024

WEEK WISE Schedule :

WEEKS	HOURS	TOPIC
WEEK-1	2.5 Hrs	Introduction to Cyber Security
WEEK-2	2.5 Hrs	Networking Fundamentals
WEEK-3	2.5 Hrs	Operating Systems Fundamentals
WEEK-4	2.5 Hrs	Footprinting & Reconnaissance
WEEK-5	2.5 Hrs	Enumeration & Scanning
WEEK-6	2.5 Hrs	Introduction to Web Application Security
WEEK-7	2.5 Hrs	Ethical Hacking Tools & Techniques
WEEK-8	2.5 Hrs	Course Wrap-up & Project Review

- **DOMIAN AREA OF INTERNSHIP :**

A cybersecurity internship offers an invaluable opportunity to delve into the dynamic world of information security. Throughout the 12-week program, interns gain hands-on experience working alongside seasoned security analysts and engineers. They tackle real-world projects, such as developing security protocols, conducting vulnerability assessments, and implementing threat detection systems. With access to cutting-edge tools and technologies, interns refine their skills in network security, penetration testing, and incident response. Collaborating in a vibrant and innovative environment, they learn to analyze security risks, respond to incidents, and interpret security reports. Mentorship sessions and weekly seminars further enrich their knowledge. By the end of the internship, participants emerge with enhanced problem-solving capabilities and a deep understanding of cybersecurity concepts, positioning them for future success in this rapidly evolving field.



PROJECT REPORT :

1. INTRODUCTION

1.1 Objective

1.2 Data Sources & Technology

1.3 Impact & Future Scope

1.4 Dis advantages

1.5 Proposed System

1.6 Snapshots Of Output

2. SOURCE CODE

- Design Structure
- Pre Processing
- Training
- Classifier Dump
- Final result

3. CONCLUSION

INTERNATIONAL INSTITUTE OF DIGITAL TECHNOLOGIES, TIRUPATI - 517520

(Information Technology, Electronics & Communication Department, Government of Andhra Pradesh,)

and

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of Andhra Pradesh.)

Certificate of Completion

Certificate Id: BBAPSCHEIIDT2024STM101909

This is to certify that Mr/Ms NAVYA SRI . V, the Roll No: 218T1A4251 from Dhanekula Institute of Engineering and Technology JNTU Kakinada, successfully completed the Eight week duration Internship program in “Cyber Security” conducted by the International Institute of Digital Technologies,Tirupati, the Blackbuck Engineers as the Knowledge Partner, and Andhra Pradesh State Council of Higher Education (APSCHE). She/He scored 97 out of 100 in the internship program and His/Her performance is "Excellent".

Anuradha Thota
Chief Executive Officer
Blackbuck Engineers Pvt. Ltd.

Dr. Sundar Balakrishna
Director General
International Institute of Digital Technologies

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P.)

Certificate of Completion

Certificate Id: BBAPSCHEIIDT2024ST103508

This is to certify that NAVYA SRI . V , bearing Reg. No: 218T1A4251, from Dhanekula Institute of Engineering and Technology of JNTU Kakinada , has successfully completed a Short-term internship for 8 Weeks on Cyber Security. This internship wasorganized by International Institute of Digital Technologies, with its industry partner Blackbuck Engineers, in association with the Andhra Pradesh State Council of Higher Education (APSCHE).

Anuradha Thota
Chief Executive Officer
Blackbuck Engineers Pvt. Ltd.

Dr. Sundar Balakrishna
Director General
International Institute of Digital Technologies

Date: 24/07/2024Place: Tirupati, Andhra Pradesh

PROJECT REPORT

1 INTRODUCTION

Chrome Extension for Detecting Phishing Websites: A Data-Driven Approach to Secure Browsing

In the modern digital landscape, protecting users from phishing websites has become an essential part of cybersecurity. Phishing attacks, which trick users into disclosing sensitive information, are a constant threat. To address this issue, we present the "Chrome Extension for Detecting Phishing Websites," a cutting-edge project that leverages data science and machine learning to identify phishing websites in real time. Our goal is to empower users with timely alerts, enabling them to make informed decisions and avoid potential security risks.

Phishing has always been a major concern for internet users, putting millions of people at risk worldwide. Machine learning, a technique that enables computers to learn from data and patterns, plays a crucial role in detecting phishing attempts without explicit instructions. These algorithms have the ability to analyze vast amounts of data and uncover patterns that are not immediately apparent to humans.

- By using machine learning algorithms, this project enables more accurate detection of phishing websites, helping users stay safe while browsing. One of the primary benefits of using machine learning to identify phishing sites is its precision and ability to adapt to evolving threats.
- The algorithms analyze a wide range of data, including website URL structures, SSL certificates, and common phishing site features. The model then uses this data to create real-time predictions, flagging potentially dangerous websites. Phishing websites often change tactics based on factors such as time and the security landscape, and this model adapts accordingly. The ultimate goal is to provide users with a secure browsing experience, while attackers continuously try to exploit vulnerabilities.
- Our proposed framework focuses on building an intelligent system that improves over time by learning from past phishing websites and user interactions. This project falls under the domain of Artificial Intelligence, specifically a subfield of Machine Learning. Machine learning, which can be supervised or unsupervised, is vast and continues to grow in importance in the field of cybersecurity. The proposed framework uses selected features from websites to build a prediction model, which generates output to determine whether a website is legitimate or a phishing attempt.

Proposed framework for Phishing Detection using data sources are as follows:

1.Data Pre-Processing :

Datasets used to train phishing detection models may contain irrelevant or duplicate fields, as well as erroneous data from previous reporting or human error. Proper data preprocessing ensures that accurate and reliable input data is used for building the machine learning model. The pre-processing phase involves cleaning, normalizing, and filtering out irrelevant or redundant data to ensure optimal model performance.

2.Feature Extraction:

Several features are extracted from the dataset to represent key aspects of phishing websites. These features may include URL characteristics, the presence or absence of certain elements like security certificates, and metadata such as the number of links, input fields, and domain age. This step allows the model to capture relevant phishing patterns. Additionally, macroeconomic and global factors like geographical information are also considered to improve prediction accuracy.

3.Feature Selection:

Feature selection techniques are employed to enhance the model's performance by analyzing the impact of each feature on the final prediction result. A feature's importance is measured based on how much it contributes to reducing impurity in decision paths. This step helps in prioritizing the most critical factors that influence whether a website is phishing or legitimate.

4.Prediction Model :

A Random Forest Classifier was chosen for the task of phishing detection. Based on empirical testing, it provides the best performance when compared to other machine learning techniques. The model is trained using labeled datasets and learns to identify phishing sites based on patterns in the extracted features.

1.1 OBJECTIVE

The primary goal of the "Chrome Extension for Detecting Phishing Websites" is to develop a reliable, real-time phishing detection tool that enhances user security while browsing. By collecting and analyzing data from phishing sites and comparing them to legitimate ones, the system utilizes machine learning algorithms to classify websites. The extension operates on the client-side to preserve user privacy while providing effective phishing detection. Users can navigate the web with increased confidence, knowing that potentially malicious sites will be flagged.

1.2 DATA SOURCES AND TECHNOLOGY

To achieve the goal of effective phishing detection, the Chrome extension utilizes Python 3.7 and the scikit-learn library, specifically version 0.19.2, for implementing a Random Forest Classifier. The classifier is trained using a dataset sourced from the UCI Repository, which provides a comprehensive collection of phishing patterns. Before training, the data undergoes rigorous preprocessing to ensure accuracy and relevance.

Key dependencies include numpy version 1.15.0 for handling numerical operations and liac-arff version 2.2.2 for loading ARFF files. The Random Forest Classifier is chosen for its ability to manage high-dimensional data, resist overfitting, and efficiently classify websites in real-time. By leveraging these tools and techniques, the extension offers a robust solution for detecting phishing sites based on various website characteristics, ensuring users are protected while browsing.

DATASET : UCI Repository ; TECHNIQUE : Random Forest Classifier .

1.3 IMPACT AND FUTURE SCOPE

The implications of the Chrome Extension for Phishing Detection go far beyond individual users, aiming to create a safer digital environment for all. By empowering users to identify and avoid malicious websites, the extension contributes to enhanced cybersecurity awareness and reduced risks of data breaches. As the platform evolves, we envision integrating additional features such as real-time threat reporting and advanced analytics, transforming it into a comprehensive cybersecurity tool. Furthermore, we aim to collaborate with industry stakeholders, including browser developers and cybersecurity firms, to strengthen online safety measures globally. In conclusion, the Chrome Extension for Phishing Detection aspires to revolutionize web security, equipping users with the tools and knowledge to navigate the internet confidently and securely.

1.4 DISADVANTAGES

While Chrome extensions for phishing detection offer significant benefits, they also face challenges such as evolving phishing tactics, which make accurate detection complex and uncertain. Limited data availability and quality, coupled with the need for real-time updates, can hinder the system's effectiveness. The scope of detection may be narrow, focusing on specific patterns like URLs or SSL certificates, potentially missing advanced phishing attempts. Overfitting to training data can reduce generalization to new threats, and incomplete information may lead to false positives or negatives. Additionally, user perception can be affected by inaccuracies, and privacy or regulatory concerns may arise when analyzing website data and user interactions. Despite these challenges, phishing detection extensions remain essential tools, and their effectiveness can be improved through continuous model updates, integration of diverse data, and transparency in detection processes.

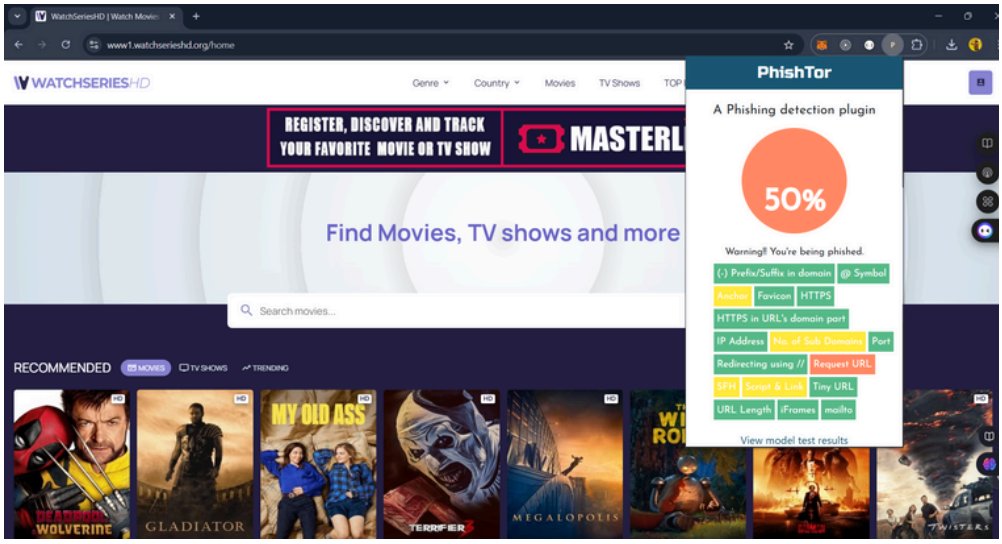
1.5 Proposed System

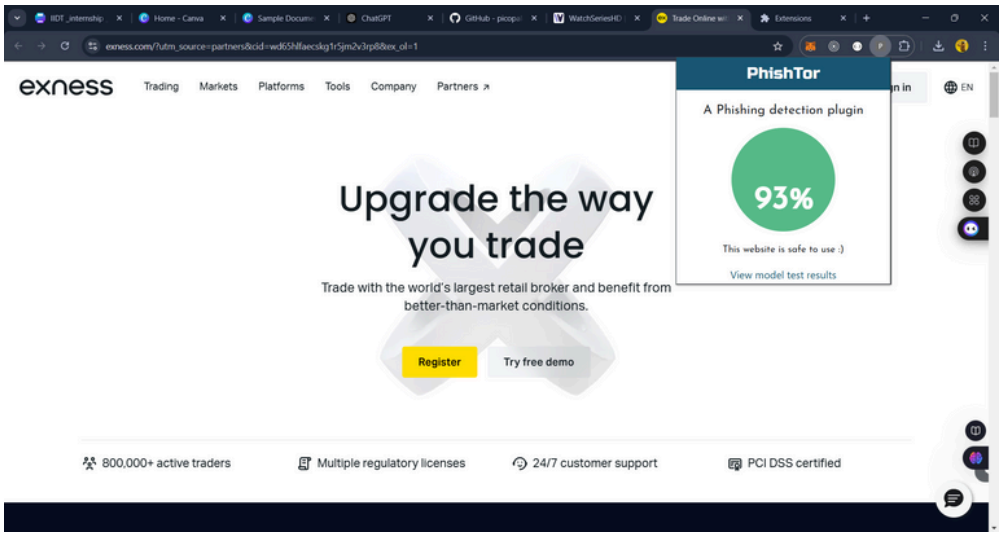
After thoroughly analyzing the challenges faced by existing phishing website detection systems, we adopted a data-driven approach to create a cutting-edge solution that prioritizes both accuracy and user privacy. Using machine learning, specifically a Random Forest Classifier, and leveraging the UCI Repository dataset, our team developed an advanced model that detects phishing websites efficiently. The classification process is conducted entirely on the client side, ensuring that no browsing data is collected, preserving the user's privacy.

By using critical features and training a robust model, we achieved an impressive F1 score of 0.905. The model is packaged as a lightweight Chrome plugin, which requires only a one-time download of the classifier, making it easy to install and use.

The demo version of the plugin showcases its seamless integration with your browser and demonstrates how it effectively warns users about potential phishing threats. Our goal is to provide a reliable, privacy-respecting tool that enhances online security without compromising the user's data. With ongoing updates and improvements, this phishing website detection tool aims to revolutionize web security, providing users with the knowledge and confidence to browse safely.

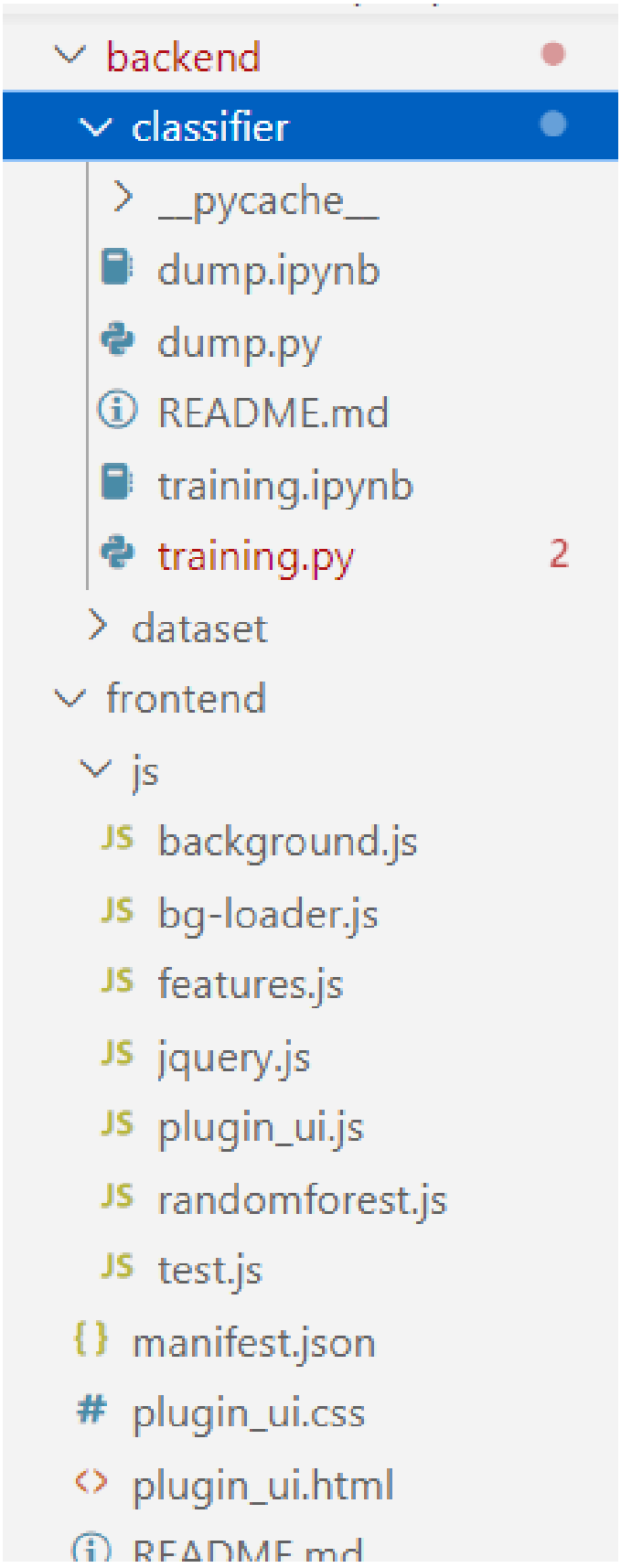
1.6 Snapshots of Output :





SOURCE CODE

Design structure :



PRE PROCESSING :

```python
import json
from sklearn.model_selection import train_test_split, KFold
# In[17]:
dataset = arff.load(open('dataset.arff', 'r'))
data = np.array(dataset['data'])
# In[18]:
print('The dataset has {0} datapoints with {1} features'.format(data.shape[0], data.shape[1]-1))
print('Features: {0}'.format([feature[0] for feature in dataset['attributes']]))
# In[19]:
data = data[:, [0, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 22, 30]]
# In[20]:
X, y = data[:, :-1], data[:, -1]
y.reshape(y.shape[0])
print('Before spliting')
print('X:{0}, y:{1}'.format(X.shape, y.shape))
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)
print('After spliting')
print('X_train:{0}, y_train:{1}, X_test:{2}, y_test:{3}'.format(X_train.shape, y_train.shape, X_test.shape, y_test.shape))
# In[21]:
np.save('X_train.npy', X_train)
np.save('X_test.npy', X_test)
np.save('y_train.npy', y_train)
np.save('y_test.npy', y_test)
print('Saved!')
# In[24]:
test_data = dict()
test_data['X_test'] = X_test.tolist()
test_data['y_test'] = y_test.tolist()
with open('../../static/testdata.json', 'w') as tdfile:
    json.dump(test_data, tdfile)
    print('Test Data written to testdata.json')
```

OUTPUT :

```
PS C:\Users\navya\OneDrive\Desktop\phishing-detection-plugin-master\backend\dataset> python3 preprocess.py
The dataset has 11055 datapoints with 30 features
Features: ['having_IP_Address', 'URL_Length', 'Shortining_Service', 'having_At_Symbol', 'double_slash_redirecting', 'Prefix_Suffix', 'having_Sub_Domain', 'SSLfina
e', 'Domain_registeration_length', 'Favicon', 'port', 'HTTPS_token', 'Request_URL', 'URL_of_Anchor', 'Links_in_tags', 'SFH', 'Submitting_to_email', 'Abnormal_URL',
irect', 'on_mouseover', 'RightClick', 'popUpWidnow', 'Iframe', 'age_of_domain', 'DNSRecord', 'web_traffic', 'Page_Rank', 'Google_Index', 'Links_pointik',k', 'Goog
k', 'Google_Index', 'Links_pointing_to_page', 'Statistical_report', 'Result']
Before spliting
X:(11055, 17), y:(11055,)
After spliting
X_train:(7738, 17), y_train:(7738,), X_test:(3317, 17), y_test:(3317,)
Saved!
Test Data written to testdata.json
```

TRAINING :

```python
# coding: utf-8
# In[1]:
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import cross_val_score
from sklearn.metrics import accuracy_score
import numpy as np
import json
import dump
# In[2]:
X_train = np.load('../dataset/X_train.npy')
y_train = np.load('../dataset/y_train.npy')
print('X_train:{0}, y_train:{1}'.format(X_train.shape, y_train.shape))
# In[3]:
clf = RandomForestClassifier()
print('Cross Validation Score: {0}'.format(np.mean(cross_val_score(clf, X_train, y_train, cv=10))))
# In[4]:
clf.fit(X_train, y_train)
# In[5]
X_test = np.load('../dataset/X_test.npy')
y_test = np.load('../dataset/y_test.npy')
# In[6]:
pred = clf.predict(X_test)
print('Accuracy: {}'.format(accuracy_score(y_test, pred)))
# In[7]:
#print(forest_to_json(clf))
json.dump(dump.forest_to_json(clf), open('../../static/classifier.json', 'w'))
```

OUTPUT :

```
PS C:\Users\navya\OneDrive\Desktop\phishing-detection-plugin-master\backend\classifier> python3 training.py
X_train:(7738, 17), y_train:(7738,)
Cross Validation Score: 0.9466257843029107
Accuracy: 0.9460355743141393
```

dump.py | training.py | background.js | test.js

backend > classifier > dump.py > ...

```python
# coding: utf-8
```

Run Cell | Run Below | Debug Cell

```python
# In[9]:
from sklearn.tree import _tree
```

Run Cell | Run Above | Debug Cell

```python
# In[10]:
def tree_to_json(tree):
    tree_ = tree.tree_
    feature_names = range(30)
    feature_name = [
        feature_names[i] if i != _tree.TREE_UNDEFINED else "undefined!"
        for i in tree_.feature
    ]
    def recurse(node):
        tree_json = dict()
        if tree_.feature[node] != _tree.TREE_UNDEFINED:
            tree_json['type'] = 'split'
            threshold = tree_.threshold[node]
            tree_json['threshold'] = "{} <= {}".format(feature_name[node], threshold)
            tree_json['left'] = recurse(tree_.children_left[node])
            tree_json['right'] = recurse(tree_.children_right[node])
        else:
            tree_json['type'] = 'leaf'
            tree_json['value'] = tree_.value[node].tolist()
        return tree_json
    return recurse(0)
```

Run Cell | Run Above | Debug Cell

```python
# In[11]:
def forest_to_json(forest):
    forest_json = dict()
    forest_json['n_features'] = forest.n_features_in_
    forest_json['n_classes'] = forest.n_classes_
    forest_json['classes'] = forest.classes_.tolist()
    forest_json['n_outputs'] = forest.n_outputs_
    forest_json['n_estimators'] = forest.n_estimators
    forest_json['estimators'] = [tree_to_json(estimator) for estimator in forest.estimators_]
    return forest_json
```

dump.py | preprocess.py ● | test.html | training.py | background.js | test.js

frontend > js > background.js > ...

```javascript
var results = {};
var legitimatePercents = {};
var isPhish = {};

function fetchLive(callback) {
  fetch("C:/Users/navya/OneDrive/Desktop/phishing-detection-plugin-master/static/classifier.json", {
  method: 'GET'
  })
  .then(function(response) {
    if (!response.ok) { throw response }
    return response.json();
  })
  .then(function(data) {
    chrome.storage.local.set({cache: data, cacheTime: Date.now()}, function() {
      callback(data);
    });
  });
}
```

dump.py | preprocess.py | test.html | training.py | background.js | test.js

frontend > js > background.js > ...

```
var results = {};
var legitimatePercents = {};
var isPhish = {};

function fetchLive(callback) {
  fetch("C:/Users/navya/OneDrive/Desktop/phishing-detection-plugin-master/static/classifier.json", {
  method: 'GET'
  })
  .then(function(response) {
    if (!response.ok) { throw response }
    return response.json();
  })
  .then(function(data) {
    chrome.storage.local.set({cache: data, cacheTime: Date.now()}, function() {
      callback(data);
    });
  });
}

function fetchCLF(callback) {
  chrome.storage.local.get(['cache', 'cacheTime'], function(items) {
      if (items.cache && items.cacheTime) {
        return callback(items.cache);
      }
      fetchLive(callback);
  });
}

function classify(tabId, result) {
  var legitimateCount = 0;
  var suspiciousCount = 0;
  var phishingCount = 0;

  for(var key in result) {
    if(result[key] == "1") phishingCount++;
    else if(result[key] == "0") suspiciousCount++;
    else legitimateCount++;
```

dump.py | preprocess.py | test.html | training.py | background.js | test.js

frontend > js > background.js > ...

```
var results = {};
var legitimatePercents = {};
var isPhish = {};

function fetchLive(callback) {
  fetch("C:/Users/navya/OneDrive/Desktop/phishing-detection-plugin-master/static/classifier.json", {
  method: 'GET'
  })
  .then(function(response) {
    if (!response.ok) { throw response }
    return response.json();
  })
  .then(function(data) {
    chrome.storage.local.set({cache: data, cacheTime: Date.now()}, function() {
      callback(data);
    });
  });
}
```

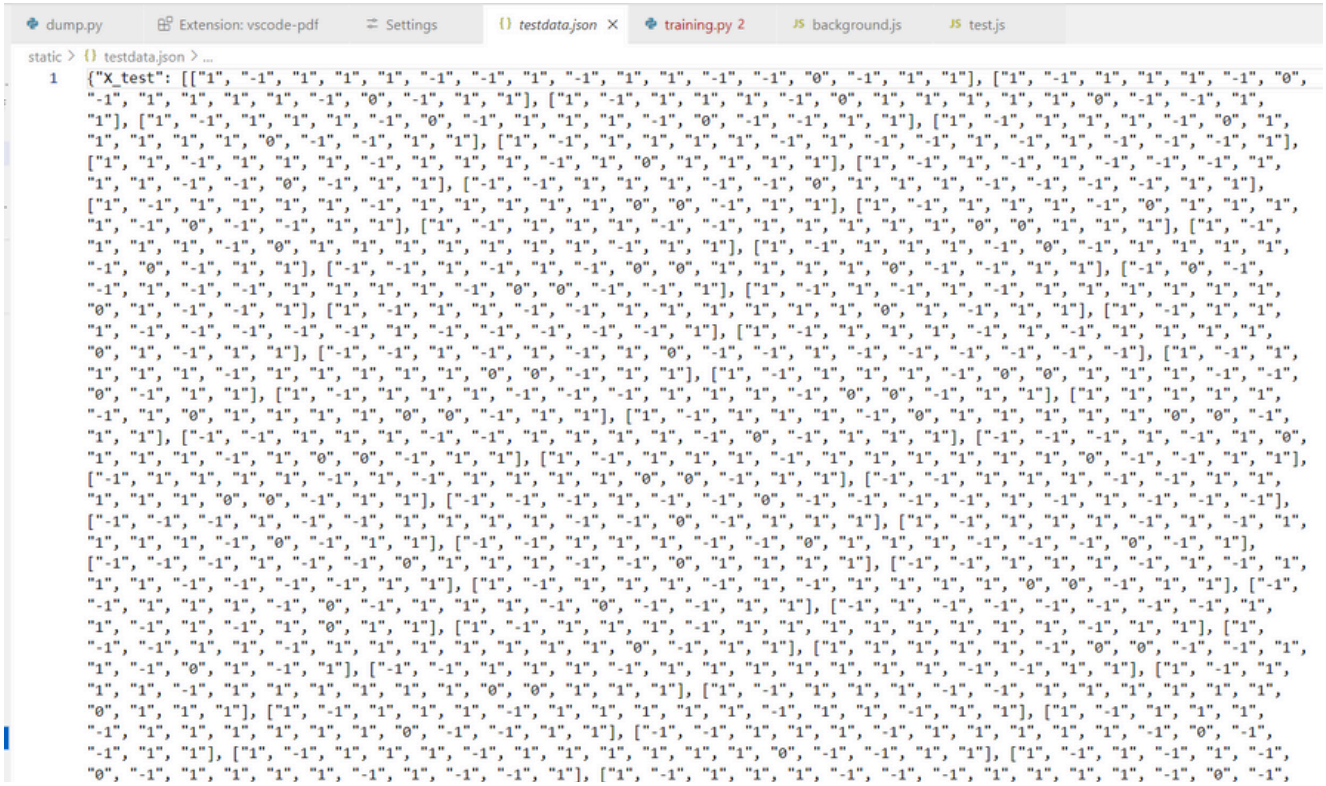
dump.py | preprocess.py ● | training.py | background.js | features.js × | test.js ×

frontend > js > features.js > ...

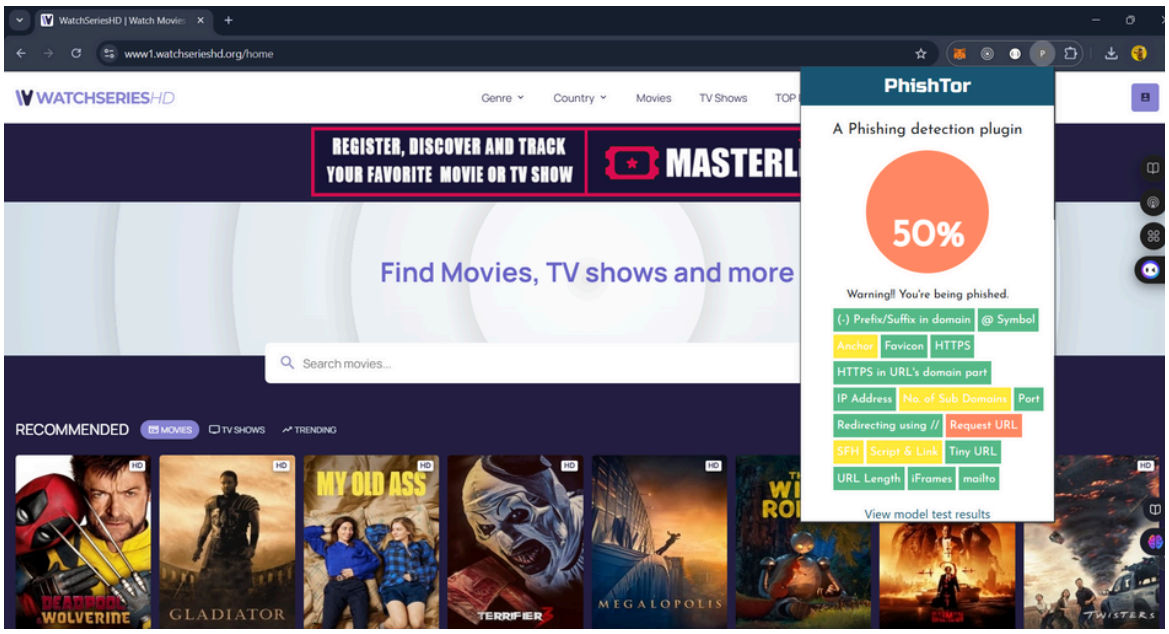
```
1
2   /*
3   $('a').click(function(){
4       alert("You are about to go to "+$(this).attr('href'));
5   });
6   */
7
8   var result = {};
9   //---------------------- 1.  IP Address  ----------------------
10
11  var url = window.location.href;
12  // alert(url);
13  var urlDomain = window.location.hostname;
14
15  //url="0x58.0xCC.0xCA.0x62"
16
17  var patt = /(25[0-5]|2[0-4][0-9]|1[0-9][0-9]|[0-9]?[0-9])(\.|$){4}/;
18  var patt2 = /(0x([0-9][0-9]|[A-F][A-F]|[A-F][0-9]|[0-9][A-F]))(\.|$){4}/;
19  var ip = /\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}/;
20
21
22  if(ip.test(urlDomain)||patt.test(urlDomain)||patt2.test(urlDomain)){
23      result["IP Address"]="1";
24  }else{
25      result["IP Address"]="-1";
26  }
27
28  //alert(result);
29
30  //---------------------- 2.  URL Length  ----------------------
31
32
33  //alert(url.length);
34  if(url.length<54){
35      result["URL Length"]="-1";
36  }else if(url.length>=54&&url.length<=75){
37      result["URL Length"]="0";
```


```
1
2   /*
3   $('a').click(function(){
4       alert("You are about to go to "+$(this).attr('href'));
5   });
6   */
7
8   var result = {};
9   //---------------------- 1.  IP Address  ----------------------
10
11  var url = window.location.href;
12  // alert(url);
13  var urlDomain = window.location.hostname;
14
15  //url="0x58.0xCC.0xCA.0x62"
16
17  var patt = /(25[0-5]|2[0-4][0-9]|1[0-9][0-9]|[0-9]?[0-9])(\.|$){4}/;
18  var patt2 = /(0x([0-9][0-9]|[A-F][A-F]|[A-F][0-9]|[0-9][A-F]))(\.|$){4}/;
19  var ip = /\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}/;
20
21
22  if(ip.test(urlDomain)||patt.test(urlDomain)||patt2.test(urlDomain)){
23      result["IP Address"]="1";
24  }else{
25      result["IP Address"]="-1";
26  }
```

CLASSIFIER DUMP :



FINAL RESULT :



CONCLUSION

In conclusion ,

The "Chrome Extension for Detecting Phishing Websites" project highlights the tremendous potential of machine learning in revolutionizing the way phishing threats are identified and mitigated. Throughout this research, various machine learning algorithms were explored, showcasing their capability to accurately distinguish between phishing and legitimate websites. By implementing a lightweight client-side solution, this project ensures user privacy while maintaining high detection accuracy.

By leveraging historical data of phishing websites, feature engineering, and robust model training, the predictive model incorporated into the extension provides real-time analysis and alerts. This empowers users to browse safely, ensuring they are warned of any potential threats before engaging with harmful sites. Such a tool is not only beneficial for individuals but also for organizations aiming to improve their security infrastructure.

However, it's important to recognize that phishing detection is a continuously evolving challenge due to the dynamic nature of cyber threats. Phishing techniques are becoming more sophisticated, and attackers constantly find new ways to mimic legitimate websites. Therefore, continuous improvements in the classifier model, along with the integration of new datasets and threat intelligence, are essential for maintaining high detection accuracy.

As machine learning techniques advance and more comprehensive phishing datasets become available, we anticipate even more accurate and sophisticated detection models. These models can evolve to not only identify existing phishing threats but also predict and recognize novel attack vectors in real-time, helping to preempt future attacks.

Ultimately, this research and project serve as a stepping stone towards a more secure internet environment. A proper implementation of this phishing detection extension can greatly reduce the risk of users falling prey to phishing attacks. By providing real-time detection and data-driven insights, the extension can save users from compromising sensitive information, such as login credentials and financial details, making the internet a safer place for all.

The challenge of phishing detection involves analyzing various factors such as URLs, website content, and email headers. In this study, we propose a set of features to represent these elements and apply them to state-of-the-art machine learning models for real-time classification. Our goal was to compare the performance of these models and assess the impact of different feature sets on detection accuracy.

The dataset used for the experiments includes information like domain name patterns, URL length, SSL certificate details, and other phishing-specific attributes. Each feature has a defined format and is categorized accordingly to ensure effective model training. Additionally, the dataset provides a wide range of phishing and legitimate websites, allowing for a comprehensive evaluation of the model's effectiveness.

To assess the performance of the machine learning models, we evaluate their detection accuracy and investigate how the accuracy varies based on the feature set used to represent phishing and legitimate websites. This analysis gives us insights into which features are more critical in detecting phishing threats and helps us refine the model for better results.

The experimental results and comparative analysis provide insights into the effectiveness of different machine learning models for phishing website detection. Moreover, by studying the dependency of accuracy on the feature set, we better understand which factors are most influential in phishing detection.
This research contributes to the cybersecurity domain by exploring the performance of multiple machine learning models and analyzing the impact of various features on detection accuracy. The findings are valuable for developers, organizations, and users in making informed decisions about online security and protecting sensitive information from phishing attacks.