IOMP Project Report on

Web Phishing Detection

SUBMITTED BY

N.AMULYA	**19UK1A0512**
V.RAHUL	**19UK1A0511**
M.MAHESH	**19UK1A0502**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

(Affiliated to JNTUH, Hyderabad)

Bollikunta, Warangal – 506005

2019– 2023

CONTENTS

CHAPTER 1

INTRODUCTION

Manually There are a number of users who purchase products online and make payments through e-banking. There are e-banking websites that ask users to provide sensitive data such as username, password & credit card details, etc., often for malicious reasons. This type of e-banking website is known as a phishing website. Web service is one of the key communications software services for the Internet. Web phishing is one of many security threats to web services on the Internet.

Common threats of web phishing:

- Web phishing aims to steal private information, such as usernames, passwords, and credit card details, by way of impersonating a legitimate entity.
- It will lead to information disclosure and property damage.
- Large organizations may get trapped in different kinds of scams.

1.1 OVERVIEW

This Guided Project mainly focuses on applying a machine-learning algorithm to detect Phishing websites.

In order to detect and predict e-banking phishing websites, we proposed an intelligent, flexible and effective system that is based on using classification algorithms. We implemented classification algorithms and techniques to extract the phishing datasets criteria to classify their legitimacy. The e-banking phishing website can be detected based on some important characteristics like URL and domain identity, and security and encryption criteria in the final phishing detection rate. Once a user makes a transaction online when he makes payment through an e-banking website our system will use a data mining algorithm to detect whether the e-banking website is a phishing website or not.

1.2 PURPOSE

By the end of this project:

- You'll be able to understand the problem to classify if it is a regression or a classification kind of problem.

- You will be able to know how to pre-process/clean the data using different data pre-processing techniques.
- Applying different algorithms according to the dataset
- You will be able to know how to find the accuracy of the model.
- You will be able to build web applications using the Flask framework.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

To complete this project, you must require the following software, concepts, andpackages.

1.Anaconda navigator and pycharm.

Anaconda Navigator is a free and open-source distribution of the Pythonand R programming languages for data science and machine learning related applications. It can be installed on Windows, Linux, and macOS. Conda is an open-source, cross-platform, package management system. Anaconda comes with tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code.

2.2 PURPOSED SOLUTION

The method or solution is Jupiter notebook and spyder we used to complete this project. and youwill use this jupiter notebook for you recommended.

To build Machine learning models you must require the following packages

Sklearn: Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms.

NumPy: NumPy is a Python package that stands for 'Numerical Python'. It is the core library for scientific computing, which contains a powerful n-dimensional array object

Pandas: pandas is a fast, powerful, flexible, and easy to use open source data analysis and manipulation tool,built on top of the Python programming language.

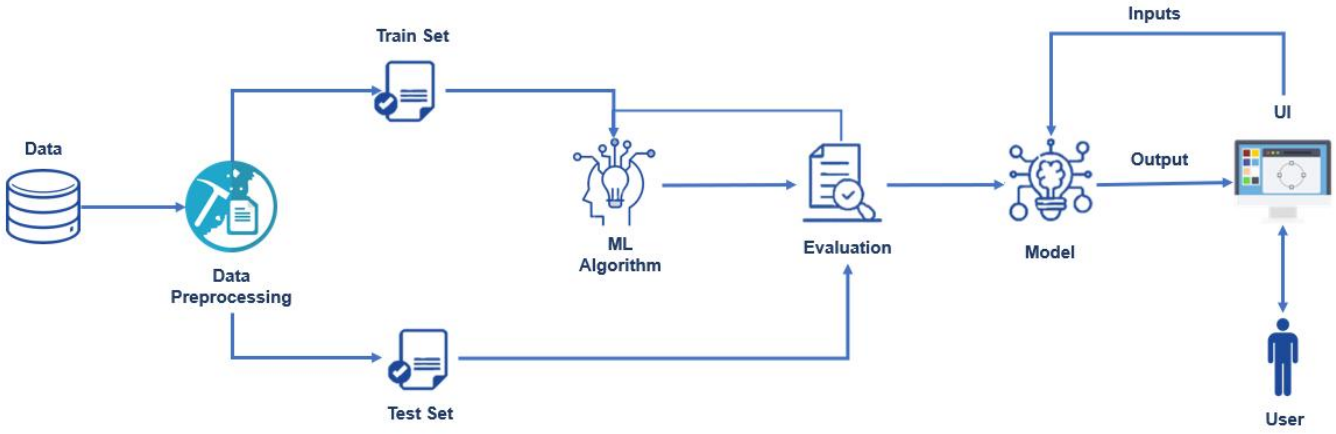
Matplotlib: It provides an object-oriented API for embedding plots into applications using general purpose GUI toolkits

Flask: Web framework used for building Web applications.

CHAPTER 3

THEORETICAL ANALYSIS

3.1 BLOCK DIAGRAM



3.2 HARDWARE / SOFTWARE DESIGNING

The hardware required for the development of this project is:

Processor : Intel CoreTM i5-9300H
Processor speed : 2.4GHz
RAM Size : 8 GB DDR
System Type : X64-based processor

SOFTWARE DESIGNING:

The software required for the development of

this project is:Desktop GUI : Anaconda

Navigator
Operating system : Windows 10
Front end : HTML, CSS,JAVASCRIPT
Programming : PYTHON
Cloud Computing Service : IBM Cloud Services

CHAPTER 4

EXPERIMENTAL INVESTIGATION

IMPORT REQUIRED LIBRARIES:

Import the necessary libraries .

Step 1 - Launch Jupyter notebook through anaconda navigator or anaconda prompt.

Step 2 - Create a new notebook by clicking on "new" button on the top right corner of the page.

The libraries can be imported using the import keyword.

READ DATASET:

The dataset is read as a **dataframe** by using pandas library. Insert the commands as shown below (Here **ds** is referred as dataframe & **pd** is the alias name given to pandas library).

CHAPTER 5

FLOWCHART

PROJECT FLOW:

Find below the project flow to be followed while developing the project.

1. Download the dataset.
2. Preprocess or clean the data.
3. Analyze the pre-processed data.

   Data Pre-Processing

   In this milestone, we will be pre-processing the dataset that is collected.

   Pre-processing includes

   A) Handling the null values.

   B) Handling the categorical values if any.

   C) Normalize the data if required.

   D) Identifying the dependent and independent variables.

   E) Split the dataset into train and test sets.
4. Train the machine with preprocessed data using an appropriate machine learning algorithm.
5. Save the model and its dependencies.
6. Build a Web application using a flask that integrates with the model built.

3. Analyze the pre-processed data.

- **Handling Null Values**

Checking for Null values in a dataset and handling if any

In this activity, we will check if there are any null values in a dataset and fill/handle them.

To know if there are any null values present in a dataset isnull() method can be used.

- **Splitting The Data**

Splitting data into independent and dependent variables

Identifying Independent & dependent variables:

In this activity, the dependent and independent variables are to be identified. The last column (Result) in the dataset is the dependent variable which is dependent on the 30 different factors. The independent columns are considered as x and the dependent column as y.

- **Splitting the data:**

After identifying the dependent and independent variables, the dataset now has to be split into two sets, one set is used for training the model and the second set is used for testing how good the model is built. The split ratio we consider is 80% for training and 20% for testing.

- **Model Building**

a. Choose The Appropriate Model

Start building Machine Learning Model

There are several Machine Learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms can be chosen according to the objective. As the dataset which we are using is a Classification dataset we can use the following algorithms,

A. Logistic Regression
B. Random Forest Regression / Classification
C. Decision Tree Regression / Classification
D. K-Nearest Neighbors
E. Support Vector Machine

In order to get appropriate predictions, the dataset can be trained with any of the above algorithms.

a. Choose The Appropriate Model

Working with Logistic Regression model

Step - 1 Here, We will be initially considering Logistic Regression model and fit the data.

Step - 2 Check the metrics of the model

Here we will be evaluating the model built. We use the test set for evaluation. The test set is given to the model for prediction and prediction values are stored in another variable called y_pred1.

The actual and predicted values are compared to know the accuracy of the model using the accuracy_score function from sklearn.metrics package.

The accuracy for logistic regression model for this dataset is 91.6%.

Note: You can use different classification models to know the performance and choose whichever works better.

Step - 3 Saving the model

The finalized model is now to be saved. We will be saving the model as a pickle or pkl file.

Use the command below to save the model.

- **APPLICATION BUILDING:**

1. Flask App (Step - 1)
2. Flask App (Step - 2)
3. Build An HTML Page
4. Execute And Test Your Model
5. Testing The Model
6. The Final Step

Building an Application to integrate the model

After the model is built, we will be integrating it to a web application so that normal users can also use it to know if any website is phishing or safe in a no-code manner.

In the application, the user provides any website URL to check and the corresponding parameter values are generated by analysing the URL using which legitimate websites are detected.

1. Flask App (Step - 1)

Build the python flask app

In the flask application, the URL is taken from the HTML page and it is scraped to get the different factors or the behavior of the URL. These factors are then given to the model to know if the URL is phishing or safe and is sent back to the HTML page to notify the user.

2. Flask App (Step - 2)

Configure app.py to fetch the URL from the UI, process the URL, get the input parameters from the URL and return the prediction.

3. Build An HTML Page

We Build an HTML page to take the URL as a text and upon clicking on the button for submission it has to redirect to the URL for "y_predict" which returns if the URL given is phishing or safe. The output is to be then displayed on the page. The HTML pages are put under the templates folder and any style sheets if present is kept in the static folder.

4. Execute And Test Your Model

Now we execute the model using Anaconda Prompt

Execute the python code by giving the command python app.py in anaconda prompt

5.Testing The Model

About the project section which gives insights about the project.

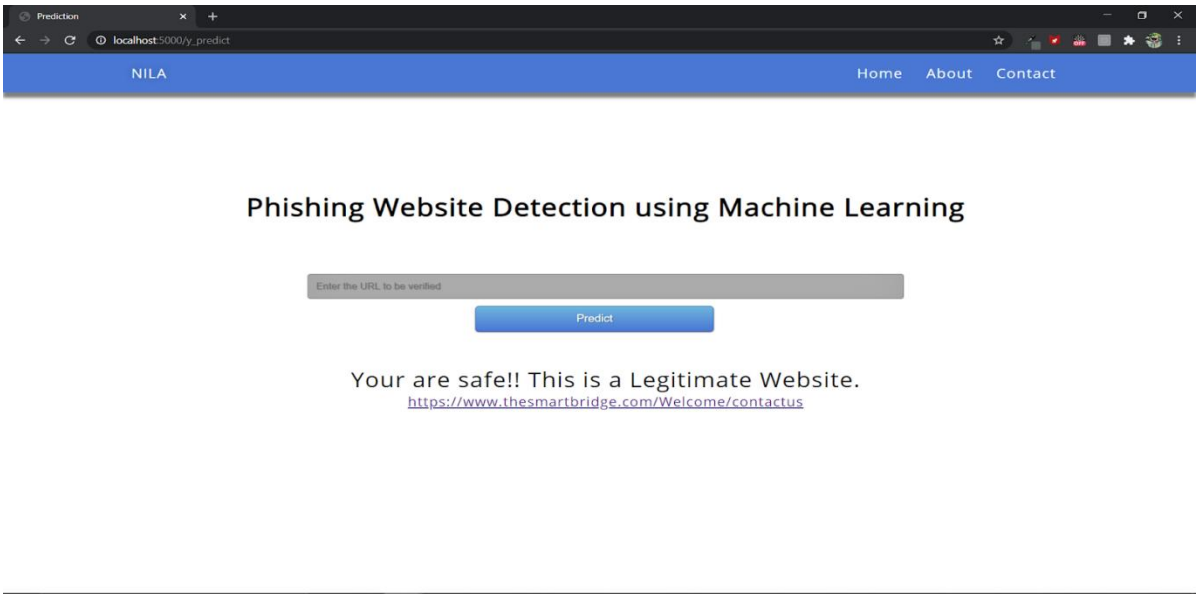
6.The Final Step

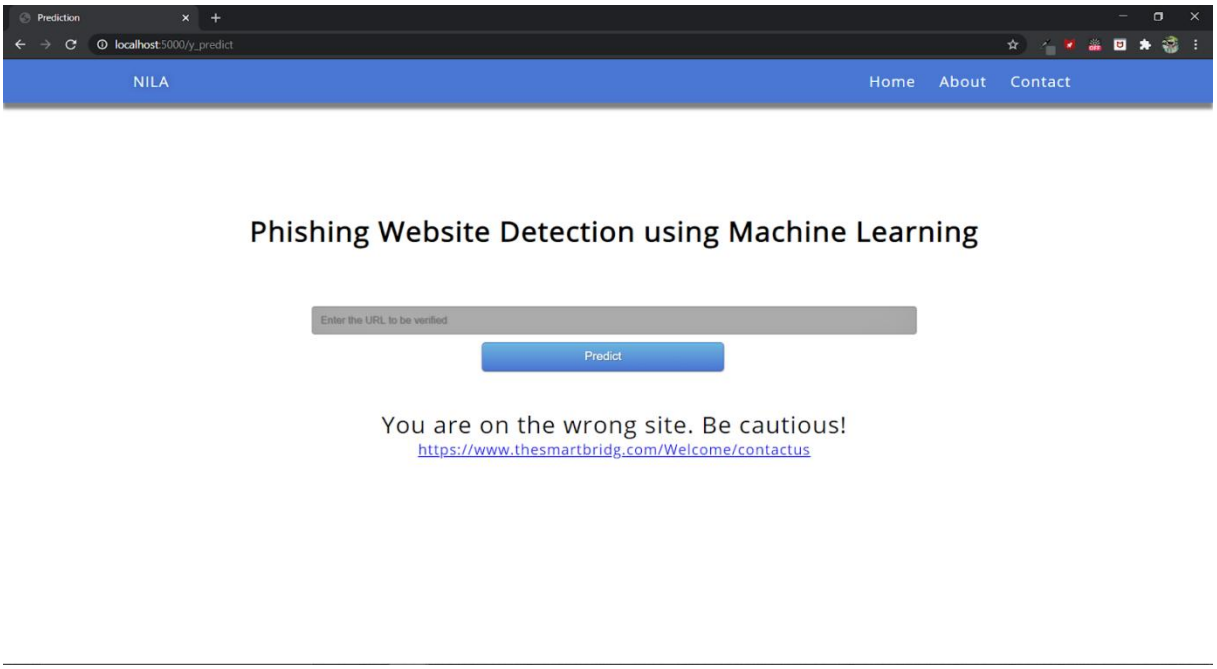
When the URL is given, the model analyses and gives the output whether it is a phishing or legitimate website.

Here, we will try to specify the same link given above by altering the spelling of the domain name. It validates with the domain name and if not found, It warns about the risk of phishing.

CHAPTER 6

RESULT





CHAPTER 7

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- This system can be used by many E-commerce or other websites in order to have good customer relationship.
- User can make online payment securely.
- Data mining algorithm used in this system provides better performance as compared to other traditional classifications algorithms.
- With the help of this system user can also purchase products online without any hesitation.

DISADVANTAGES

- If Internet connection fails, this system won't work.
- All websites related data will be stored in one place.

CHAPTER 8

APPLICATIONS

phishing detection by blacklists, heuristics, visual similarity, and data mining solutions and found that the solutions based on ML techniques are most promising. They found that detection at hour zero and low false-positive rate are critical measures for phishing solutions.

CHAPTER 9

CONCLUSION AND FUTURESCOPE

CONCLUSION

Thus to summarize, we have seen how phishing is a huge threat to the security and safety of the web and how phishing detection is an important problem domain. We have reviewed some of the traditional approaches to phishing detection; namely blacklist and heuristic evaluation methods, and their drawbacks. We have tested two machine learning algorithms on the Phishing Websites Dataset and reviewed their results. We then selected the best algorithm based on its performance and built a Chrome extension for detecting phishing web pages. The extension allows easy deployment of our phishing detection model to end users. We have detected phishing websites using Random Forest algorithm with and accuracy of 97.31%. For future enhancements, we intend to build the phishing detection system as a scalable web service which will incorporate online learning so that new phishing attack patterns can easily be learned and

FUTURESCOPE

Although the use of URL lexical features alone has been shown to result in high accuracy (97%), phishers have learned how to make predicting a URL destination difficult by carefully manipulating the URL to evade detection. Therefore, combining these features with others, such as host, is the most effective approach .

For future enhancements, we intend to build the phishing detection system as a scalable web service which will incorporate online learning so that new phishing attack patterns can easily be learned and improve the accuracy of our models with better feature extraction.

CHAPTER 10

BIBILOGRAPHY

- M. Karabatak and T. Mustafa, Performance comparison of classifiers on reduced phishing website dataset, 6th Int. Symp. Digit. Forensic Secur. ISDFS 2018 – Proceeding, vol. 2018Janua, pp. 15, 2018.

- S. Parekh, D. Parikh, S. Kotak, and P. S. Sankhe, A New Method for Detection of Phishing Websites: URL Detection, in 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, vol. 0, no. Icicct, pp. 949952.

- K. Shima et al., Classification of URL bitstreams using bag of bytes, in 2018 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), 2018, vol. 91, pp. 15.

- W. Fadheel, M. Abusharkh, and I. Abdel-Qader, On Feature Selection for the Prediction of Phishing Websites, 2017 IEEE 15th Intl Conf Dependable, Auton. Secur. Comput. 15th Intl Conf Pervasive Intell. Comput. 3rd Intl Conf Big Data Intell. Comput. Cyber Sci. Technol. Congr., pp. 871876, 2017.

- X. Zhang, Y. Zeng, X. Jin, Z. Yan, and G. Geng, Boosting the Phishing Detection Performance by Semantic Analysis, 2017.

APPENDIX

A Source Code of Flask:

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix,accuracy_score
```

```
#Import Dataset
ds= pd.read_csv("dataset_website.csv")
ds.head()
```

	index	having_IPhaving_IP_Address	URLURL_Length	Shortining_Service	having_At_Symbol	double_slash_redirecting	Prefix_Suffix	having_Sub_Domair
0	1	-1	1	1	1	-1	-1	-1
1	2	1	1	1	1	1	-1	0
2	3	1	0	1	1	1	-1	-1
3	4	1	0	1	1	1	-1	-1
4	5	1	0	-1	1	1	-1	1

5 rows × 32 columns

```
#Analysing the data using pandas and Checking if the dataset contains any Null values.
ds.info()
ds.isnull().any() #no nullvalues
```

```
#Splitting data as independent and dependent
#removing index column in independent dataset
x=ds.iloc[:,1:31].values
y=ds.iloc[:,-1].values
print(x,y)
```

```
#Splitting data into train and test
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

Working with Logistic Regression model

Step-1

```
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
lr.fit(x_train,y_train)
```

Step-2

```
y_pred1=lr.predict(x_test)
from sklearn.metrics import accuracy_score
log_reg=accuracy_score(y_test,y_pred1)
log_reg
```

```
0.9167797376752601
```

Step-3

```
import pickle
pickle.dump(lr,open('Phishing_Website.pkl','wb'))
```

Flask App (Step - 1)

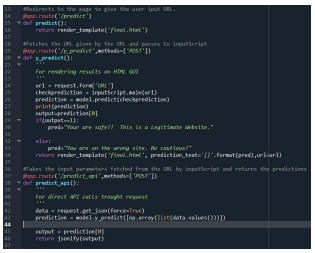
Input the following commands to Import required libraries

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle
#importing the inputScript file used to analyze the URL
import inputScript
```

Load the model and initialize Flask App

```
#load model
app = Flask(__name__)
model = pickle.load(open('Phishing_Website.pkl', 'rb'))

```

Flask App (Step - 2)



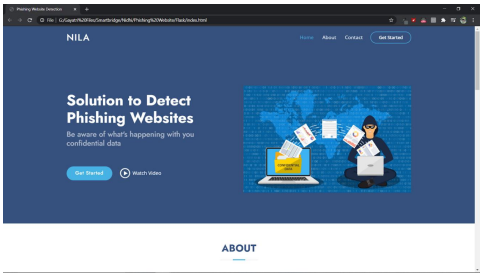
Run the app

```

if __name__ == '__main__':
    app.run(host='0.0.0.0', debug=True)

```

Now we execute the model using Anaconda Prompt

```
(base) G:\Gayatri Files\Smartbridge\Nidhi\Phishing Website\Flask>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with windowsapi reloader
 * Debugger is active!
 * Debugger PIN: 715-830-168
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

This is the home page of the web application(index.html)



Testing The Model