PHISHING WEBSITE DETECTION

A Mini-Project Report submitted in partial fulfilment of the requirements for the award of the degree of,

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Mahendra Reddy G	**321910307015**
Harsha Vardhan Reddy B	**321910307004**
Jagadeesh A	**321910307039**
Balu Reddy K	**321910307059**

Under the esteemed guidance of

Kavya G

Assistant Professor

Department of Computer Science & Engineering,

GITAM SCHOOL OF TECHNOLOGY

GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Deemed to be University)

Bengaluru Campus

November 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GITAM SCHOOL OF TECHNOLOGY
GITAM
(Deemed to be University)

DECLARATION

We, hereby declare that the project report entitled "Phishing Website Detection "is an original work done in the Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University) Bengaluru submitted in partial fulfilment of the requirements for the award of the degree of B.Tech. in Computer Science and Engineering. The work has not been submitted to any other college or University for the award of any degree.

Date: 26/10/22

Registration No(s).	Name(s)	Signature(s)
321910307015	**Mahendra Reddy G**	
321910307004	**Harsha Vardhan Reddy B**	
321910307039	**Jagadeesh A**	
321910307059	**Balu Reddy K**	

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GITAM SCHOOL OF TECHNOLOGY

GITAM

(Deemed to be University)

CERTIFICATE

This is to certify that the project report entitled **“Phishing Website Detection”** is a Bonafede record of work carried out by **Mahendra Reddy. G (321910307015), Harsha Vardhan Reddy. B (321910307004), Jagadeesh. A (321910307039), Balu Reddy K (321910307059)** submitted in partial fulfillment of requirement for the award of degree of **Bachelors of Technology in Computer Science and Engineering**.

Project Guide	**Head of the Department**
Signature of guide	Signature of HOD
Kavya G Assistant Professor	**Dr. Vamsidhar Yendapalli** Professor

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose consistent guidance and encouragement crowned our efforts with success. We consider it our privilege to express our gratitude to all those who guided us in the completion of the project. We express our gratitude to Director Prof. T. Nageshwara Rao for having provided us with the golden opportunity to undertake this project work in their esteemed organization.

We sincerely thank Dr. Y. Vamshidhar, HOD, Department of Computer Science and Engineering, Gandhi Institute of Technology and Management, Bengaluru for the immense support given to us.

We express our gratitude to our project guide Kavya G, Assistant Professor, Department of Computer Science and Engineering, Gandhi Institute of Technology and Management Bengaluru, for their support, guidance, and suggestions throughout the project work.

Student Names	**Registered Number**
Mahendra Reddy G	321910307015
Harsha Vardhan Reddy B	321910307004
Jagadeesh A	321910307039
Balu Reddy K	321910307059

ABSTRACT

Phishing is a common attack on credulous people by making them to disclose their unique information using counterfeit websites The objective of phishing website URLs is to purloin the personal information like user name, passwords and online banking transactions. Phishers use the websites which are visually and semantically like those real websites. Phishing is popular among attackers, since it is easier to trick someone into clicking a malicious link which seems legitimate than trying to break through a computers defense system. The malicious links within the body of the message are designed to make it appear that they go to the spoofed organization using that organizations logos and other bad contents As technology continues to grow, phishing techniques started to progress rapidly and this needs to be prevented by using anti-phishing mechanisms to detect phishing. Machine learning is a powerful tool used to strive against phishing attacks. Here we explain phishing domain (or Fraudulent Domain) characteristics, the features that distinguish them from legitimate domains, why it is important to detect these domains, and how they can be detected using machine learning techniques. In this project we will create a model where the we will enter the URL in our List and the algorithm will tell whether the URL is good or bad. When a new URL is entered, it checks the list and detects whether it is a bad website or not. In addition, the model learns about the data set of the URLs that are enter.

Key Words:

Machine learning, Malicious URL, Logistic algorithm, Phisher, Malware.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.INTRODUCTION

Nowadays Phishing becomes a main area of concern for security researchers because it is not difficult to create the fake website which looks so close to good website. Experts can identify fake websites but not all the users can identify the fake website and such users become the victim of phishing attack. Main aim of the attacker is to steal banks account credentials. Phishing attacks are becoming successful because lack of user awareness. Since phishing attack exploits the weaknesses found in users, it is very difficult to mitigate them but it is very important to enhance phishing detection techniques. Phishing may be a style of broad extortion that happens once a pernicious web site act sort of a real one memory that the last word objective to accumulate unstable info, as an example, passwords, account focal points, or MasterCard numbers. all the same, the means that their square measure some of contrary to phishing programming and techniques for recognizing potential phishing tries in messages and characteristic phishing substance on locales, phishes think about new and crossbreed procedures to bypass the open programming and frameworks.

Phishing may be a fraud framework that uses a mixture of social designing what is additional, advancement to sensitive and personal data, as an example, passwords associate degree open-end credit unpretentious elements by presumptuous the highlights of a reliable individual or business in electronic correspondence. Phishing makes use of parody messages that square measure created to seem substantial and instructed to start out from true blue sources like money connected institutions, online business goals, etc, to draw in customers to go to phony destinations through joins gave within the phishing websites.

1.1 Problem Definition, significance, and objective

Problem Definition:

Our main objective is to detect fraudulent websites and reduce the cybercrime and data extraction by these sites. Under this project, using machine learning, we will train our model to the data set containing the features of the malicious websites. Once the system is trained using the machine learning and Logistic regression Algorithm, we will be able to analyse any URL and detect if it is an authentic website or not. Dataset cleaning involves removing noise from our dataset. Noise is unnecessary characters in the text data, punctuations, and repetitive words. Removing noise from our dataset will enable the model to focus only on the most important information in the dataset. This will increase the model performance. The model will be able to make accurate predictions. In this tutorial, we will first split the texts, then remove repetitions in the dataset. Finally, we will remove the com from each URL. At this point, we can take the input from the users, in the form of URL and analyse through the different features of that website. We will then test this URL and display whether the URL is Good or Bad.

Significance:

This project report is to provide comprehensive information about the technical aspects and real-time of the project - Phishing Website Detection using a machine learning technique.

Objective:

The objective of this project is to detect fake URL links using a machine learning algorithm and reduce cyber attacks

1.2 Methodologies

We will take the sample dataset which contains some malicious URL's and some non-malicious URL's. From the dataset and the use of machine learning algorithm the program can predict that the entered URL or website is malicious or not. The first thing to do after the getting of datasets is data slicing. Here, the datasets are divided into two parts. They are as follows: testing and training dataset. The training datasets are used to train a model. The testing dataset issued once the trained model is ready (for testing purpose). Once the model is trained, we test its accuracy level on the testing datasets. While training the model on the training datasets, we find its accuracy by doing repeated cross-validation on the datasets. This also permit us to do tuning of the parameters in the two datasets to find out which parameter gives the greater accuracy for the datasets. Once, the best suitable parameter for the model is get, the training of the model is done, and then we can go to do testing of the trained model on the testing datasets. We are going to do the classification task on the datasets. We make use of Logistic algorithm

1.3 Outline of the project

At the end of the project, our code will reduce these things.

Reduce Cyber attacks

Public knowing about this attack

New techniques learn victims

Data losing reduce

1.4 Scope of the Project

This project is intended to work on various devices such as mobile PC / Laptops. Because of data set will be stored device but the path we must give correctly then only it will work.

The objectives of the project will be determined. Then which language we used and how it detects how converts string to binary form.

Improving detecting URLs correctly high accuracy

1.5 Organization Of The Report

This report will clearly explain the features, algorithm used, proposed implementations, results of the implementations, and conclusions drawn from the results. After this introduction section, section 2, the literature survey, will introduce the problem in detail. It tells accuracy, techniques, and discuss how they will be used. We will then go over the existing solutions that have already been presented along withany other related works.

Sect ion 3, We will investigate the details of how exactly we implement and how this technique is run and implemented differently from the existing solutions. implementation architectures, module descriptions.

Next, section 4 will show the implementation of the proposed system. This section will depict the implementation of the execution through architecture. The algorithms will also be presented in this section. data sets would also be presented and described in this section.

section 5, We will look at how the code run at each step and displaying all the outputs and results obtained from the training and execution of our proposed model.

Finally in section 6, we will conclude, based on our implementation and results, whether the execution successful. Finally, after observing all the results and drawing conclusions, we present our recommendations on how to use this code, who should use this algorithm, what scenarios is it most effective in, how we can improve it, and so on. The future works and scopes of this project will also be discussed in this section.

2. LITERATURE SURVEY

[1] (J. Shad and S. Sharma) 2018 In emerging technology, industry, which deeply influence today's security problems, has given a headache to many employers and home users. Occurrences that exploit human vulnerabilities have been on the upsurge in recent years. In these new times there are many security systems being enabled to ensure security is given the outmost priority and prevention to be taken from being hacked by those who are involved in cyber-offenses and essential prevention is taken as high importance in organization to ensure network security is not being compromised. Cyber security employee is currently searching for trustworthy and steady detection techniques for phishing websites detection. Due to wide usage of internet to perform various activities such as online bill payment, banking transaction, online shopping, etc. Customer face numerous security threats like cybercrime. Many cybercrimes are being casually executed for example spam, fraud, identity theft cyber terrorisms and phishing. Among this phishing is known as the most common cybercrime today. Phishing has become one amongst the top three most current methods of law breaking in line with recent reports, and both frequency of events and user weakness has increased in recent years, more combination of all these methods result in greater danger of economic damage.

[2] (Atharva Deshpande, Omkar Pedamkar, Nachiket Chaudhary) 2021 Phishing is a social engineering attack that targets and exploiting the weakness found in the system at the user's end. This paper proposes the Agile Unified Process (AUP) to detect duplicate websites that can potentially collect sensitive information about the user. The system checks the blacklisted sites in dataset and learns the patterns followed by the phishing websites and applies it to further given inputs. The system sends a pop-up and an e-mail notification to the user, if the user clicks on a phishing link and redirects to the site if it is a safe website. This system does not support real time detection of phishing sites; user has to supply the website link to the system developed with Microsoft Visual Studio 2010 Ultimate and MySQL stocks up data and to implement database in this system.

[3] (X. Zhang, Y. Zeng, X. Jin, Z. Yan,) 2017 Phishing costs Internet user's lots of money. It refers to misusing weakness on the user side, which is vulnerable to such attacks. The basic ideology of the proposed solution is use to all the three-hybrid solution blacklist and whitelist, heuristics and visual similarity. The proposed system carries out a set of procedures before giving out the results. First, it tracks all "http" traffic of client system by creating a browser extension. Then compare domain of each URL with the white list of trusted domains and the blacklist of illegitimate domains. Further various characters in the URL is considered like number of '@', number of '-'and many more. Next approach is to extract and compare CSS of doubtful URL and compare it with the CSS of each of the legitimate domains in queue. This method will look into visual based features of the phished websites and machine-learning classifiers such as decision tree, logistic regression, random forest are applied to the collected data, and a score is generated. The match score and similarity score is evaluated. If the score is greater than threshold then the URL marked as phishing and blocked. This approach provides a three-level security block.

[4] (L. MacHado and J. Gadge) 2018 Phishing is a dangerous effort to steal private data from users like address, Aadhar number, PAN card details, credit or debit card details, bank account details, personal details etc. The various types of phishing attacks like spoofing, instant spam spoofing, Hosts file poisoning, malware-based phishing, Man-in-the middle, session hijacking, DNS based phishing, deceptive phishing, key loggers/loggers, Web Trojans, Data theft, Content-injection phishing, Search engine phishing, Email /Spam, Web based delivery, Link Manipulation, System reconfiguration, Phone phishing, etc. are discussed in the paper. The recent approaches to prevent the attacks like heuristics approach, blacklist approach, fuzzy rule-based approach, machine learning approach etc. are also discussed and finally filtering all detection techniques based on accuracy and performance proposed a framework to detect and prevent phishing attacks. A combination of supervised and unsupervised machine learning techniques is used to detect malicious attacks.

2.1 Introduction to the problem

Our main objective is to detect fraudulent websites and reduce the cybercrime and data extraction by these sites. Under this project, using machine learning, we will train our model to the data set containing the features of the malicious websites. From the dataset and the use of machine learning algorithm the program can predict that the entered URL or website is malicious or not. The first thing to do after the getting of datasets is data slicing. Here, the datasets are divided into two parts. They are as follows: testing and training dataset. The training datasets are used to train a model. The testing dataset issued once the trained model is ready. Phishing may be a fraud framework that uses a mixture of social designing what is additional, advancement to sensitive and personal data, as an example, passwords associate degree open-end credit unpretentious elements by presumptuous the highlights of a reliable individual or business in electronic correspondence. Once the system is trained using the machine learning and Logistic regression Algorithm, we will be able to analyse any URL and detect if it is an authentic website or not. At this point, we can take the input from the users, in the form of URL and analyse through the different features of that website. We will then test this URL and display whether the URL is Good or Bad.

2.2 Existing solutions for Detection

In The Existing system already proposed algorithms accuracy like this SVM -80.1% and Decision Tree, Random Forest combined – 95.01%, Random Forest – 94.11%

Disadvantages

Time Complexity is high, Less Accuracy in The Existing algorithms

2.3 Related works

Phishing as a term did not exist until 1996 when it was first mentioned by 2600 a popular hacker newsletter after an attack on AOL (Ollmann, 2004). Since then, there has been an exponential increase in phishing attacks, with it becoming one of the most prevalent

methods of cybercrime. According to Verizon (2019), phishing played a part in 78% of all Cyber-Espionage incidents and 87% of all installations of C2 malware in the first quarter of 2019 (Verizon, 2019). In the earlier report by Widup et al. (2018), it is reported that 78% of people didn't click a single phish all year, meaning that 22% of people did click one. Moreover, only 17% of these phishing campaigns were reported by users. It is also emphasized that even though training can reduce the number of incidents, phish happens (Widup)et al., 2018). Since only a single e-mail is needed to compromise an entire organization, protection against it should be taken seriously. Cyber-criminals use phishing attacks to either harvest information or steal money from their victims through deceiving them with a reflection of what would seem like a regular e-mail or website. By redirecting the victim to their disguised website, they can see everything the victim inserts in any forms, login pages or payment sites. Cyber-criminals either copy the techniques used by digital marketing experts or take advantage of the fuss created by viral events to guarantee a high click rate. Vergelis and Shcherbakova (2019), reported a spike in phishing redirects to Apple's sites before each new product announcement (Vergelis and Shcherbakova, 2019). Regular phishing attacks are usually deployed widely and are very generic, such that they can be deployed to target as many people as possible. A Spear Phishing attack, instead, targets a specific individual, but requires that information be gathered about the victim prior to crafting a successful spear-phishing email. A more advanced version of this attack is a Whaling attack, which specifically targets a company's senior executives to obtain higher-level access to the organization's system. Targeted phishing attacks are

increasingly gaining popularity because of their high success rates (Krebs, 2018)

2.4 Technologies used

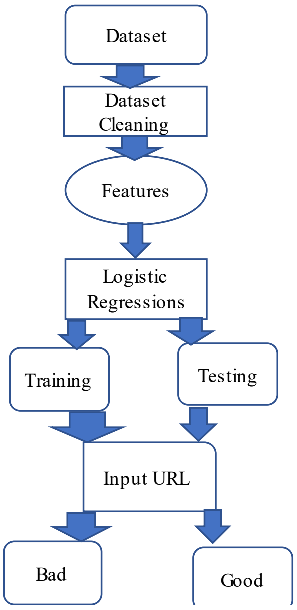
Machine Learning – skit learn is a python library. Used for implementation algorithm. We used machine learning for detecting fake URLs

3 PROBLEM STATEMENT

3.1 Objectives

Our main objective is to detect fraudulent websites and reduce the cybercrime and data extraction by these sites. Under this project, using machine learning, we will train our model to the data set containing the features of the malicious websites. Once the system is trained using the machine learning and Logistic regression Algorithm, we will be in a position to analyse any URL and detect if it is an authentic website or not. At this point, we can take the input from the users, in the form of URL and analyse through the different features of that website. We will then test this URL and display whether the URL is Good or Bad. Dataset cleaning involves removing noise from our dataset. Noise is unnecessary characters in the text data, punctuations, and repetitive words. Removing noise from our dataset will enable the model to focus only on the most important information in the dataset. This will increase the model performance. The model will be able to make accurate predictions. In this tutorial, we will first split the texts, then remove repetitions in the dataset. Finally, we will remove the com from each URL. We will create a custom Python function to clean our dataset. Features are the unique data points in our dataset that are used as input for the model during training. Features are represented by the URL column, which is our input column. In machine learning, a label is the model's output after prediction. It is represented using the label column. The model's output can either be bad or good. To build our machine learning model, we have various Python packages that are essential for this process. We will import all the important packages. Logistic Regression This is a Scikit-learn algorithm that we will use to train our model. This algorithm will enable our model to understand patterns and relationships in our dataset. The model will gain useful knowledge and insight, which it will use to make predictions. train_test_split This is the function in Sklearn model selection for splitting data arrays into two subsets for training data and for testing data. TfidfVectorizer This package will enable the model to understand and manipulate text data. Text is a big problem for machines, machines cannot consume text in its raw form. We need to convert text into vectors of numbers that machines can read and understand. TfidfVectorizer is used to convert the raw text data into vectors of numbers that represents the original text. This text is converted based on the frequency of occurrence of each word in the text data. We chose the model with least time complexity and highest accuracy.

4.1 Block Diagram



10

4.2 Theoretical Foundation/Algorithm

Logistic Regression

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False etc. This is a Logistic Regression algorithm that we will use to train our model. This algorithm will enable our model to understand patterns and relationships in our dataset. The model will gain useful knowledge, which it will use to make predictions. Logistic regression and linear regression are similar and can be used for evaluating the likelihood of class. When the dependent variable is categorical or binary, logistic regression is suitable to be conducted. However, logistic regression is not suitable to predict continuous data such as age, size, etc. Nevertheless, logistic regression signifies the association between one dependent binary variable and one or more independent variables which can be nominal, ordinal, or ratio level variables.

Binomial:

target variable can have only 2 possible types: “0” or “1” which may represent “win” vs “loss”, “pass” vs “fails”, “dead” vs “alive”, etc.

Multinomial:

target variable can have 3 or more possible types which are not ordered (i.e., types have no quantitative significance) like “disease A” vs “disease B” vs “disease C”.

Ordinal:

It deals with target variables with ordered categories. For example, a test score can be categorized as: “very poor”, “poor”, “good”, “very good”. Here, each category can be given a score like 0, 1, 2, 3.

Low Precision/High Recall:

In applications where we want to reduce the number of false negatives without necessarily reducing the number of false positives, we choose a decision value that has a low value of Precision or a high value of Recall. For example, in a cancer diagnosis application, we do not want any affected patient to be classified as not affected without giving much heed to if the patient is being wrongfully diagnosed with cancer. This is because the absence of cancer can be detected by further medical diseases but the presence of the disease cannot be detected in an already rejected candidate.

High Precision/Low Recall:

In applications where we want to reduce the number of false positives without necessarily reducing the number of false negatives, we choose a decision value that has a high value of Precision or a low value of Recall. For example, if we are classifying customers whether they will react positively or negatively to a personalized advertisement, we want to be sure that the customer will react positively to the advertisement because otherwise, a negative reaction can cause a loss of potential sales from the customer.

This is a Scikit-learn algorithm that we will use to train our model. This algorithm will enable our model to understand patterns and relationships in our dataset. The model will gain useful knowledge and insight, which it will use to make predictions.

TF-IDF

Term frequency-inverse document frequency is a text vectorizer that transforms the text into a usable vector. It combines 2 concepts, Term Frequency (TF) and Document Frequency (DF). The term frequency is the number of occurrences of a specific term in a document. Term frequency indicates how important a specific term in a document. Term frequency represents every text from the data as a matrix whose rows are the number of documents and columns are the number of distinct terms throughout all documents. Document frequency is the number of documents containing a specific term. Document frequency indicates how common the term is. Inverse document frequency (IDF) is the weight of a term, it aims to reduce the weight of a term if the term's occurrences are scattered throughout all the documents. IDF can be calculated as follow:

$$idf_i = \log(\frac{n}{df_i})$$

Where idf_i is the IDF score for term i, df_i is the number of documents containing term i, and n is the total number of documents. The higher the DF of a term, the lower the IDF for the term. When the number of DF is equal to n which means that the term appears in all documents, the IDF will be zero, since log (1) is zero, when in doubt just put this term in the stop word list because it doesn't provide much information. The TF-IDF score as the name suggests is just a multiplication of the term frequency matrix with its IDF, it can be calculated as follow:

$$w_{i,j} = tf_{i,j} \times idf_i$$

Where w_{ij} is TF-IDF score for term I in document j, tf_{ij} is term frequency for term I in document j, and idf_i is IDF score for term I.

Data set

diaryofagameaddict.com,bad

espdesign.com.au,bad

https://www.facebook.com/,good

iamagameaddict.com,bad

kalantzis.net,bad

slightlyoffcenter.net,bad

toddscarwash.com,bad

tubemoviez.com,bad

ipl.hk,bad

crackspider.us/toolbar/install.php?pack=exe,bad

pos-kupang.com/,bad

rupor.info,bad

svision-online.de/mgfi/administrator/components/com_babackup/classes/fx29id1.txt,bad

officeon.ch.ma/office.js?google_ad_format=728x90_as,bad

sn-gzzx.com,bad

sunlux.net/company/about.html,bad

outporn.com,bad

timothycopus.aimoo.com,bad

xindalawyer.com,bad

freeserials.spb.ru/key/68703.htm,bad

deletespyware-adware.com,bad

orbowlada.strefa.pl/text396.htm,bad

ruiyangcn.com,bad

zkic.com,bad

adserving.favorit-network.com/eas?camp=19320;cre=mu&grpid=1738&tag_id=618&nums=FGApbjFAAA,bad

cracks.vg/d1.php,bad

juicypussyclips.com,bad

nuptialimages.com,bad

andysgame.com,bad

bezproudoff.cz,bad

ceskarepublika.net,bad

hotspot.cz,bad

gmcjjh.org/DHL,bad

nerez-schodiste-zabradli.com,bad

nordiccountry.cz,bad

nowina.info,bad

obada-konstruktiwa.org,bad

otylkaaotesanek.cz,bad

pb-webdesign.net,bad

pension-helene.cz,bad

podzemi.myotis.info,bad

smrcek.com,bad

spekband.com,bad

m2132.ehgaugysd.net/zyso.cgi?18,bad

webcom-software.ws/links/?153646e8b0a88,bad

worldgymperu.com,bad

zgsysz.com,bad

oknarai.ru,bad

realinnovation.com/css/menu.js,bad

hardcorepornparty.com,bad

zous.szm.sk,bad

noveslovo.com,bad

dimsnetwork.com,bad

luckyblank.info,bad

luckyclean.info,bad

luckyclear.info,bad

luckyeffect.info,bad

975kgkl.com/,good

98.127.224.134/purecountry/,good

987ampradio.radio.com/category/contests/,good

98country.com/tags/scotty-mccreary/,good

991.com/Sell/Wanted.aspx?Artist=Record+Labels,good

991thefox.com/Jennifer-Grant/1496419,good

997now.radio.com/,good

997now.radio.com/2011/04/11/jennifer-lopez-reveals-love-track-list/,good

99americans.org/,good

9inchfloater.com/,good

9rank.com/acomba.net,good

9rank.com/amour.fr,good

9rank.com/barbot.pt,good

9rank.com/bayloo.com,good

9rank.com/bolduc.ca,good

9rank.com/ckoi.com,good

9rank.com/disneyxd.se,good

9rank.com/ericalexandre.com,good

9rank.com/eros.es,good

9rank.com/fep.umontreal.ca,good

9rank.com/gratuit-chat.fr,good

9rank.com/hymer.com,good

9rank.com/hymer.dk,good

9rank.com/ionline.tv,good

9rank.com/jpmetalamerica.com,good

9rank.com/latido.com.mx,good

9rank.com/militarybest.com,good

9rank.com/mujeres-maduras.es,good

9rank.com/saputo.com,good

9rank.com/scandella.it,good

9rank.com/sexxxy.com,good

9rank.com/sonnik.org,good

9rank.com/telequebec.tv,good

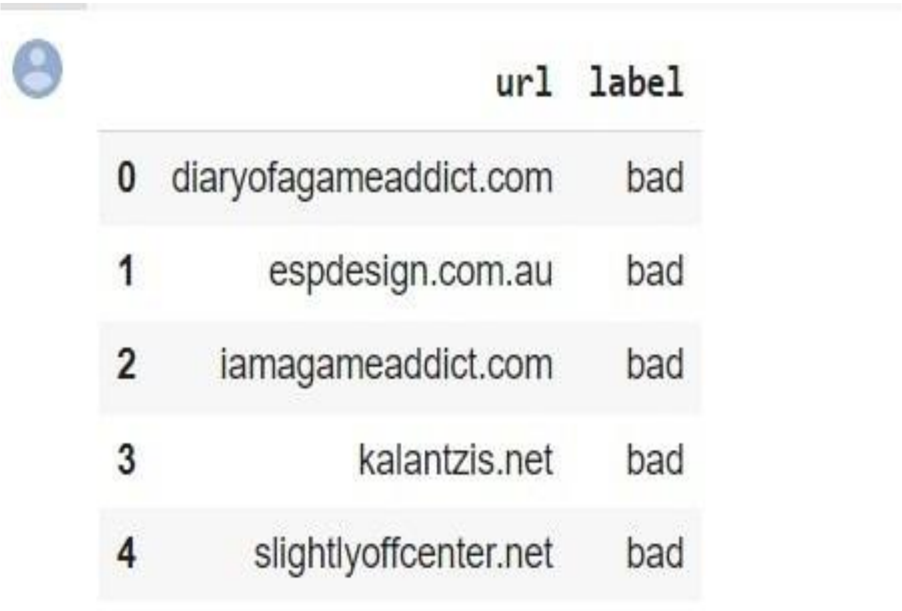
zgsysz.com,bad

oknarai.ru,bad

realinnovation.com/css/menu.js,bad

hardcorepornparty.com,bad

zous.szm.sk,bad



	url	label
0	diaryofagameaddict.com	bad
1	espdesign.com.au	bad
2	iamagameaddict.com	bad
3	kalantzis.net	bad
4	slightlyoffcenter.net	bad

5 SOFTWARE AND HARDWARE SPECIATIONS

5.1 Introduction

We must use operating systems in this using as windows or mac, Laptops or pc like electronic gadgets we must use to execute the purpose, without this thing, we cannot move forward with data storage like this gadget's

5.2 Specific Requirements

Windows

5.3Hardware and Software Requirement

5.3.1 Hardware Requirement

2GB RAM (minimum)

100GB HDD (minimum)

Intel 1.66 GHz Processor Pentium 4 (minimum)

5.3.2 Software Requirement

Jupiter Notebook

WINDOWS 7

6 IMPLEMENTATION

CODE

```
# We will use the Pandas package to load our package. To import Pandas
import pandas as pd
# We can now load the dataset
urls_data = pd.read_csv("C:/Users/mahendra/Downloads/urldata (1).csv")
# After loading the dataset, let’s see how it is structured
urls_data.head()
def makeTokens(f):
tkns_BySlash = str(f.encode('utf-8')).split('/')# make tokens after splitting by slash
total_Tokens = []
for i in tkns_BySlash:
tokens = str(i).split('-')# make tokens after splitting by dash
tkns_ByDot = []
for j in range(0,len(tokens)):
temp_Tokens = str(tokens[j]).split('.')# make tokens after splitting by dot
tkns_ByDot = tkns_ByDot + temp_Tokens
total_Tokens = total_Tokens + tokens + tkns_ByDot
total_Tokens = list(set(total_Tokens))           #remove redundant tokens
if 'com' in total_Tokens:
total_Tokens.remove('com')   #removing .com since it occurs a lot of times and it should not be included in our features
return total_Tokens
url_list = urls_data["url"]
y = urls_data["label"]
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
vectorizer = TfidfVectorizer(tokenizer=makeTokens)
X = vectorizer.fit_transform(url_list)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.100, random_state=75)
logit = LogisticRegression()
logit.fit(X_train, y_train)
# Calculating the model’s accuracy score
""" To calculate the accuracy score"""
print("Accuracy ",logit.score(X_test, y_test))
Accuracy  0.963421885033415
In [10]:
X_predict = ["https://www.section.io/engineering-education/",
"https://www.youtube.com/",
"https://www.traversymedia.com/",
"https://www.kleinehundezuhause.com",
"http://ttps://www.mecymiafinance.com",
"https://www.atlanticoceanicoilandgas.com",
  "freeserials.spb.ru/key/68703.htm"
#We can run predictions on these URLs
X_predict = vectorizer.transform(X_predict)
New_predict = logit.predict(X_predict)
print(New_predict)
['good' 'good' 'bad' 'bad' 'bad' 'bad' 'bad']
X_predict1 =map(str,input().split())
X_predict1 = vectorizer.transform(X_predict1)
New_predict1 = logit.predict(X_predict1)
print(New_predict1)
freeserials.spb.ru/key/68703.htm
['bad']
```

browncountyohio.gov/,good

brownforillinois.com/,good

brownielocks.com/givepeaceachance.html,good

brownmarkfilms.com/,good

brownmccarroll.com/attorneys/kevin-koronka,good

brownpapertickets.com/event/107289,good

brownpapertickets.com/event/119194,good

brownpapertickets.com/event/165622,good

brownpapertickets.com/event/186175,good

brownpapertickets.com/event/186722,good

brownpapertickets.com/profile/14870,good

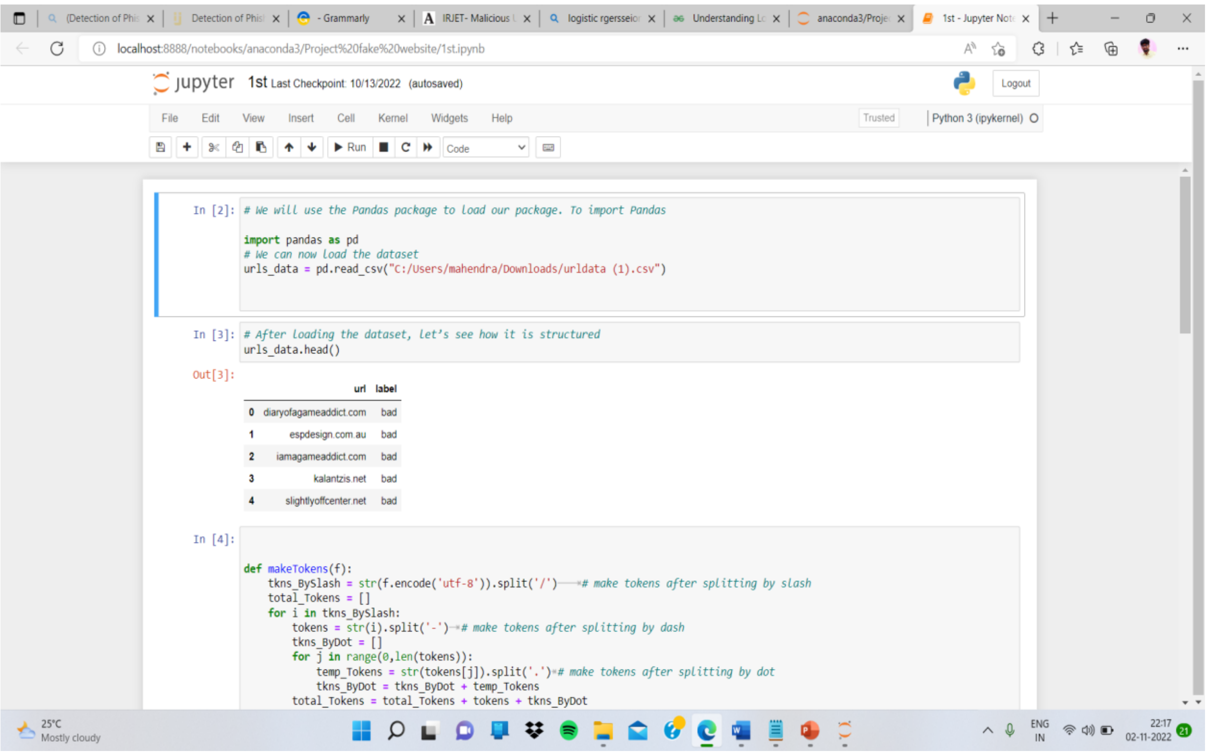
brownplatform.com/2011/10/mad-as-hatter.html,good

brownsgab.com/,good

brownsgab.com/2011/08/31/five-bold-predictions-for-the-browns-2011-season/,good

brownsthebutchers.co.uk/,good

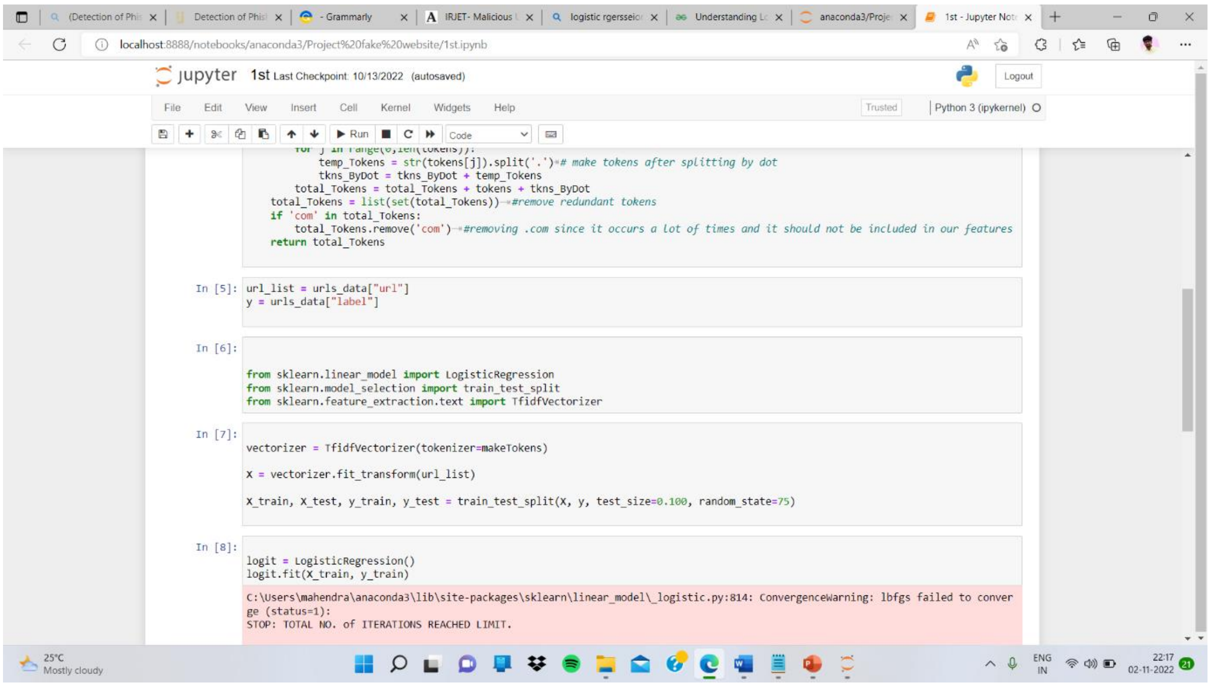
brownsvilleherald.com/articles/big-132967-previews-college.html,good


localhost:8888/notebooks/anaconda3/Project%20fake%20website/1st.ipynb
jupyter 1st Last Checkpoint: 10/13/2022 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
25°C
Mostly cloudy
ENG
IN
22:17
02-11-2022

```
In [2]: # We will use the Pandas package to load our package. To import Pandas

        import pandas as pd
        # We can now load the dataset
        urls_data = pd.read_csv("C:/Users/mahendra/Downloads/urldata (1).csv")

In [3]: # After loading the dataset, let's see how it is structured
        urls_data.head()

Out[3]:
```

	url	label
0	diaryofagameaddict.com	bad
1	espdesign.com.au	bad
2	iamagameaddict.com	bad
3	kalantzis.net	bad
4	slightlyoffcenter.net	bad

```
In [4]:

        def makeTokens(f):
            tkns_BySlash = str(f.encode('utf-8')).split('/')    # make tokens after splitting by slash
            total_Tokens = []
            for i in tkns_BySlash:
                tokens = str(i).split('-')  # make tokens after splitting by dash
                tkns_ByDot = []
                for j in range(0,len(tokens)):
                    temp_Tokens = str(tokens[j]).split('.') # make tokens after splitting by dot
                    tkns_ByDot = tkns_ByDot + temp_Tokens
                total_Tokens = total_Tokens + tokens + tkns_ByDot
```


localhost:8888/notebooks/anaconda3/Project%20fake%20website/1st.ipynb
jupyter 1st Last Checkpoint: 10/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
25°C
Mostly cloudy
ENG
IN
22:17
02-11-2022

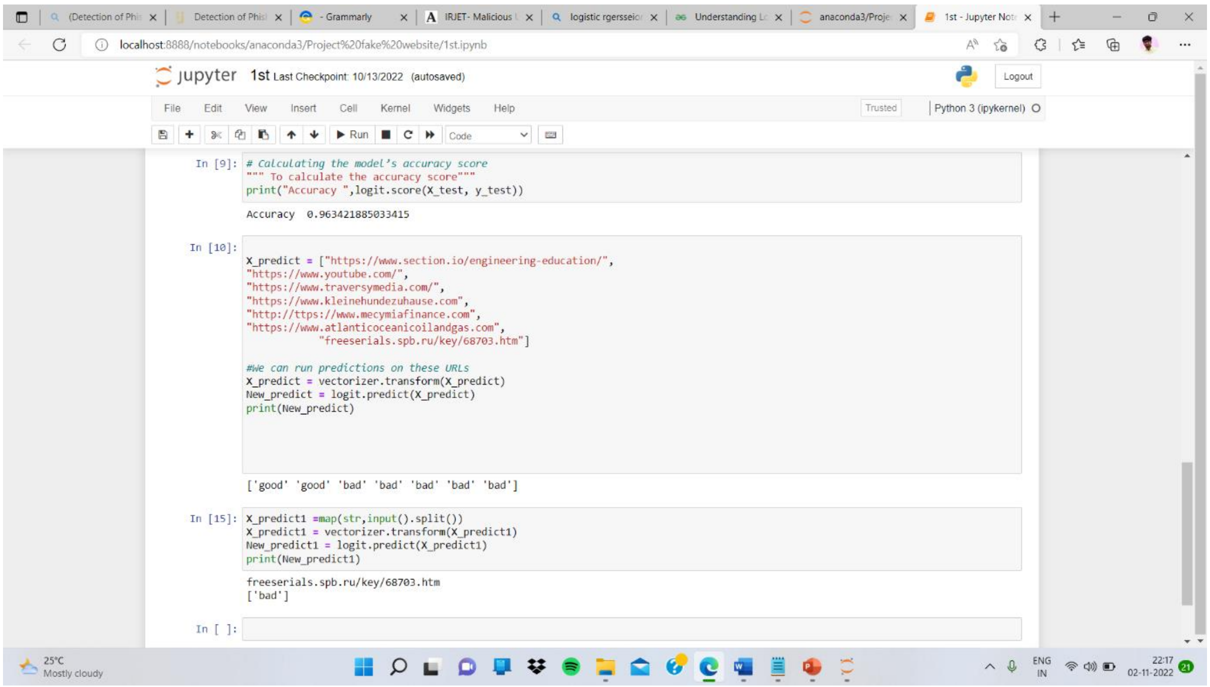
```
        for j in range(0,len(tokens)):
            temp_Tokens = str(tokens[j]).split('.')# make tokens after splitting by dot
            tkns_ByDot = tkns_ByDot + temp_Tokens
        total_Tokens = total_Tokens + tokens + tkns_ByDot
    total_Tokens = list(set(total_Tokens))#remove redundant tokens
    if 'com' in total_Tokens:
        total_Tokens.remove('com')#removing .com since it occurs a lot of times and it should not be included in our features
    return total_Tokens
```

```
In [5]: url_list = urls_data["url"]
        y = urls_data["label"]
```

```
In [6]:

        from sklearn.linear_model import LogisticRegression
        from sklearn.model_selection import train_test_split
        from sklearn.feature_extraction.text import TfidfVectorizer
```

```
In [7]:
        vectorizer = TfidfVectorizer(tokenizer=makeTokens)

        X = vectorizer.fit_transform(url_list)

        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.100, random_state=75)
```

```
In [8]:
        logit = LogisticRegression()
        logit.fit(X_train, y_train)

        C:\Users\mahendra\anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarning: lbfgs failed to conver
        ge (status=1):
        STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```


localhost:8888/notebooks/anaconda3/Project%20fake%20website/1st.ipynb
jupyter 1st Last Checkpoint: 10/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
25°C
Mostly cloudy
ENG
IN
22:17
02-11-2022

```
In [9]: # Calculating the model's accuracy score
        """ To calculate the accuracy score"""
        print("Accuracy ",logit.score(X_test, y_test))

        Accuracy  0.963421885033415
```

```
In [10]:
         X_predict = ["https://www.section.io/engineering-education/",
         "https://www.youtube.com/",
         "https://www.traversymedia.com/",
         "https://www.kleinehundezuhause.com",
         "http://ttps://www.mecymiafinance.com",
         "https://www.atlanticoceanicoilandgas.com",
                     "freeserials.spb.ru/key/68703.htm"]

         #We can run predictions on these URLs
         X_predict = vectorizer.transform(X_predict)
         New_predict = logit.predict(X_predict)
         print(New_predict)

         ['good' 'good' 'bad' 'bad' 'bad' 'bad' 'bad']
```

```
In [15]: X_predict1 =map(str,input().split())
         X_predict1 = vectorizer.transform(X_predict1)
         New_predict1 = logit.predict(X_predict1)
         print(New_predict1)

         freeserials.spb.ru/key/68703.htm
         ['bad']
```

```
In [ ]:
```

7 TESTING

Without errors executed in Jupiter notebook means it use for executing python language codes it is a compiler, Successful run and we got correct output

8 EXPERIMENTAL RESULTS

8.1 Logistic Regression

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False etc. This is a Scikit-learn algorithm that we will use to train our model. This algorithm will enable our model to understand patterns and relationships in our dataset. The model will gain useful knowledge and insight, which it will use to make predictions. This is a Logistic Regression algorithm that we will use to train our model. This algorithm will enable our model to understand patterns and relationships in our dataset. The model will gain useful knowledge, which it will use to make predictions.

Algorithm	**Accuracy on training dataset**	**Accuracy on testing dataset**
Logistic Regression	96.3%	96.3%

9 CONCLUSIONS

This document has presented three important elements of the study, a theory of phishing crime, a review of anti- phishing technique of the research gaps. Phishing will never be eliminated, but it is important to understand this crime before proposing any solution. Here, we have discussed about different features of phishing attacks and different techniques to detect phishing websites. And detection of URL.

Malicious URL detection is very useful in determining that the certain website is malicious or not and it should be visited or not. This will help the user a lot in knowing that which of the websites should be avoided. Hence, it will prevent them in revealing their sensitive information to the phisher. It can be very useful for security purpose**,** learn about new phishing techniques that are being developed to avoid falling prey to one. Think before you click Never click on hyperlinks without examining the hidden URL.

10 FUTURE WORK

The future work of the proposed system is to evaluate these machine learning classifiers with larger dataset.

AI and machine learning techniques are now used in finance, psychology, and economics… and will soon be present in many Infosec processes.

Cyber security and all the subsets of AI and Computer Science can work together to create intelligent and effective solutions to the new threats and issues that are breaking innovation.

Automatized detection of bad URLs based on machine learning and not human instructions can be a little piece of this puzzle. However, machine learning is not a magic solution and is not without its threats.

REFERENCES

[1] J. Shad and S. Sharma, A Novel Machine Learning Approach to Detect Phishing Websites Jaypee Institute of Information Technology, pp. 425430, 2018.

[2] Atharva Deshpande, Omkar Pedamkar, Nachiket Chaudhary, Dr. Swapna Borde, A Paper Detection of Phishing Websites using Machine Learning in IJERT ,2021

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

[4] L. MacHado and J. Gadge, Phishing Sites Detection Based on C4.5 Decision Tree Algorithm, in 2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017, 2018, pp. 15.

[5] A. Desai, J. Jatakia, R. Naik, and N. Raul, Malicious web content detection using machine leaning, RTEICT 2017 – 2nd IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. Proc., vol. 2018Janua, pp. 14321436, 2018.

[6] 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.

[7] 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