

Malicious URL Prediction Based On Community Detection

Submitted in partial fulfillment of the requirements for
the award of

Bachelor of Technology degree in Information Technology by

Vaishnavi D (37120079)

Suwetha S (37120901)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC JEPPIAAR
NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600

119

MARCH – 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" Grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119.

Phone: 044 - 2450 3150 / 51 / 52 / 54 / 55 Fax: 044 - 2450 2344

www.sathyabama.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Vaishnavi D (37120079) and Suwetha S (37120901) who carried out the project entitled "Malicious URL prediction based on community detection" under our supervision from November 2020 to March 2021.

Internal Guide

Dr.Y.Bevish Jinila, M.E., Ph.D.,

Head of the Department

Dr. R. SUBHASHINI, M.E., Ph.D.,

Submitted for Viva voice Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I am Vaishnavi D (37120079) hereby declare that the project report entitled on **“Malicious URL prediction based on community detection”** done by me under the guidance of Dr.Y.Bevis Jinila, M.E.,Ph.D.,is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE:

PLACE: Chennai

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr.SUBHASHINI M.E.,Ph.D.**, Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms Dr.Y.Bevis Jinila, M.E., Ph.D.**,for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTRACT

There are number of users who purchase products online and make payment through various websites. There are multiple websites who ask user to provide sensitive data such as username, password or credit card details etc. often for malicious reasons. This type of websites is known as phishing website. In order to detect and predict phishing website, we proposed an intelligent, flexible and effective system that is based on using Machine learning technique. We implemented classification algorithm and techniques to extract the phishing data sets criteria to classify their legitimacy. The 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. Our system will use machine learning algorithm to detect whether the website is phishing website or not. This application can be used by many E-commerce enterprises in order to make the whole transaction process secure. Machine learning 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.

LIST OF FIGURES

FIGURE NO	NAME OF THE FIGURE	PAGE NO.
1.1	Machine learning architecture	2
1.2	Supervised learning	3
1.3	Unsupervised learning	4
1.4	Reinforcement learning	5
3.1	System Architecture	12
3.2	Dataset collection	14
3.3	Data preprocessing	15
3.4	Decision tree classifier	17
3.5	Logistic regression	18
5.1	Graph-URL Dataset	21

LIST OF TABLES

TABLE NO	NAME OF THE TABLE	PAGE NO.
1.1	Accuracy table-Decision tree classifier	22
1.2	Accuracy table-Logistic Regression	22

LIST OF SYMBOLS AND ABBREVIATIONS

CNN	Convolutional Neural Network
ML	Machine Learning
ReLU	Rectified Linear Unit
SARSA	State-Action-Reward-State-Action
DDPG	Deep Deterministic Policy Gradient

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO.
	ABSTRACT	iv
	LIST OF FIGURES	v
	LIST OF TABLES	vi
	LIST OF SYMBOLS AND ABBREVIATION	vii
1	INTRODUCTION	1
	1.1 TECHNOLOGY USED	1
	1.1.1 Machine learning	1
	1.1.2 Machine learning approaches	3
	1.2 OBJECTIVES	5
2	LITERATURE SURVEY	6
3	SYSTEM ANALYSIS	11
	3.1 PROPOSED SYSTEM	11
	3.2 ARCHITECTURE DIAGRAM	12
	3.3 MODULE DESCRIPTION	13
4	SOFTWARE DESCRIPTION	19
	4.1 JUPYTER NOTEBOOK	19
	4.2 PYTHON	20
5	RESULTS AND DISCUSSION	21
	5.1 FINAL RESULTS OBTAINED	21
6	CONCLUSION AND FUTURE WORK	23

6.1 CONCLUSION	23
6.2 FUTURE WORK	23
REFERENCES	24
APPENDIX	
(A)Source code	25
(B)Snapshots	29
(C)Plagiarism Report	30

CHAPTER -1

INTRODUCTION

Phishing is a cyber attack that uses disguised email as a weapon. The goal is to trick the email recipient into believing that the message is something they want or need — a request from their bank, for instance, or a note from someone in their company — and to click a link or download an attachment. What really distinguishes phishing is the form the message takes: the attackers masquerade as a trusted entity of some kind, often a real or plausibly real person, or a company the victim might do business with. It's one of the oldest types of cyberattacks, dating back to the 1990s, and it's still one of the most widespread and pernicious, with phishing messages and techniques becoming increasingly sophisticated. Phishing accounted for 29 per cent of all fraud attacks in the first quarter of 2019 and India stood second to the US on the list of top phishing hosting countries, Phishing is the fraudulent attempt to obtain sensitive information such as usernames, passwords and credit card details by disguising oneself as a trustworthy entity in an electronic communication. Phishing and malware-based attacks are the most prolific online fraud tactics globally and India is one of the top three target countries for such attacks, says a report. According to the RSA Quarterly Fraud Report for the period between January 1 to March 31, 2018, phishing accounted for 48 per cent of all cyber-attacks. The report that contains fraud attack and consumer fraud data and analysis, noted that Canada, the United States, India and Brazil were the countries most.

1.1 TECHNOLOGY USED

1.1.1 Machine Learning

Machine Learning is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn. Machine learning is actively being used today, perhaps in many more places than one would expect.A

subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. In its application across business problems, machine learning is also referred to as predictive analytics.

Machine learning involves computers discovering how they can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks. For simple tasks assigned to computers, it is possible to program algorithms telling the machine how to execute all steps required to solve the problem at hand; on the computer's part, no learning is needed. For more advanced tasks, it can be challenging for a human to manually create the needed algorithms. In practice, it can turn out to be more effective to help the machine develop its own algorithm, rather than having human programmers specify every needed step.

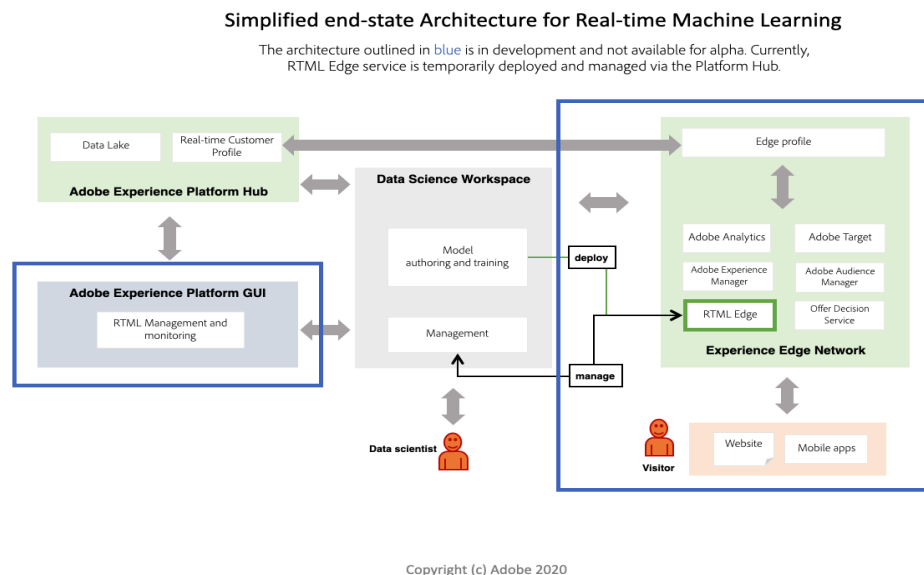


Figure 1.1 Machine learning architecture

1.1.2 Machine Learning approaches

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

Supervised learning:

Supervised learning as the name indicates the presence of a supervisor as a teacher. Basically, supervised learning is a learning in which we teach or train the machine using data which is well labelled that means some data is already tagged with the correct answer. After that, the machine is provided with a new set of examples(data) so that supervised learning algorithm analyses the training data(set of training examples) and produces a correct outcome from labelled data.

Supervised learning is where there are input variables (x) and an output variable (Y) and an algorithm is used to learn the mapping function from the input to the output.

$$Y = f(X)$$

The goal is to approximate the mapping function so well that when there is a new input data (x) that the output variables (Y) for that data can be predicted easily.

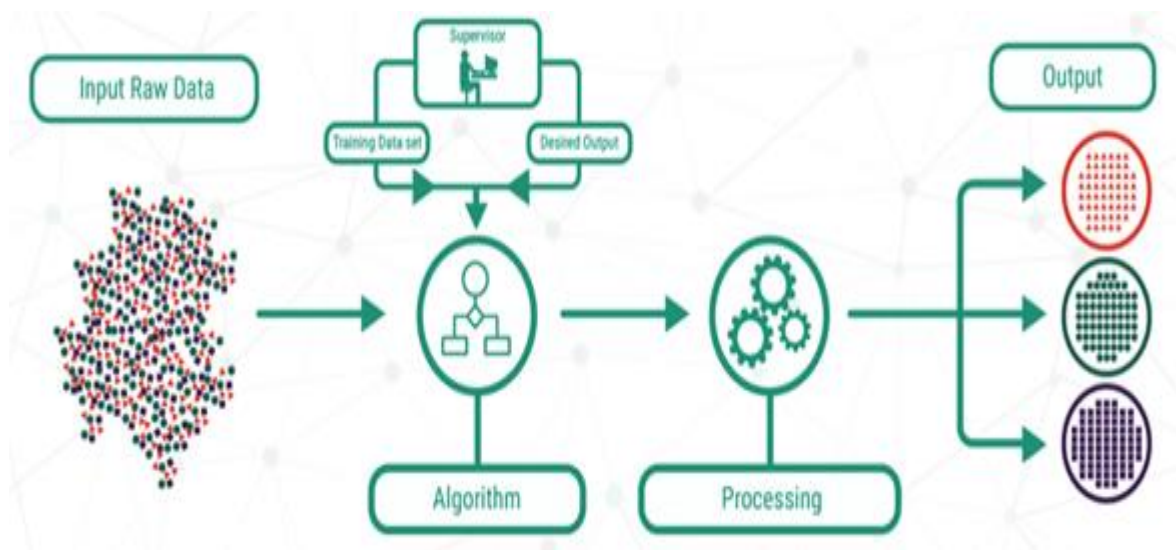


Figure 1.2 Supervised learning flowchart

Unsupervised learning:

Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data.

Unlike supervised learning, no teacher is provided that means no training will be given to the machine. Therefore, machine is restricted to find the hidden structure in unlabeled data by itself.

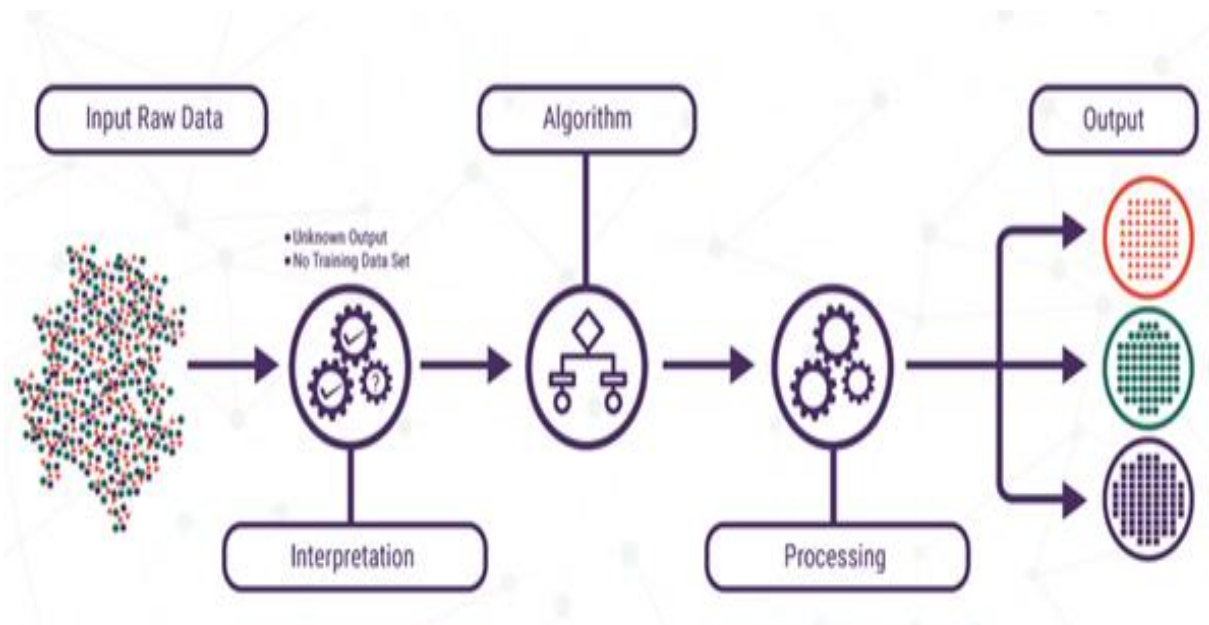


Figure 1.3 Unsupervised learning flowchart

Reinforcement learning:

A computer program interacts with a dynamic environment in which it must perform a certain goal. As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.



Figure 1.4 Reinforcement learning flowchart

1.2 OBJECTIVES:

- URLs sometimes known as “Web links” are the primary means by which users locate information in the Internet.
- Our aim is to derive classification models that detect phishing web sites by analysis of the lexical and host-based features of URLs.
- We analyze different classifying algorithms to find the spam website.

CHAPTER -2

LITERATURE SURVEY

Qi Li, Mingyu Cheng, Junfeng Wang, Bowen Sun,"LSTM based Phishing Detection for Big Email Data"[2020]

In recent years, cyber criminals have successfully invaded many important information systems by using phishing mail, causing huge losses. The detection of phishing mail from big email data has been paid public attention. However, the camouflage technology of phishing mail is becoming more and more complex, and the existing detection methods are unable to confront with the increasingly complex deception methods and the growing number of email. In this paper, we proposed a LSTM based phishing detection method for big email data. The new method includes two important stages, sample expansion stage and testing stage under sufficient samples. In the sample expansion stage, we combined KNN with K-Means to expand the training data set, so that the size of training samples can meet the needs of in-depth learning. In the testing stage, we first preprocess these samples, including generalization, word segmentation and word vector generation. Zuochao Dou, Student Member, IEEE, Issa Khalil, Member, IEEE, Abdallah Khreishah, Member, IEEE, Ala Al-Fuqaha, Senior Member, IEEE and Mohsen Guizani, Fellow, IEEE,"Systematization of Knowledge (SoK): A Systematic Review of Software Based Web Phishing Detection"

Then, the preprocessed data is used to train a LSTM model. Finally, on the basis of the trained model, we classify the phishing emails. By experiment, we evaluate the performance of the proposed method, and experimental results show that the accuracy of our phishing detection method can reach 95%.

Pros:

The system effectively identifies the phishing attacks through E-mail.

Cons:

The system is still prone to other forms of the phishing attacks.

C.Emiten shyni, Anesh D Suda, G.S. Edwin Ebby Phishing Detection in websites using Parse Tree validation-2018

Phishing is a form of cyber attack that leverages social engineering approaches and other sophisticated techniques to harvest personal information from users of websites. The average annual growth rate of the number of unique phishing websites detected by the Anti Phishing Working Group is 36.29% for the past six years and 97.36% for the past two years. In the wake of this rise, alleviating phishing attacks has received a growing interest from the cyber security community. Extensive research and development have been conducted to detect phishing attempts based on their unique content, network, and URL characteristics. Existing approaches differ significantly in terms of intuitions, data analysis methods, as well as evaluation methodologies. This warrants a careful systematization so that the advantages and limitations of each approach, as well as the applicability in different contexts, could be analyzed and contrasted in a rigorous and principled way. This paper presents a systematic study of phishing detection schemes, especially software based ones. Starting from the phishing detection taxonomy, we study evaluation datasets, detection features, detection techniques, and evaluation metrics. Finally, we provide insights that we believe will help guide the development of more effective and efficient phishing detection schemes.

Pros:

Provides insights on improving the phishing detection methodologies.

Cons:

The system is less accurate.

Happy chapla;Riddhi kotak;Mittal joiser-“A machine Approach for URL based web phishing using Fuzzy logic as classifier”-2019

Phishing is the major problem of the internet era. In this era of internet the security of our data in web is gaining an increasing importance. Phishing is one of the most harmful ways to unknowingly access the credential information like username, password or account number from the users. Users are not aware of this type of attack and later they will also become a part of the phishing attacks. It may be the losses of financial found, personal information, reputation of brand name or trust of brand. So the

detection of phishing site is necessary. In this paper we design a framework of phishing detection using URL.

Pros:

The proposed method improved the forecast accuracy over selected benchmark algorithms by about 20% to 80%, according to four performance metrics: MAE, MAPE, RMSE, NRMSE.

Cons:

However, the demand for energy has dissimilar profiles variations depending on the type of day, weather conditions and geographical area.

Junaid Rashid, Toqeer Mahmood, Muhammad Wasif Nisar Phishing Detection using machine learning techniques -2020

Today, everyone is highly dependent on the internet. Everyone performed online shopping and online activities such as online Bank, online booking, online recharge and more on internet. Phishing is a type of website threat and phishing is Illegally on the original website Information such as login id, password and information of credit card. This paper proposed an efficient machine learning based phishing detection technique. Overall, experimental results show that the proposed technique, when integrated with the Support vector machine classifier, has the best performance of accurately distinguishing 95.66% of phishing and appropriate websites using only 22.5% of the innovative functionality. The proposed technique exhibits optimistic results when benchmarking with a range of standard phishing datasets of the “University of California Irvine (UCI)” archive. Therefore, proposed technique is preferred and used for phishing detection based on machine learning.

Pros:

An optimal weighted ensemble approach is employed to combine these forecasts and provide the final forecasting result.

Cons:

The system uses only a single machine learning algorithm; hence it may not produce accurate results.

JIAN MAO¹ , (Member, IEEE), WENQIAN TIAN¹ , PEI LI¹ , TAO WEI² , (Member, IEEE), AND ZHENKAI LIANG³ , (Member, IEEE),”Phishing-Alarm: Robust and Efficient Phishing Detection via Page Component Similarity”[2017]

Social networks have become one of the most popular platforms for users to interact with each other. Given the huge amount of sensitive data available in social network platforms, user privacy protection on social networks has become one of the most urgent research issues. As a traditional information stealing technique, phishing attacks still work in their way to cause a lot of privacy violation incidents. In a Web-based phishing attack, an attacker sets up scam Web pages (pretending to be an important Website such as a social network portal) to lure users to input their private information, such as passwords, social security numbers, credit card numbers, and so on. In fact, the appearance of Web pages is among the most important factors in deceiving users, and thus, the similarity among Web pages is a critical metric for detecting phishing Websites. In this paper, we present a new solution, called Phishing-Alarm, to detect phishing attacks using features that are hard to evade by attackers. In particular, we present an algorithm to quantify the suspiciousness ratings of Web pages based on the similarity of visual appearance between the Web pages. Since cascading style sheet (CSS) is the technique to specify page layout across browser implementations, our approach uses CSS as the basis to accurately quantify the visual similarity of each page element. As page elements do not have the same influence to pages, we base our rating method on weighted page-component similarity. We prototyped our approach in the Google Chrome browser. Our large-scale evaluation using real-world websites shows the effectiveness of our approach. The proof of concept implementation verifies the correctness and accuracy of our approach with a relatively low performance overhead.

Pros:

An optimal weighted ensemble approach is employed to combine these forecasts and provide the final forecasting result.

Cons:

The system uses only a single machine learning algorithm; hence it may not produce accurate results.

CHAPTER - 3

SYSTEM ANALYSIS

3.1 PROPOSED SYSTEM

There are number of users who purchase products online and make payment through various websites. There are multiple websites who ask user to provide sensitive data such as username, password or credit card details etc. often for malicious reasons. This type of websites is known as phishing website. In order to detect and predict phishing website, we proposed an intelligent, flexible and effective system that is based on using Machine learning technique. We implemented classification algorithm and techniques to extract the phishing data sets criteria to classify their legitimacy. The 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. Our system will use machine learning algorithm to detect whether the website is phishing website or not. This application can be used by many E-commerce enterprises in order to make the whole transaction process secure. Machine learning 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.

ADVANTAGES OF PROPOSED SYSTEM:

- The system makes use of two machine learning algorithms logistic regression and decision tree classifier to effectively identify and predict phishing attacks.
- The system gives a comparative analysis of the results obtained from using the two machine learning algorithms.
- The system makes use of the best machine learning algorithm to accurately identify the phishing attacks.

APPLICATIONS:

- Online cybersecurity
- Security softwares
- Other cybersecurity applications

3.2 ARCHITECTURE DIAGRAM:

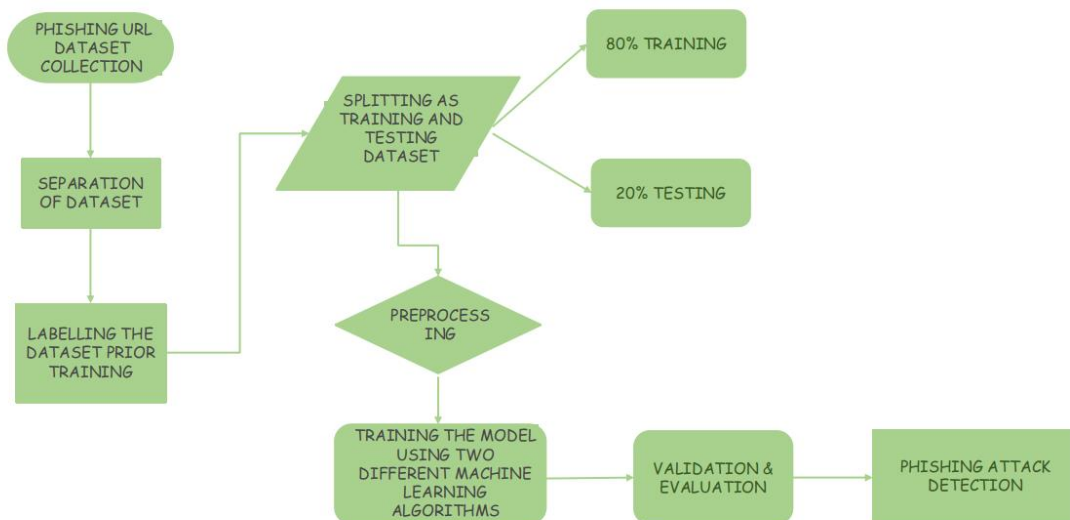


Figure 3.1 System Architecture

WORKING:

In order to detect and predict phishing website, we proposed an intelligent, flexible and effective system that is based on using Machine learning technique. We implemented classification algorithm and techniques to extract the phishing data sets criteria to classify their legitimacy. The 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. Our system will use machine learning algorithm to detect whether the website is phishing website or not. This

application can be used by many E-commerce enterprises in order to make the whole transaction process secure. Machine learning 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.

3.3MODULE DESCRIPTION:

Dataset collection

- Data preprocessing
- Splitting of Dataset
- Dataset clustering

Training with algorithm

Evaluation

DATASET COLLECTION MODULE:

A data set is a collection of data. Machine learning has become the go-to method for solving many challenging real-world problems. It's definitely by far the best performing method for prediction tasks. These machine learning machines that have been working so well need fuel lots of fuel; that fuel is data. The more labelled data available, the better our model performs. The idea of more data leading to better performance has even been explored at a large-scale by Google with a dataset of 300 Million images! When deploying a machine learning model in a real-world application, data must be constantly fed to continue improving its performance. And, in the machine learning era, data is very well arguably the most valuable resource. There are three steps of collecting data.

Classification. When an algorithm to answer binary yes-or-no questions or to make a multi-class classification (grass, trees, or bushes; cats, dogs, or birds etc.)

Regression. For an algorithm to yield some numeric value. For example, if you spend too much time coming up with the right price for your product since it depends on many factors, regression algorithms can aid in estimating this value.

Ranking. Some machine learning algorithms just rank objects by a number of features. Ranking is actively used to recommend movies in video streaming services or show the products that a customer might purchase with a high probability based on his or her previous search and purchase activities.

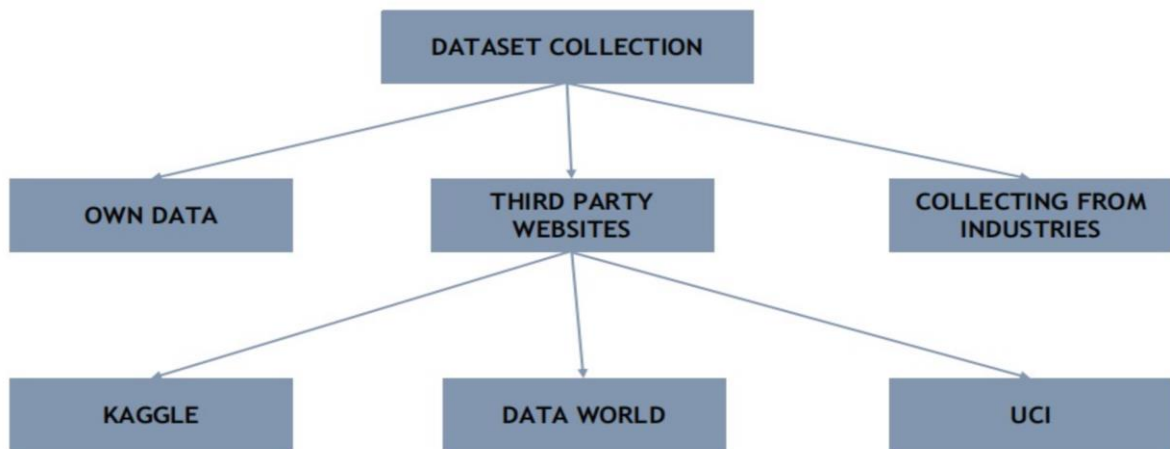


Figure 3.2 Dataset collection

DATA PRE-PROCESSING:

Data pre-processing is a cleaning technique which is used to convert / transform the raw data into a clean and properly structured dataset suitable for further analysis. Data is usually collected and gathered from various sources , so it should be good enough and in some specific format before the model learns or gets trained with the data. This will help in achieving better and accurate results with valuable information. The basic steps in pre-processing involve filling up missing values and null values , getting rid of possible outliers and normalisation.

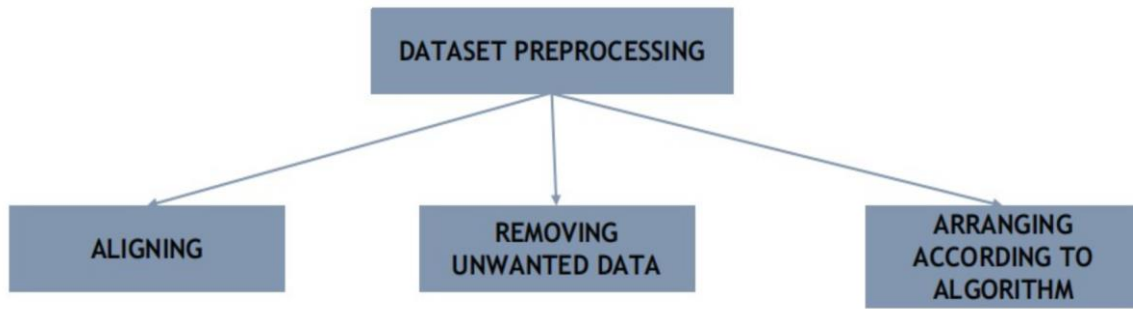


Figure 3.3 Dataset preprocessing

SPLITTING OF DATASET

In machine learning ,any dataset is usually split into two : training data and test data. The output variable along with other variables are included in the training set . The model learns the data and tries to generate some pattern . The other part of the dataset serves as a test set to validate our model's prediction. The scikit library has a function called `train_test_split` to divide our data . `test_size` is the parameter which gives us the percentage of data that should belong to the test set. `train_size` stores the remaining part as the training dataset , either of which should be specified. `random_state` acts as a random number generator . For our dataset , we split the training and testing set with 80 , 20 ratio the random state is passed as 0.

DATASET CLUSTERING

Dataset Clustering or cluster analysis is a machine learning technique, which groups the unlabelled dataset. It can be defined as "A way of grouping the data points into different clusters, consisting of similar data points. The objects with the possible similarities remain in a group that has less or no similarities with another group."

It does it by finding some similar patterns in the unlabelled dataset such as shape, size, color, behavior, etc., and divides them as per the presence and absence of those similar patterns.It is an unsupervised learning method, hence no supervision is provided to the algorithm, and it deals with the unlabeled dataset.After applying this

clustering technique, each cluster or group is provided with a cluster-ID. ML system can use this id to simplify the processing of large and complex datasets. The clustering technique is commonly used for statistical data analysis.

TRAINING WITH ALGORITHM

The system gives a comparative analysis of the accuracies achieved from training using two different machine learning algorithms logistic regression and decision tree classifier.

Decision tree classifier

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A decision tree simply asks a question, and based on the answer (Yes/No), it further splits the tree into subtrees. There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision tree:

- Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.
- The logic behind the decision tree can be easily understood because it shows a tree-like structure.

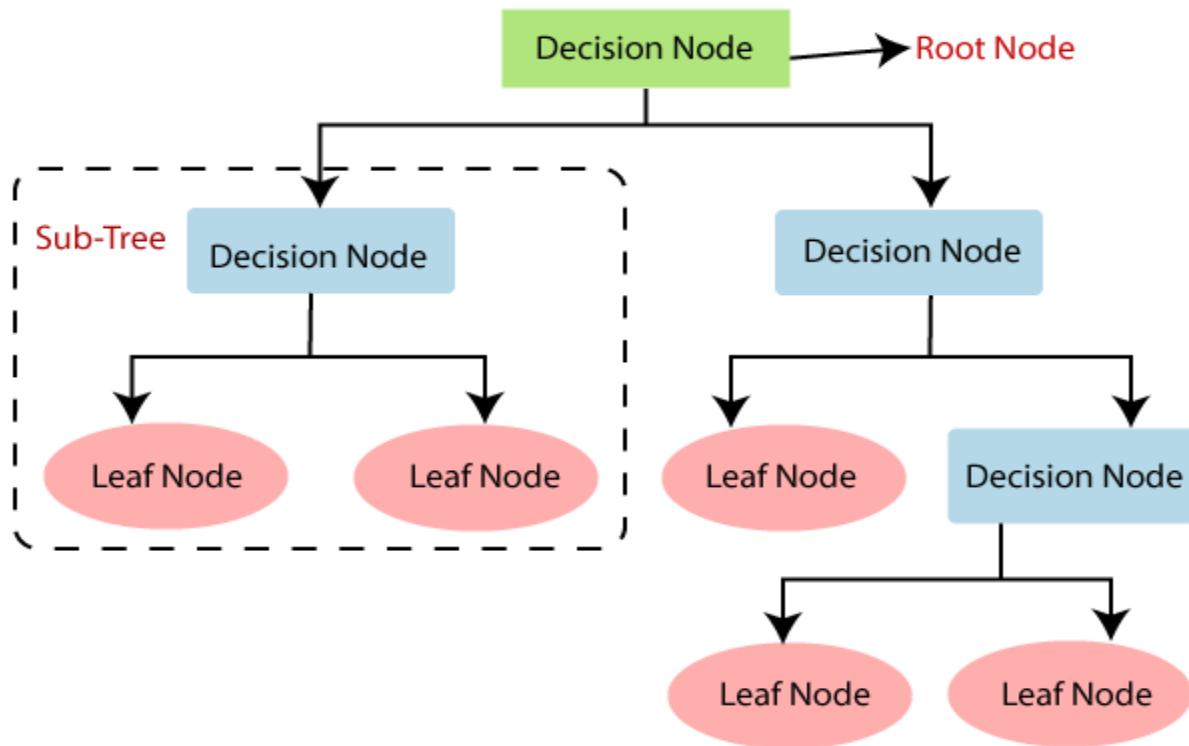


Figure 3.4 Decision tree classifier

Logistic regression

Logistic Regression statistical method is used for analyzing the dataset and produces a binary outcome. One or more autonomous variables may have consisted of the dataset. The result is determined by these variables that are dichotomous in nature. Which means only two results are possible. It is a specific category of regression and it is used in the best way to predict the binary and categorical output. Logistical Regression method is used to regulate the impact of numerous autonomous variables which are conferred at the same time. This method also predicts any one of the two independent categories of variables. Logistic regression designs the best-fitting function with the help of the maximum likelihood method in order to maximize the probability of

classifying the recognized data into the proper division. Various appliances of logistic regression are forecast market trends, to find the success and failure rates in results, the true or false category in recruiting employees based on their performance in need of employment in a company, image categorization, health care and analyze a group of people affected by disease.

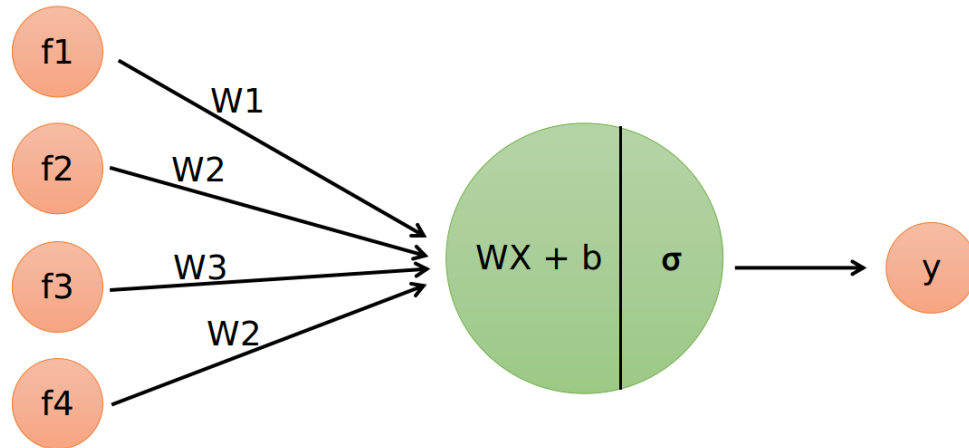


Figure 3.5 Logistic regression

EVALUATION:

To evaluate the accuracy and to detect spam websites, we have used precision recall F1 score and F1 measure in logistic regression and decision tree classifier. As a result we get 85% for the decision tree classifier and 97% for logistic regression. The output of the projected model displays whether the URL is malicious or not.

CHAPTER 4

SOFTWARE DESCRIPTION

4.1 JUPYTER NOTEBOOK:

In this project the jupyter notebook is used as an IDE.

At some point, we all need to show our work. Most programming work is shared either as raw source code or as a compiled executable. The source code provides complete information, but in a way that's more "tell" than "show." The executable shows us what the software does, but even when shipped with the source code it can be difficult to grasp exactly how it works.

Imagine being able to view the code and execute it in the same UI, so that you could make changes to the code and view the results of those changes instantly, in real time? That's just what Jupyter Notebook offers. Jupyter Notebook was created to make it easier to show one's programming work, and to let others join in. Jupyter Notebook allows you to combine code, comments, multimedia, and visualizations in an interactive document — called a notebook, naturally — that can be shared, re-used, and re-worked.

And because Jupyter Notebook runs via a web browser, the notebook itself could be hosted on your local machine or on a remote server.

4.2 PYTHON:

In this project python is used as programming language. In technical terms, Python is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and dynamic binding options. Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program

maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Additionally, Python supports the use of modules and packages, which means that programs can be designed in a modular style and code can be reused across a variety of projects. Once you've developed a module or package you need, it can be scaled for use in other projects, and it's easy to import or export these modules. One of the most promising benefits of Python is that both the standard library and the interpreter are available free of charge, in both binary and source form. There is no exclusivity either, as Python and all the necessary tools are available on all major platforms. Therefore, it is an enticing option for developers who don't want to worry about paying high development costs. If this description of Python is over your head, don't worry. You'll understand it soon enough. What you need to take away from this section is that Python is a programming language used to develop software on the web and in app form, including mobile. It's relatively easy to learn, and the necessary tools are available to all free of charge.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 FINAL RESULTS OBTAINED

To begin with, testing of the trained model, we can split our project into modules of implementation that is done.

In this project makes use of two different machine learning algorithms logistic regression and decision tree classifier. The output of all the classifiers listed on the phishing dataset is contrasted in this project. We validated these algorithms on 1003 research samples using a number of performance metrics, and the results are tabulated here with graphs

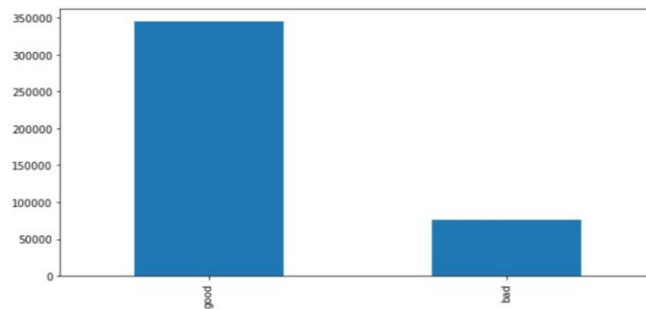


Fig 5.1:Graph-URL dataset

The above graph represents the performance analysis of the URL Datasets . Many Machine Learning researchers have developed algorithms and methods to identify phishing websites. After a detailed analysis, we have come to the conclusion that the vast majority of the study is conducted by well-known machine learning algorithms. In terms of accuracy, precision, memory, and other considerations, a number of variations were used. by trial and error detection strategies that are thriving phishing website URLs. The extension permits the straightforward preparation of our phishing detection model to end-users. For future enhancements, we intend to build the phishing detection system as scalable internet service will incorporate on-line learning so new phishing attack patterns can simply be learned and improve the accuracy of our models with better feature extraction.

The performance of the Decision tree classifier using the various measures such as precision, recall, and F Score, we determine the accuracy as 85%.

	precision	recall	f1-score	Support
Bad	0.82	0.26	0.40	14964
Good	0.86	0.99	0.92	69129
Accuracy			0.86	84093
macro avg	0.84	0.62	0.66	84093
weighted avg	0.85	0.86	0.83	84093

Table 1:accuracy table-Decision tree classifier

The performance of the Logistic regression using the various measures such as precision, recall, and F Score, we determine the accuracy as 97%.

	precision	recall	f1-score	Support
Bad	0.98	0.88	0.93	14964
Good	0.97	1.00	0.98	69129
Accuracy			0.97	84093
macro avg	0.98	0.94	0.96	84093
weighted avg	0.97	0.97	0.97	84093

Table 2:accuracy table-Logistic Regression

The graph above illustrates the precision of each classifier as calculated against test samples. However, when learning the average success of a classifier in phishing detection, false positives and false negatives are given particular consideration. We don't want users to be able to navigate phishing URLs, so we use false positives are considered to be vital whereas deciding the most effective classifier.

Thus,from the above results and discussions it is clear we have successfully implemented the project for effective phishing detection.

CHAPTER – 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

In this project, we have successfully implemented a system to effectively detect phishing attacks using two different machine learning algorithms such as logistic regression and decision tree classifier. By implementation of the project it is found that logistic regression algorithm is the most accurate algorithm compared to the decision tree classifier. The present cybersecurity methods are inaccurate and inefficient. The system has provided an easy and efficient solution at very cost-effective approach.

6.2 FUTURE WORK

In the coming future, we review this application of phishing attacks detection to detect and overcome large scale cyber attacks with more accuracy and efficiency. The application has good use in the security field. In the electricity department, they have more chance to develop or convert this project in many ways. Thus, this project has an efficient scope in coming future where manual predicting can be converted to computerized prediction in a cheap way.

REFERENCES

- [1] Qi Li, Mingyu Cheng, Junfeng Wang, Bowen Sun, "LSTM based Phishing Detection for Big Email Data"[2020]
- [2] JIAN MAO¹ , (Member, IEEE), WENQIAN TIAN¹ , PEI LI¹ , TAO WEI² , (Member, IEEE), AND ZHENKAI LIANG³ , (Member, IEEE), "Phishing-Alarm: Robust and Efficient Phishing Detection via Page Component Similarity"
- [3] Muhammad Qamar Raza, Student Member, IEEE, N. Mithulananthan, Senior Member, IEEE, Jiaming Li, Kwang Y. Lee, Fellow, IEEE, "Multivariate Ensemble Forecast Framework for Demand Prediction of Anomalous Days"
- [4] C. E. Shyni, Anesh D Suda, G.S. Edwin Ebby "Phishing Detection in websites using Parse Tree validation-2018"
- [5] Mao Tan, Member, IEEE, Siping Yuan, Shuaihu Li, Yongxin Su, Hui Li, and Feng He, "Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning"
- [6] Happy Chapla; Riddhi Kotak; Mittal Joiser- "A machine Approach for URL based web phishing using Fuzzy logic as classifier"-2019
- [7] Junaid Rashid, Toqeer Mahmood, Muhammad Wasif Nisar "Phishing Detection using machine learning techniques -2020"
- [8] Mehmet Korkmaz; Ozgur Koray Sahingoz; Banu Diri- "Detection of phishing websites by using machine learning-Based URL analysis-2020"
- [9] Yong Jie Huang; Jinghui Qiu; Washao Wen- "Phishing URL detection via capsule-based neural network-2019"
- [10] Erzhan Zhu, Yuyang Chau, Cheng Cheng Ye, Xuejun Li, Feng Lin- "OFS-NN: An effective phishing websites detection model based on optimal feature selection on neural network-2019"

APPENDIX

(A) SCOURCE CODE:

```
In [1]: import pandas as pd

In [2]: import numpy as np

In [3]: import random

In [4]: from sklearn.feature_extraction.text import CountVectorizer

In [5]: from sklearn.feature_extraction.text import TfidfVectorizer

In [6]: from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier

In [7]: from sklearn.model_selection import train_test_split, GridSearchCV

In [8]: # Load Url Data
urls_data = pd.read_csv("urldata.csv")

In [9]: type(urls_data)

Out[9]: pandas.core.frame.DataFrame

In [10]: urls_data.head()
```

```
In [9]: type(urls_data)

Out[9]: pandas.core.frame.DataFrame
```

```
In [10]: urls_data.head()

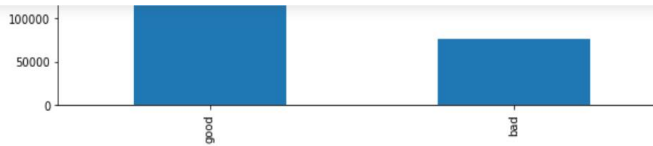
Out[10]:
```

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

```
In [11]: events = urls_data[['label']].replace(' ', 'None')
events.label.value_counts().plot(kind='bar', figsize=(10,5))
```

```
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x1adb704a340>
```





```
In [12]: 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 because it occurs a lot of times
         return total_Tokens
```

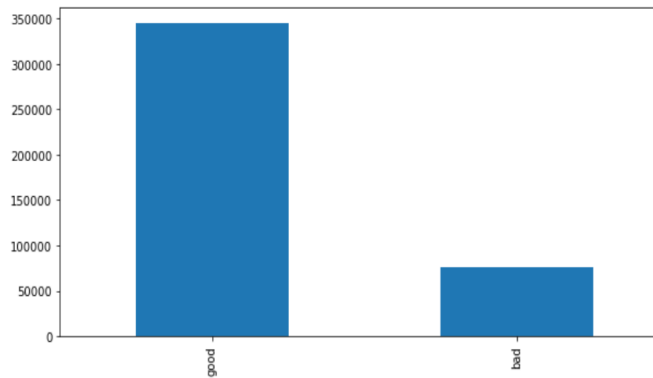
```
In [13]: # Labels
         y = urls_data["label"]
```

```
In [14]: # Features
         url_list = urls_data["url"]
```

```
In [15]: # Using Custom Tokenizer
```

```
In [11]: events = urls_data[['label']].replace(' ', 'None')
         events.label.value_counts().plot(kind='bar', figsize=(10,5))
```

```
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x1adb704a340>
```



```
In [12]: 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 = []
```

```
In [15]: # Using Custom Tokenizer
vectorizer = CountVectorizer(tokenizer=makeTokens)

In [16]: # Store vectors into X variable as Our XFeatures
X = vectorizer.fit_transform(url_list)

In [17]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

In [18]: dt = DecisionTreeClassifier(criterion = "gini", random_state = 100,max_depth=3, min_samples_leaf=5) #using DecisionTree
dt.fit(X_train, y_train)

Out[18]: DecisionTreeClassifier(max_depth=3, min_samples_leaf=5, random_state=100)
```

```
In [19]: from sklearn.metrics import classification_report

dt_pred = dt.predict(X_test)
print(classification_report(y_test,dt_pred))
```

	precision	recall	f1-score	support
bad	0.82	0.26	0.40	14964
good	0.86	0.99	0.92	69129
accuracy			0.86	84093
macro avg	0.84	0.62	0.66	84093
weighted avg	0.85	0.86	0.83	84093

```
In [20]: #using Logistic regression
```

macro avg	0.84	0.62	0.66	84093
weighted avg	0.85	0.86	0.83	84093

```
In [20]: #using Logistic regression
logit = LogisticRegression()
logit.fit(X_train, y_train)
y_pred = logit.predict(X_test)

C:\Users\VAISHNAVI\anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:762: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear\_model.html#logistic-regression
n_iter_i = _check_optimize_result(
```

```
In [21]: from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
bad	0.98	0.88	0.93	14964
good	0.97	1.00	0.98	69129
accuracy			0.97	84093
macro avg	0.98	0.94	0.96	84093
weighted avg	0.97	0.97	0.97	84093

```

In [22]: import tkinter as tk
         root = tk.Tk()

In [*]: canvas1 = tk.Canvas(root,width = 800, height = 600)
         canvas1.pack()

         label1 = tk.Label(root, text='Phishing Website Prediction')
         label1.config(font=('helvetica', 24))
         canvas1.create_window(400, 50, window=label1)

         label2 = tk.Label(root, text='Paste Here:')
         label2.config(font=('helvetica', 20))
         canvas1.create_window(400, 200, window=label2)

         entry1 = tk.Entry (root)
         canvas1.create_window(400, 280, window=entry1)

         def getSquareRoot():
             x1 = entry1.get()

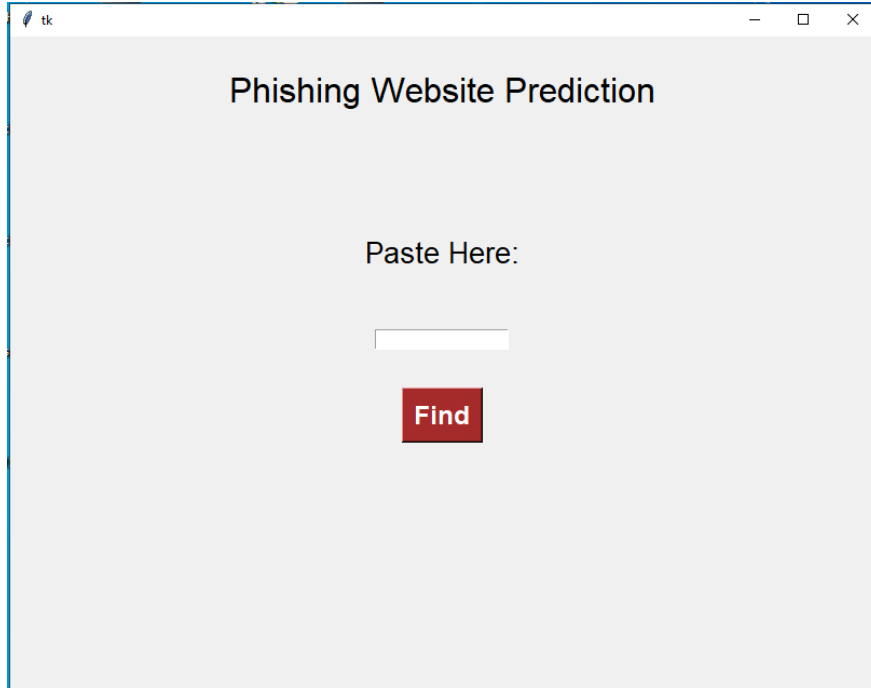
             label3 = tk.Label(root, text= 'The Result For ' + x1 + ' is:',font=('helvetica', 20))
             canvas1.create_window(400, 420, window=label3)

             res = vectorizer.transform([x1])
             prediction = logit.predict(res)

             label4 = tk.Label(root, text= prediction,font=('helvetica', 20, 'bold'))
             canvas1.create_window(400, 460, window=label4)

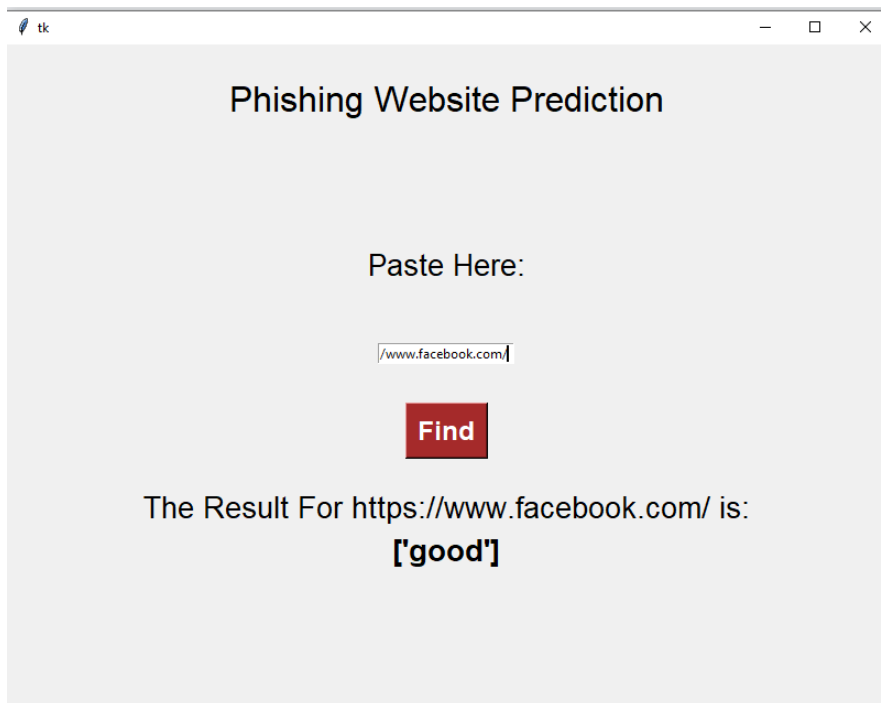
```

(B)SNAPSHOTS:



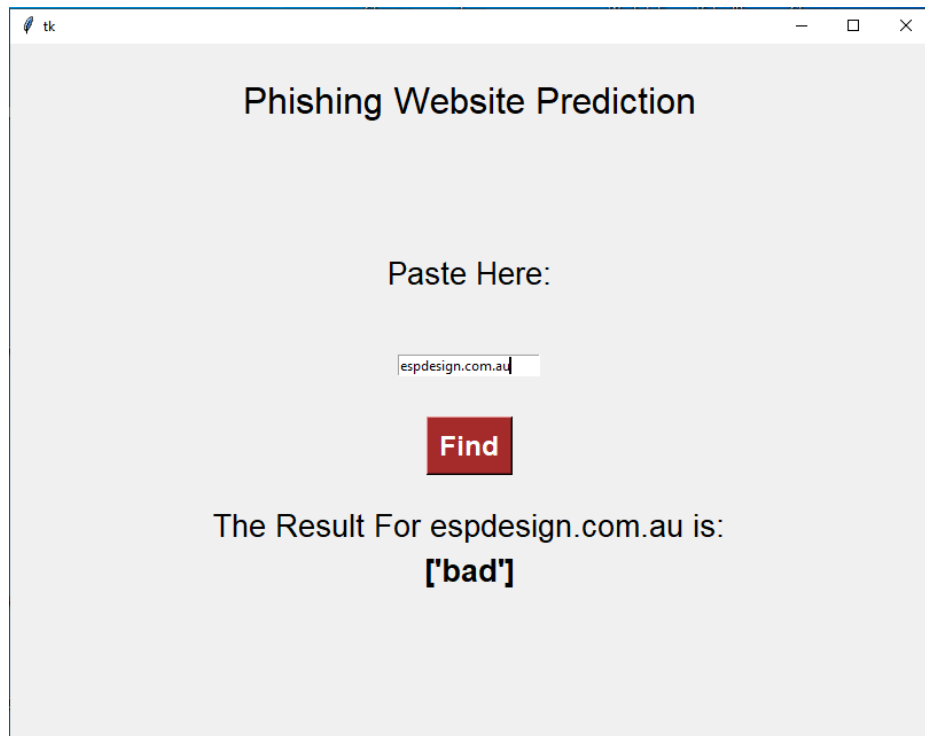
A screenshot of a web application window titled "tk". The main heading is "Phishing Website Prediction". Below it, the text "Paste Here:" is displayed. Underneath is a text input field. Below the input field is a red button with the text "Find".

Phishing web site for phishing URL detection



A screenshot of the same web application window. The text input field now contains the URL "https://www.facebook.com/". The red "Find" button is still present. Below the button, the text "The Result For https://www.facebook.com/ is:" is displayed, followed by the prediction result **['good']** in bold.

Example for a legitimate website URL



Example for a malicious website URL

[C] PLAGIARISM REPORT:

Vaishnavi Swetha - IT			
ORIGINALITY REPORT			
13%	7%	11%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	machinelearningmastery.com	1%	
	Internet Source		
2	Erzhou Zhu, Chengcheng Ye, Dong Liu, Feng Liu, Futian Wang, Xuejun Li. "An Effective Neural Network Phishing Detection Model Based on Optimal Feature Selection", 2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom), 2018	1%	
	Publication		
3	dblp.dagstuhl.de	1%	
	Internet Source		
4	researchr.org	1%	
	Internet Source		
5	espace.library.uq.edu.au		