A

Minor Project Report

on

DETECTION OF PHISHING WEBSITE USING MACHINE LEARNING

Submitted in Partial Fulfillment of
the Requirements for the Third Year
of

Bachelor of Engineering

in

Computer Engineering

to

Kavayitri Bahinabai Chaudhari
North Maharashtra University, Jalgaon

Submitted by

Rohit K. Sonawane
Sakshi P. Mahajan
Chetan J. Mahajan
Vaishnavi V. Gadhe

Under the Guidance of

Mr. Ashish T. Bhole

DEPARTMENT OF COMPUTER ENGINEERING
SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)

2021 - 2022

SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY, BAMBHORI, JALGAON - 425 001 (MS)

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the minor project entitled *Detection of Phishing Website using Machine Learning*, submitted by

Rohit K. Sonawane
Sakshi P. Mahajan
Chetan J. Mahajan
Vaishnavi V. Gadhe

in partial fulfillment of the Third Year of *Bachelor of Engineering* in *Computer Engineering* has been satisfactorily carried out under my guidance as per the requirement of Kavayitri Bahinabai Chaudhari North Maharashtra University, Jalgaon.

Date: April 23,2022
Place: Jalgaon

Mr. Ashish T. Bhole
Guide

Dr. Manoj E. Patil
Head

Prof. Dr. G. K. Patnaik
Principal

Acknowledgements

We would like to express our deep gratitude and sincere thanks to all who helped us in completing project report successfully. Many thanks to almighty God who gave us the strength to do. Our sincere thanks to Prof. Dr. G. K. Patnaik, Principal for providing the facilities to complete the Project report. We would like to express our gratitude and appreciation to all who gave us the possibility to complete the report. A special thanks to Dr. Manoj E. Patil, Associate Professor, Head of the Department, whose help, stimulating suggestions and encouragement, helped us in writing the report. We would also like to thank Mr. Ashish T. Bhole, Associate Professor and Project Guide, who has given his full effort in guiding us and achieving the goal as well as his encouragement to maintain the progress in track. I am also sincerely thankful to Mrs. Shital A. Patil, Assistant Professor, Incharge of the Project, for his valuable suggestions and guidance. We would also like to appreciate the guidance given by other supervisor which has improved our presentation skills by their comments and tips. Last but not the least, We are extremely thankful to our parents and friends without whom it could not reach its successful completion.

Rohit K. Sonawane
Sakshi P. Mahajan
Chetan J. Mahajan
Vaishnavi V. Gadhe

Contents

List of Tables

List of Figures

Abstract

Phishing costs Internet users billions of dollars per year. It refers to luring techniques used by identity thieves to fish for personal information in a pond of unsuspecting internet users. Phishers use spoofed e-mail, phishing software to steal personal information and financial account details such as usernames and passwords.The malicious links within the body of the message are designed to make it appear which go to the spoofed organization using organization's logos and other legitimate contents.Our project deals with methods for detecting phishing web sites by Natural Language Processing and by Machine learning techniques.

Machine learning is a powerful tool used to strive against phishing attacks. Discuss the methods used for detection of phishing websites based on NLTK libraries such as word segmentation, stemming and lemmatization(methods of trimming words down to the roots), and tokenization(for breaking phrases, sentences and paragraph into tokens help computer to better understand text) The fine-tuned parameters are useful in selecting the apt machine learning algorithm for classifying the phishing sites and benign sites. And classified them into 'Good' sites and 'Bad' sites.

Chapter 1

Introduction

Phishing costs Internet users billions of dollars per year. It refers to luring techniques used by identity thieves to fish for personal information in a pond of unsuspecting internet users. Phishers use spoofed e-mail, phishing software to steal personal information and financial account details such as usernames and passwords.. The project deals with methods for detecting phishing web sites by doing feature extraction of urls by Machine learning techniques and Natural Language Processing. The organization of Chapter 1 is as follows.

Section 1.1 describes Background of the project.Motivation of the project is represented in Section 1.2 Section 1.3 represents Problem statement of the project. Scope of the project is described in Section 1.4 Section 1.5 describes Objective of the project. The selection of life cycle model selection 1.6 describes. Section 1.7 shows the organisation of report.The summary is described in 1.8

1.1 Background

In recent years, advancements in Internet and cloud technologies have led to a significant increase in electronic trading in which consumers make online purchases and transactions. The growth leads to unauthorized access to users' sensitive information and damages the resources of an enterprise. Phishing content and gain the information. In terms of website interface and uniform resource locator (URL), most phishing webpages look identical to the actual webpages. Various strategies for detecting phishing websites, such as blacklist, heuristic, Etc., have been suggested. However, due to inefficient security technologies, is an exponential increase in the number of victims. The anonymous and uncontrollable framework of the Internet is more vulnerable to phishing attacks. Existing research works show the performance of the phishing detection system is limited. is a demand for an intelligent technique to protect users from the cyber-attacks. In study, the author proposed a URL detection technique based on machine learning approaches

1.2 Motivation

As discussed earlier phishing is a major threat to our society now a days and evolution of AI enabled us so many ways to deal with such kind of scams. So with the help of machine learning algorithms Build softwares can detect phishing websites. Remain a motivation for us to build a model can detect phishing websites and tell users about them.

1.3 Problem Defination

URLs sometimes known as "Web links" are the primary means by which users locate information in the Internet. Aim of the phishers is to acquire critical information like username, password and bank account details. Our aim is to derive classification models detect phishing urls using machine learning and natural language processing. In JuPyter environment. A general statement of the phishing website detection using machine learning can be formulated as follows Given a website into the input, detect whether it is phishing or not.

1.4 Scope

The Scope of detection of phishing website using machine learning are given below:-

- Around 83 percent of IT teams in Indian organizations said the number of phishing emails targeting the employees increased during 2020, according to the findings of a global survey titled 'Phishing Insights 2021' by Sophos, a cyber security company.
- The good news is, most organizations in India (98) have implemented cybersecurity awareness programme to combat phishing. Respondents said they use computer-based training programme (67), human-led training programme (60), and phishing simulations (51).
- So taking such a numbers and current scenario in consideration, is huge scope for phishing website detection with the help of latest technologies such as AI, Machine learning.

1.5 Objective

The objective of detection of phishing website using machine learning are given below:-

- To explain what phishing websites are? And how they are major threat to peoples.
- To collect phishing websites database and perform processing on them.
- After by applying various feature extraction techniques,fitting of the model with machine learning algorithm.
- Improving acccuracy of the model.
- Deplyoment of model of web-page and make it ready no use for end users,
- Users can enter the website on our website and check whether it is phishing or not.

1.6 Selection of Life Cycle Model for Development

The software development life cycle model selected for the project is the Waterfall Model. Waterfall approach was the first SDLC Model to be widely used in software engineering to ensure success of project. It was developed by Winston W. Royce in 1970. Classical waterfall model divides the life cycle into a set of phases. Our model considers one phase can be started after completion of the previous phase. Which is the output of one phase will be the input to the next phase. So the development process can be considered as a sequential flow in the waterfall.

SDLC models can be described along a spectrum of agile to iterative to sequential. Agile methodologies, such as XP and Scrum, focus on light-weight processes which allow for rapid changes along the development cycle. Iterative methodologies, such as Rational Unified Process and Dynamic Systems Development Method, focus on limited project scopes and expanding or improving products by multiple iterations.

In a Project management, a project can be defined both with a project life cycle (PLC) and an SDLC, during which slightly different activities occur. The project life cycle encompasses all the activities of the project, while the systems development life cycle focuses on realizing the product requirements. Figure 1.1 describes the Selection of life cycle Model.

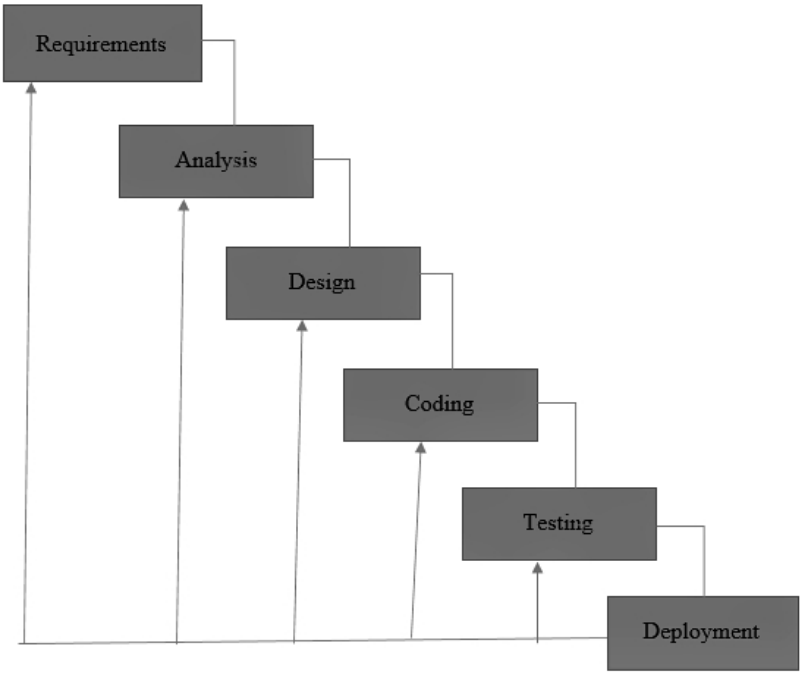


Figure 1.1: Selection of life cycle Model

1.7 Organization of Report

The report is described in following way.

Chapter 1 : Titled as Introduction describes the Background, Motivation, Problem Defination, Scope and Objectives.

Chapter 2 : Titled Project Planning and Management describes literature survey, feasibility study of proposed system.

Chapter 3 : Titled at System Analysis, which presents Requirement Collection and Identification, Software/Hardware Requirements, Functional and Non-Functiotal requirements and Software Requirement Specification.

Chapter 4 : Titled as System Design presents System Architecture data flow diagrams and the UML Diagrams

Chapter 5 : Titled as Conclusion concludes the minor project.

1.8 Summary

In this chapter Introduction is described. In the next chapter, The Project Planning and Management is presented.

Chapter 2

Project Planning and Management

Project planning is a procedural step in project management. It is the practice of initiating, planning, executing, controlling and closing the work team to achieve specific goals. Project planning and management is important because it ensures the right people do the right things, at the right time. It also ensures the proper project life cycle.

The organization of chapter 2 is as follows. Section 2.1 shows the Feasibility Study of the project.Risk Analysis of the project is represented in Section 2.2 and Project Scheduling is described in Section 2.3. Section 2.4 and 2.5 describe the Effort Allocation and Cost Estimation respectively. The Summary is mentioned in Section 2.6.

2.1 Feasibility Study

System Feasibility analysis divided into following analysis :
Technical feasibility study is an assessment of the practicality of a proposed project or system. A feasibility study aims to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained. A well-designed feasibility study should provide a historical background of the business or project, a description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations.Generally, feasibility studies precede technical development and project implementation. A feasibility study evaluates the project's potential for success; therefore, perceived objectivity is an important factor in the credibility of the study for potential investors and lending institutions. It must therefore be conducted with an objective, unbiased approach to provide information upon which decisions can be based.

2.1.1 Technical Feasibility

The assessment is based on an outline design of system requirements, to determine whether the company has the technical expertise to handle completion of the project. At the level, the concern is whether the proposal is both technically and legally feasible (assuming moderate cost). It is an evaluation of the hardware and software and how it meets the need of the proposed system. The website is developed using Html, Css, Flask, etc. Also all the other technologies used are capable of building such a platform and serve as well as maintain it for longer period time. All the required hardware and software are easily available in the market. Hence the portal is technically feasible.

2.1.2 Economical Feasibility

Describes how much time is available to build the new system, when it can be built, whether it interferes with normal business operations, type and amount of resources required, dependencies, and developmental procedures with company revenue prospectus. As the necessary hardware and the software are easily available in the market at low cost, the initial investment is the only cost incurred and does not need further enhancement. Hence it is economically feasible.

2.1.3 Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The operational feasibility assessment focuses on the degree to which the proposed development project fits in with the existing business environment and objectives with regard to development schedule, delivery date, corporate culture and existing business processes 0.7.5 2.1.3 Economic feasibility Describes how much time is available to build the new system, when it can be built, whether it interfere.

2.2 Risk Analysis

Risk Analysis and Management is a key project management practice to ensure the least number of surprises occur while project is underway. While never predict the future with certainty, apply a simple and streamlined risk management process to predict the uncertainties in the projects and minimize the occurrence or impact of such uncertainties. Which improves the chance of successful project completion and reduces the consequences of such risks. Risk analysis can be categorized as.

2.2.1 Risk Based on Dependencies

The system uses python 3.9 for development. Programming languages changes according current needs of the developers and for the future aspects of technology.Python will also receive time to time updates in future too. Various modules of python may change functionally after update. And issue can be avoided by creating a virtual environment.

2.2.2 Risk Based on Development Teams

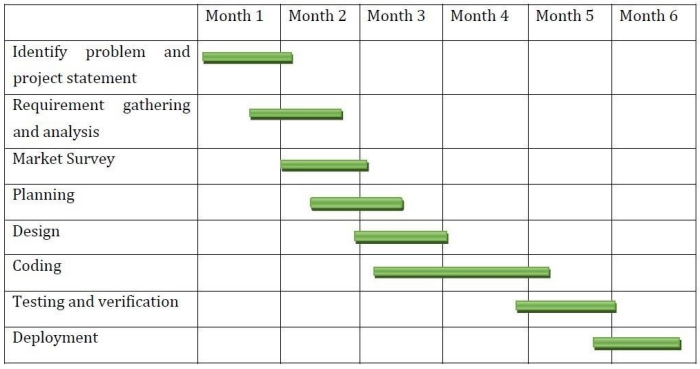
In today's scenario technology rapidly changes it may be possible the project team does not have required amount of knowledge regarding the project and the required technical proficiency to develop the system. Risk can be overcome by giving required training to developers. Bootcamps, seminars and doubt solving sessions can be conducted by the project lead to give required knowledge to the team. will resolve the risk based on development teams.

2.3 Project Scheduling

Generally, project scheduling can be stated as the estimated time required for any project from its time to beginning to the end of the project. In details, for every task,is a deadline because all the tasks for completion of project are planned earlier. So each task is scheduled to certain time limit.

In short, in project management, listing of project milestones, activities and all from starting ending date, are considered in the project scheduling. A schedule is generally used in the project planning and management of the project with kind of attributes as budget, task allocation and duration, resource allocation and all. Table 2.1 describes Gannt chart for the Project Scheduling.

Table 2.1: Gannt Chart For the Project Scheduling



2.4 Effort Allocation

Effort Allocation is necessary so every team member can give its best to the project. Project was divided into smaller module and task form, for simplification and easy understanding of project overall. Some modules include every team associate , presence to take advantage of team decision taking skills, and some task include some individual member to work on it with precision. can be divided the project into 6 modules.

1. Collection of dataset and gathering of information
2. Planning/Requirement Analysis
3. Study of technology used
4. Selection of Life cycle Model
5. Planning and Management
6. Analysis and Design UML

Table 2.2 describes the Work Contribution of the team members.

Table 2.2: Work Contribution

Sr. No	Work	Team associates			
		Rohit	Sakshi	Chetan	Vaishnavi
1	Collection of dataset	Yes		Yes	
2	Planning/requirement analysis	Yes	Yes		Yes
3	Study of technologies used	Yes	Yes		Yes
4	Selection of Life cycle model			Yes	Yes
5	Planning and management	Yes	Yes	Yes	Yes
6	Analysis & UML	Yes	Yes	Yes	

2.5 Cost Estimation

Cost Estimation is an important phase for any project. It predicts if the project investment is adequate or will shortage of capital. It presents the total cost required for development of project. Cost Estimation should be done before initiating the development to prevent loss of efforts and project failure during development. The cost estimation model i.e CoCoMo (Constructive Cost Model) is a regression model based on LOC, i.e number of Lines of Code.

It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality. It was proposed by Barry Boehm in 1970 and is based on the study of 63 projects, which make it one of the best-documented models. The key parameters which define the quality of any software products, which are also an outcome of the Cocomo are primarily Effort Schedule:

- **Effort:** Amount of labor which will be required to complete a task. It is measured in person months unils.
- **Schedule:**: Simply means the amount of time required for the completion of the job, which fs, of course, proportional to the effort put. It is measured in the units time such as weeks, months. The cost of any software project is calculaled by the formula,
  C = aLb.....equ(1).
  C = cost cf project, a = 1.4 (constant),
  b = 0.93 (consLant),
  t = size of code.
  For our project considering the number of lines of code to be 21000 based on the average number or lines of code for similar projects, able calculate the cost as follows:
  Cost of project
  C = 1.4*(3000)*0.93
  C = Rs 3,906.....from equ(1).

2.6 Summary

In this chapter , Project Planning and Management of project is described. In the next chapter , Analysis is presented.

Chapter 3

Analysis

The development of computer-based information system includes the system analysis phase which produces or enhances the data model which itself is to creating or enhancing a database. A number of different approaches to system analysis. The analysis is the process which is used to analyze, refine and scrutinize the gathered information of entities in order to make consistence and unambiguous information. Analysis activity provides a graphical view of the entire System.

System Analysis is the process of gathering and interpreting facts, diagnosing problems and using the facts to improve the system. System analysis chapter will show overall system analysis of the concept, description of the system, meaning of the system. System analysis is the study of sets of interacting entities, including computer system analysis.

- The Function of project are perform in Python and Open CV library.
- The Behaviours of the project must be Detections of Face.
- The analysis process move from implementation of Face Detection and Data Gathering

The organization of chapter is as follows. Section 3.1 presents Requirement Collection and Identification. Section 3.2 presents Hardware Requirements and Software Requirements. Section 3.3 presents Functional Requirements and Non Functional. Section 3.4 presents Summary.

3.1 Requirement Collection and Identification

Requirement collection is the process which is used to gather, analyze, and documentation and reviews the requirements. Requirements describe what the system will do in place of how. In practical application, most projects will involve some combination of various methods in

order to collect a full set of useful requirements. Requirements collection is initiated when the project need is first identified and the project "solution" is to be proposed. Requirements refinement continues after the project is "selected" and as the scope is defined, aligned and approved. Model will only require a website which has to be entered by user. So the model can predict the output.

3.1.1 Model Features

The product features are high level attributes of a software or product such as software performance, user-friendly interface, security portability, etc. The attributes are defined according to the product, in such a case, a software product. such as follows: Model will be able to achieve 97It will be able to predict whether the website is Good or Bad. Output will be displayed in less than 1 sec. It will also provide some useful information about phishing scams and how to be safe from them.

3.2 Hardware and Software Requirements

All computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended.
The various software and hardware requirements of the system can be summarized here:

3.2.1 Hardware Requirements

Hardware Requirements of our project is as follows.

1. 2GB RAM (minimum)
2. 100GB HDD (minimum)
3. Intel 1.66 GHz Processor Pentium 4 (minimum)
4. Internet Connectivity

3.2.2 Software Requirements

Software Requirements of our project is as follows.

1. Operating System: Windows 10 or higher

2. Python 3.6.0 or higher
3. visual studio code
4. Flask
5. HTML
6. Dataset of Phishing Website

3.3 Functional and Non-functional Requirement

The various functional and non functional requirements of the system can be summarized here:

3.3.1 User Interface

The platform's user interface has been designed to cater the users with simplicity and to be able to perform task with minimal efforts. The home screen offers a simple input box in which user have to enter website and click the predict button.

3.4 Summary

In this chapter , Analysis is described. In the next chapter, Design is presented.

Chapter 4

Design

Design is the activity to design and model the various component of software system. The system design provides the understanding and procedural details necessary for implementing the system. Design is helpful for a better understanding of the project. It contains the UML diagrams, data flow diagrams. UML is a modeling language which is used to document the object-oriented analysis and design..

The organization of the Chapter is as follows Section 4.1 describes the work- flow of model building of the project DFD of the project are represented in Section 4.2 Section 4.3 represents UML Diagrams such as use case diagram, work-flow diagram, bar graphs, confusion matrix of results, etc Finally, the Summary is described in last Section 4.4.

4.1 System Architecture

An architecture description is a formal description and representation of a system, organized in a way supports reasoning about the structures and behaviours of the system. it can consist of system components and the subsystems developed, will work together to implement the overall system. One can think of system architecture as a set of representations of an existing system. Figure 4.1 describes the System Architecture.

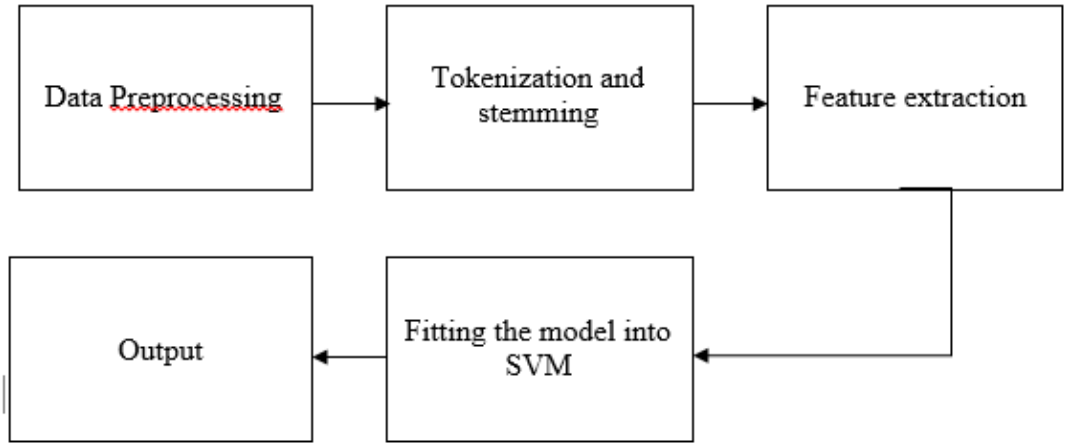


Figure 4.1: System Architecture

4.2 Data Flow Diagram

A Data Flow Diagram is a structured analysis and design tool can be used for flow- charting. A DFD is a network describes the flow of data and the processes change or transform the data throughout a system. The network is constructed by using a set of symbols do not imply any physical implementation. It has the purpose of clarifying system requirements and identifying major transformations. So it is the starting point of the design phase functionally decomposes the requirements specifications down to the lowest level of detail. DFD can be considered to an abstraction of the logic of an information-oriented or a processor oriented system flow-chart. For such reasons DFD's are often referred to as logical data ow diagrams. Data flow diagram (DFD), also called as Bubble chart is a graphical technique, which is used to represent information ow, and transformers are applied when data moves from input to output.

4.2.1 DFD Level 0

It is also known as context diagram. It's designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represent the entire system as single bubble with input and output data indicated by incoming/outgoing arrows. Figure 4.2 describes DFD Level 0.

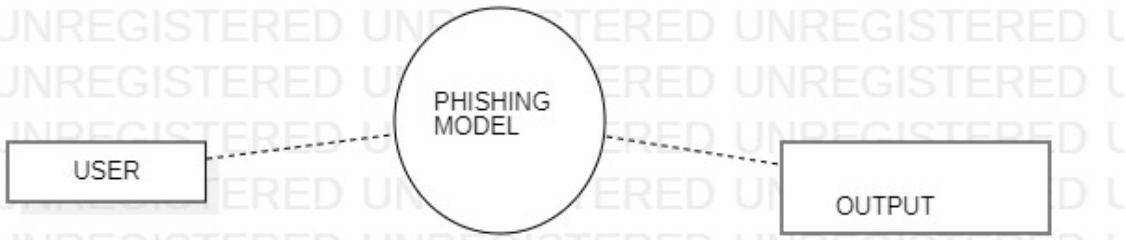


Figure 4.2: DFD Level 0

4.2.2 DFD Level 1

In 1-level DFD, context diagram is decomposed into multiple bubbles/processes. Such level highlight the main functions of the system and breakdown the high level process of 0-level DFD into sub processes. Processes in diagram 0 (with a whole number) can be exploded further to represent details of the processing activities.
Figure 4.3 describes DFD Level 1.

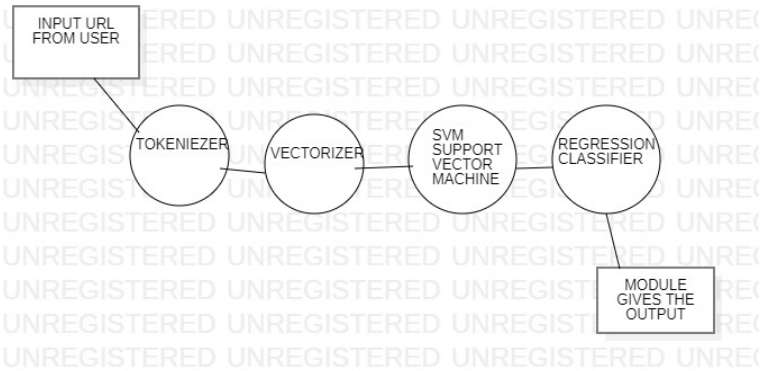


Figure 4.3: DFD Level 1 Diagram

4.2.3 Work flow and Feature Extraction

Firstly checked whether is any null value present in the dataset or not and if so removed it.
• Use of RegexpTokenizer : with the help of tokenize.regexp() module, able to extract the tokens from strings by using regular expression with RegexpTokenizer() method.
• Use of PorterStemmer : stemmer is used to produce morphological variants of a root/base

word. Stemming is desirable as it may reduce redundancy as most of the time the word stem and derived words mean the same. And then after stemming joining of words is done.
• Feature Extraction using TfidfVectorizer: It's the main part of natural language processing . TF-IDF is an abbreviation for Term Frequency Inverse Document Frequency. which is very common algorithm to transform text into a meaningful representation of numbers which is used to fit machine algorithm for prediction.
• Splitting the data: Train-test-split is used to split the data into training dataset and testing dataset.
• Algorithms used: Support Vector Machine (SVM) is used. The SVM performs classification by finding the hyper plane maximizes the margin between two classes. The vectors define the hyper plane are the support vectors.

4.3 UML Diagrams

Following are UML diagrams required for the project.

4.3.1 Use Case Diagram

Use case diagrams consists of actors , use cases and relationships. The diagram is used to model the system subsystem of an application. A single use case diagram captures a particular functionality of a system. Hence to model the entire system, a number of use case diagrams are used. The diagram shows the step by step procedure how the working is done and model has built. Each step is connected. Figure 4.4 describes the Use Case diagram.

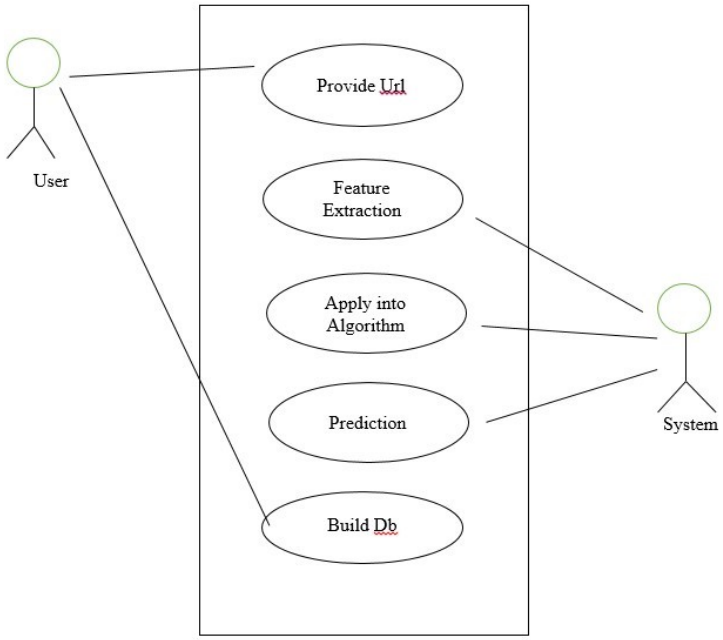


Figure 4.4: Use case Diagram

4.3.2 Sequence Diagram

A diagram shows the existence of Objects over time , and the Messages pass between such Objects over time to carry out some behaviour. A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which interactions take place. Also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. Figure 4.5 describes the Sequence Diagram.

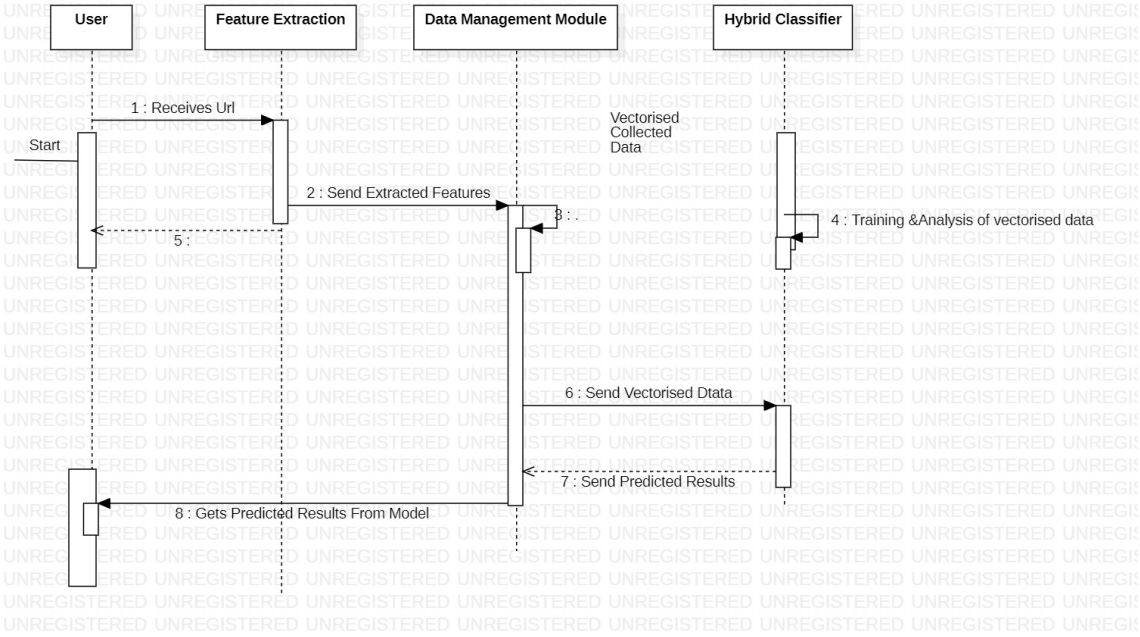


Figure 4.5: Sequence Diagram

4.3.3 Class Diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling, translating the models into programming code. Class diagrams can also be used for data modeling.The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed. Figure 4.6 describes the Class Diagram.

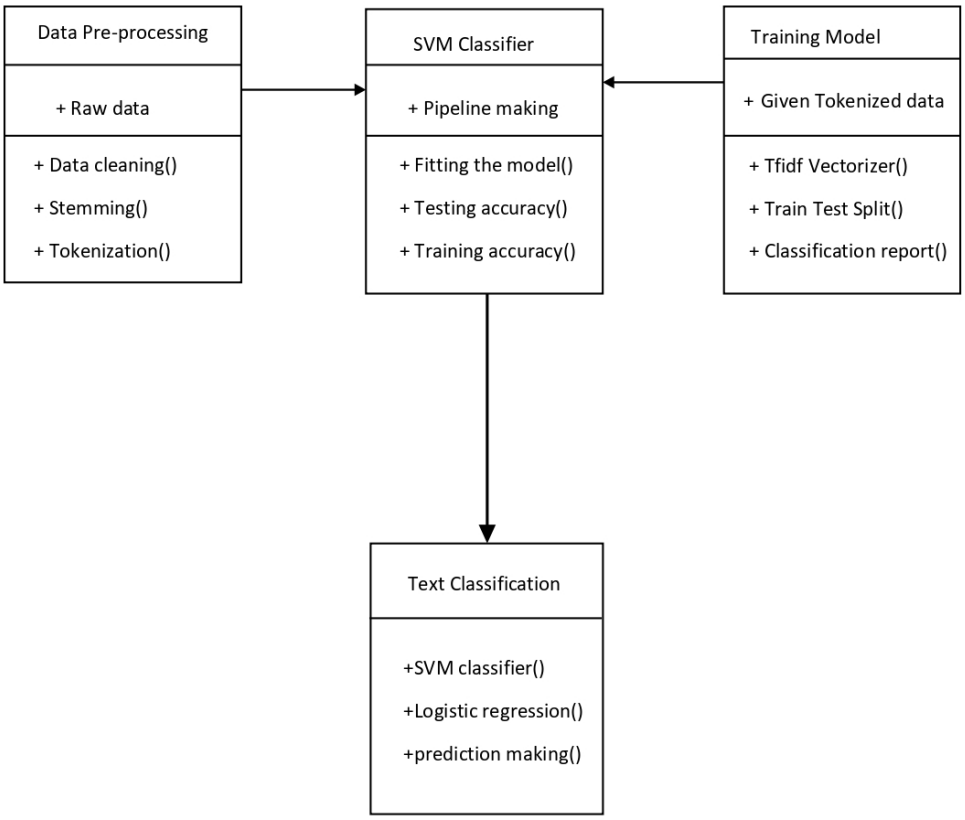


Figure 4.6: Class Diagram

4.3.4 Component Diagram

Component diagram is UML structure diagram which shows component and dependencies between the component. Model diagrams allow to show different views of a system, for example, as multi-layered (aka multi-tiered) application - multi-layered application model. Following component diagram shows the components of the proposed system, hierarchy of components and relationships with each other. Figure 4.7 describes Component Diagram.

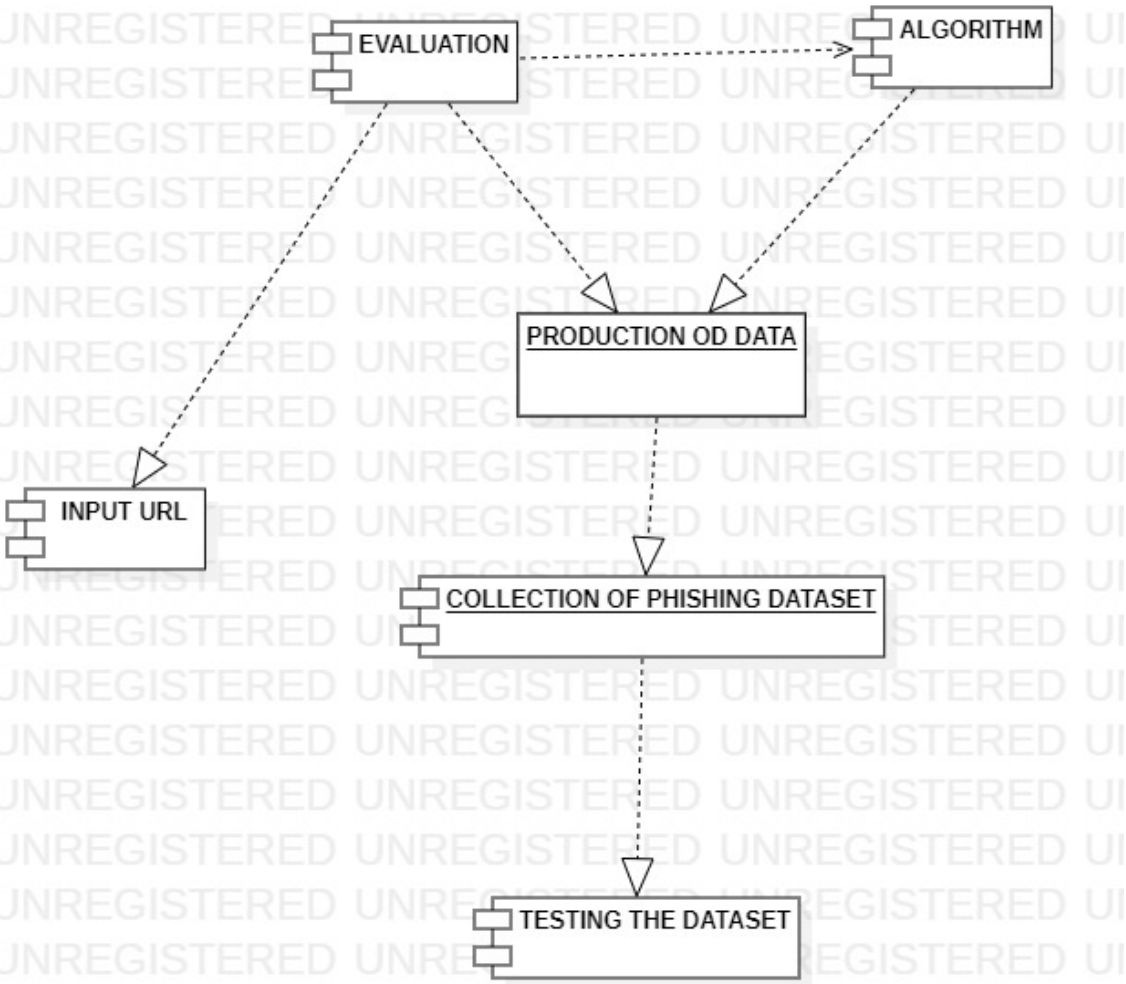


Figure 4.7: Component Diagram

4.3.5 Deployment Diagram

A UML deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams is a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often be used to model the static deployment view of a system (topology of the hardware). Figure 4.8 describes the Deployment Diagram.

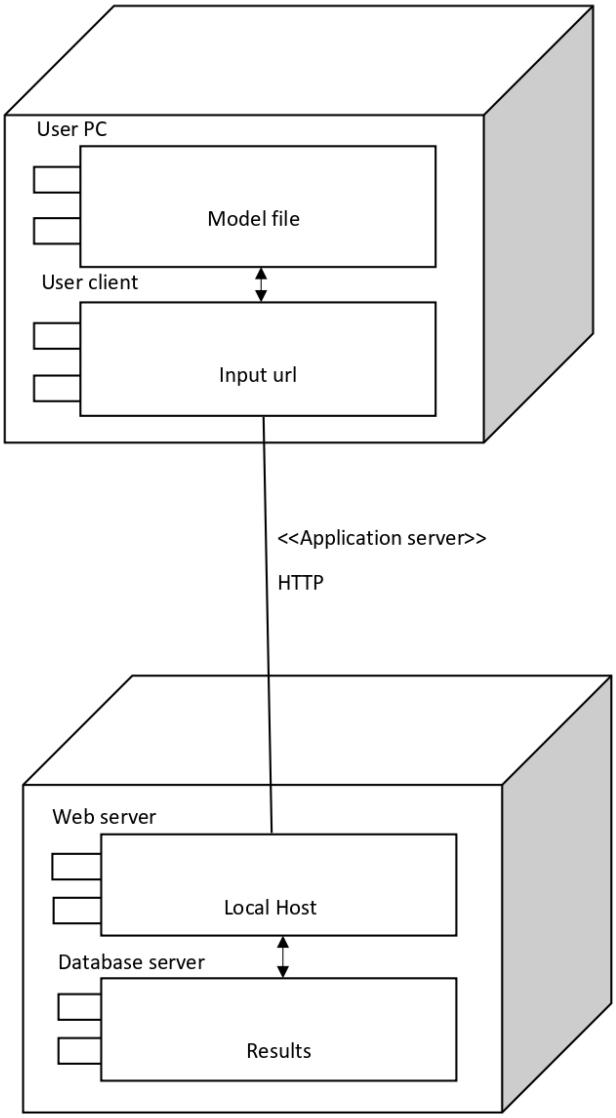


Figure 4.8: Deployment Diagram

4.4 Summary

In this chapter , Design is described. In the next chapter, Coding and Implementation is presented.

Chapter 5

Coding and Implementation

Implementation phase is longest and most important phase in software development. When the designing of the software is completed, then a group of developers starts coding of the design using a programming language. The interface of the software and all its internal working according to design phase is implemented in implementation phase.

The chapter mainly contains following sections: Section 5.1 describes the Algorithms / Steps, Hardware and Software Requirements are described in the section 5.2 . Section 5.3 describes the Project Modules. Summary of the Chapter described in the section 5.4.

5.1 Algorithms / Steps .

Support Vector Machine (SVM) is a supervised machine learning algorithms can be used for both classification or regression challenges.
SVM algorithm finds the closest point of the lines from both the classes. And points are called support vectors.
1. Importing some useful libraries.
2. Data preprocessing.
3. Tokenizing the strings.
4. Stemming.
5. Vectorization.
6. feature extraction.
7. Fitting the model with Support Vector Machine.
8. Making the pipeline.
9. Loading the model with pickle.
10. predict output.

5.2 Software and Hardware Requirements .

The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product.

- WINDOWS 7 or higher .
- Python 3.6.0 or higher .
- Visual Studio Code .
- Flask.
- HTML .
- Dataset of Phishing Website.
- 2GB RAM (minimum).
- 100GB HDD (minimum).
- Intel 1.66 GHz Processor Pentium 4 (minimum) .
- Internet Connectivity .

5.3 Modules In Project .

In Application many modules which develop each module separately. When all modules are ready ultimately integrated all the modules into one application. The section briefly describe all the modules and the functionality of such a modules.

Pandas.

Numpy.

Sklearn.

Matplotlib.

Nltk.

Pickle.

5.4 Summary

In this chapter , Coding and Implementation is described. In the next chapter, Testing is presented.

Chapter 6

Testing

Testing goes side by side with the implementation is aimed at ensuring the system works accurately before the live operation is performed. The common view of testing held by the user is to ensure they are no errors in a program. Testing usually means the process of executing a program with explicit intention of handling errors. It depends on the process and the associated stakeholders of the project.

The chapter mainly contains following sections:- Section 6.1 describes the Black Box Testing. And the White Box Testing is described in the Section 6.2. Summery of the chapter is described in the section 6.3.

6.1 Black Box Testing

Black Box testing also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. Tests can be functional or non-functional, though usually functional. The method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see. such method attempts to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in data structures or external database access
- Behavior or performance errors
- Initialization and termination errors

6.2 White Box Testing

White Box Testing is also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing. It is a soft- ware

testing method in which the internal structure, design, implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a 41 system. The method is named so because the software program, in the eyes of the tester, is like a white or transparent box; inside which one clearly sees.

6.3 Summary

In this chapter , Testing is described. In the next chapter Result is presented.

Chapter 7

Result

The results section is a section containing a description about the main findings of a research, whereas the discussion section interprets the results for readers and provides the significance of the finding.

Writing the results and discussion as separate sections allows to focus first on what results which obtained and set out clearly what happened in the experiments and/or investigations without worrying about the implication. Section 7.1 decribes the Result and Summery is described in the section 7.2.

7.1 Results

To summarize, how phishing is a huge threat to the security and safety of the web and how phishing detection is an important problem domain.

- Scikit-learn tool has been used to import Machine learning algorithms. Each classifier is trained using training set and testing set is used to evaluate performance of classifiers
- Performance of classifiers has been evaluated by calculating classifier's accuracy score.
- Here got model accuracy of 9

So Support Vector Machine is ideal to use.

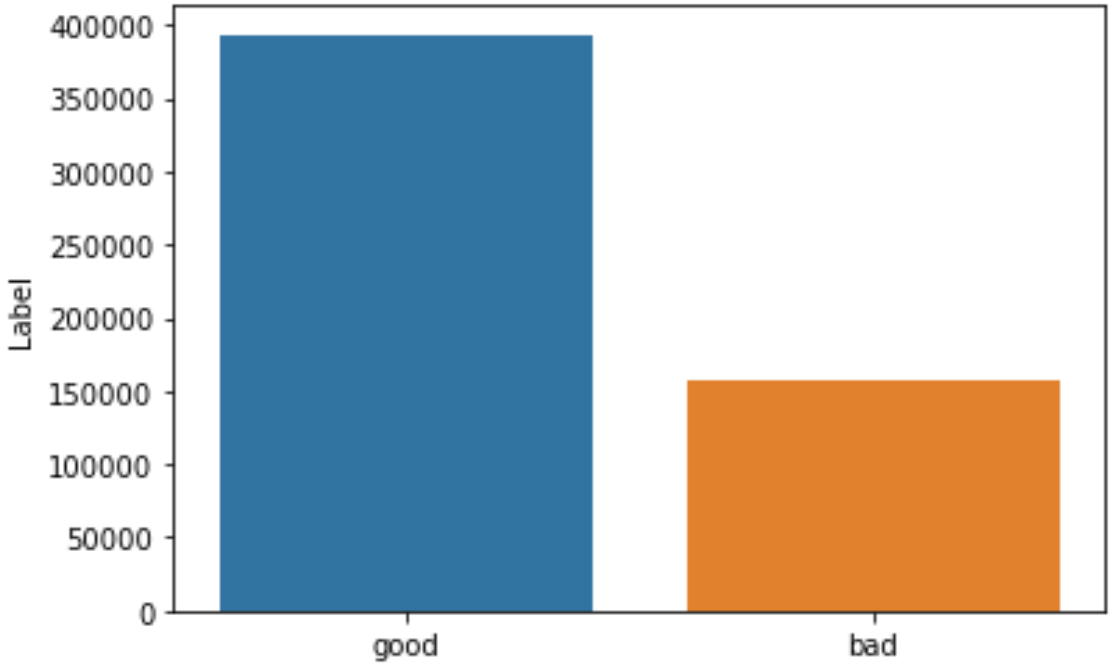
To be tested SVM machine learning algorithm on the 'Phishing Websites Dataset and reviewed the results. Built a Chrome extension for detecting phishing web pages. The extension allows easy deployment of our phishing detection model to end users. and detected phishing websites using Support Vector Machine with and accuracy of 97.94

For future enhancements, Intend to build the phishing detection system as a scalable web service which will incorporate online learning so new phishing attack patterns can easily be learned and improve the accuracy of our models with better feature extraction. Table 7.1 describes the Bibliography and Table 7.2 describes the Bar Graph.

Table 7.1: Bibliography

Classifier	Accuracy Rate%	Error Rate%
SVM	Training-99% Testing-97%	3%

Table 7.2: Bar Graph



7.2 Summary

In this chapter , Result is described. In the next chapter, Conclusion is presented.

Chapter 8

Conclusion

A good conclusion will summarize final thoughts and main points, combining all the relevant information with an emotional appeal for a final statement resonates with readers. section 8.1 of the chapter describes the Conclusion and the Future Scope. Section 8.1 describes the Conclusion and Summery is described in the Section 8.2.
To summarize, how phishing is a huge threat to the security and safety of the web and how phishing detection is an important problem domain. Tested SVM machine learning algorithm on the 'Phishing Websites Dataset and reviewed the results. As built a Chrome extension for detecting phishing web pages. The extension allows easy deployment of our phishing detection model to end users. And detected phishing websites using Support Vector Machine with and accuracy of 97.94.

Future Scope For future enhancements, Intend to build the phishing detection system as a scalable web service which will incorporate online learning so new phishing attack patterns can easily be learned.

Bibliography

[1] Detection of phishing websites using machine learning international journal of engineering research and technology (ijert) ijertv10is050235 (this work is licensed under a creative commons attribution 4.0 international license.). Published by : 10 Issue 05, May-2021 Intbrnationjl License.) Puelished by : Issue 05, May-2021 .

[2] Mustafa AYDIN. Feature extraction and classification phishing websites based on url. ieee. 2015.

[3] Ciza Thomas Joby James, Sandhya L. s detection of phishing websites using machine learning techniques. 2013 international conference on control communication and computing (iccc). 2013.

The References are cited in [1],[3], [2]

Index