

**CAR PRICE PREDICTION**

**PROJECT REPORT**

***Submitted by***

# KAMALESH KUMAR- 412519104055

# JHAYA KRISHNA-412519104153

***in partial fulfillment for the award of the***

***degree of***

# BACHELOR OF ENGINEERING

**IN**

### COMPUTER SCIENCE AND ENGINEERING

**SRI SAI RAM ENGINEERING COLLEGE**

### (An Autonomous Institution; Affiliated to Anna University, Chennai -600 025)

**ANNA UNIVERSITY : CHENNAI 600 025**

**JULY 2022**

**SRI SAI RAM ENGINEERING COLLEGE**

**(An Autonomous Institution; Affiliated to Anna University, Chennai -600 025)**

**ANNA UNIVERSITY, CHENNAI – 600 025 BONAFIDE CERTIFICATE**

Certified that this project report **“CAR PRICE PREDICTION”**

is the bonafide work of “ **KAMALESH KUMAR(412519104055) ,**

**JHAYA KRISHNA(412519104153)”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

Dr.B.LATHA M.E,PhD. MRS. . RAMYA N

**HEAD OF THE DEPARTMENT ( SUPERVISOR)**

Department of Computer Science and Engineering

Sri Sai Ram Engineering College, (Autonomous) West Tambaram, Chennai- 600 044

Department of Computer Science and Engineering

Sri Sai Ram Engineering College, (Autonomous) West Tambaram, Chennai- 600 044

Submitted for the project viva-voce examination held on

### INTERNAL EXAMINER EXTERNAL EXAMINER

**ACKNOWLEDGEMENT**

We would like to express our profound gratitude to our beloved and respected founder, (Late) Thiru.MJF. Ln. LEO MUTHU , Sri Sairam Engineering College for having provided plenty of resources.

We express our gratitude to our CEO Shri. J. SAI PRAKASH LEO MUTHU for his constant encouragement in completing this project.

## We extend solemn thanks to our Principal Dr. K PORKUMARAN for having given us his whole-hearted encouragement for completing this project.

We are indebted to our Head of the Department Dr.B.LATHA , for her support during the entire course of this project.

## We extend our gratitude to our project supervisor Mrs. KAVITHA V , for her support and supervision during the entire course of the project.

Our special thanks to our Project Coordinator Mr J.JEGAN AMARNADTH for his kind support in bringing out this project successfully.

## Finally, we would like to thank all the staff members of the Department of Computer Science and Engineering, our parents, friends and all others who contributed directly and indirectly for the successful completion our project.

|  |  |  |
| --- | --- | --- |
|  | **TABLE OF CONTENTS** |  |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT** | **iii** |
|  | **LIST OF FIGURES** | **iv** |
|  | **LIST OF ABBREVATION** | **v** |
|  | **REQUIREMENTS** | **vi** |
|  | **INSTALLING LIBRARIES** | **vi** |
|  | **IMPORTING DATA** | **vi** |
| **1.** | **INTRODUCTION** | **1** |
|  | 1.1 Objectives | 3 |
|  | 1.2 Scope | 3 |
| **2.** | **LITERATURE SURVEY** | **4** |
| **3.** | **METHODOLOGY** | **6** |

* 1. Machine learning classifiers and methods

to detect phishing website. 6

* 1. [Proposed system](#_bookmark0) [8](#_bookmark0)
  2. Lexical feature analysis 10

1. FEATURE EXTRACTION 12
   1. [Long URL](#_bookmark1) [12](#_bookmark1)
   2. [URL’s having “@” symbol](#_bookmark2) [12](#_bookmark2)
   3. [Redirecting using “//”](#_bookmark3) [12](#_bookmark3)

i

* 1. Adding prefix or suffix separated by (-) to the domain 13
  2. [Sub-domains and multi sub-domains](#_bookmark4) [13](#_bookmark4)

1. SYSTEM DESIGN 16
   1. [Use case Diagram](#_bookmark6) [16](#_bookmark6)
   2. [Activity Diagram](#_bookmark5) [16](#_bookmark5)
   3. [Sequence Diagram](#_bookmark7) [17](#_bookmark7)
   4. [Class Diagram](#_bookmark8) [18](#_bookmark8)
2. MODEL BUILDING 19
   1. [Building a model](#_bookmark9) [19](#_bookmark9)
   2. [Source code](#_bookmark10) [20](#_bookmark10)
   3. [Result](#_bookmark11) [21](#_bookmark11)
3. IMPLEMENTATION AND TESTING 22
   1. [Hardware requirements](#_bookmark12) [22](#_bookmark12)
   2. [Software requirements](#_bookmark13) [22](#_bookmark13)
   3. [Implementation steps](#_bookmark14) [22](#_bookmark14)
   4. [Flowchart](#_bookmark15) [24](#_bookmark15)
4. RESULTS 25
5. CONCLUSIONS 28

[REFERENCES 29](#_bookmark16)

ii

**ABSTRACT**

## In this project, we investigate the application of supervised machine learning techniques to predict the price of used cars in India. Different techniques like linear regression analysis, lasso regression have been used to make the predictions. The predictions are then evaluated and compared in order to find those which provide the best performances. All the three methods provided comparable performance. Regression Algorithms are used because they provide us with continuous value as an output and not a categorized value. Because of which it will be possible to predict the actual price a car rather

## than the price range of a car. Earlier problem includes a process where a seller decides a price randomly and buyer has no idea about the car and it’s value in the present day scenario. In fact, seller also has no idea about the car’s

## existing value or the price he should be selling the car at. To overcome this problem we have developed a model which will be highly effective. We will compare the performance of various algorithms like Linear Regression, Ridge Regression, Lasso Regression, Tree Regressor and choose the best out of it. Depending on various parameters we will determine the price of the car

Keywords: Car Price Prediction, supervised learning, linear regression, LASSO regression, classification

|  |  |  |
| --- | --- | --- |
|  | **LIST OF FIGURES** |  |
| **FIGURE NO** | **TITLE** | **PAGE NO** |
| 3.1 | PROPOSED SYSTEM BLOCK DIAGRAM | 9 |
| 4.1 | UNSTRUCTURED DATA | 12 |
| 4.2 | LOADING THE DATA IN OUR PROGRAM | 13 |
| 5.1 | ACTIVITY DIAGRAM | 17 |
| 5.2 | ACTIVITY DIAGRAM FOR COMPOSE, |  |
|  | SEND AND RECEIVE MAIL | 17 |
| 5.3 | ACTIVITY DIAGRAM FOR MAIL SYSTEM | 18 |
| 5.4 | SEQUENCE DIAGRAM FOR COMPOSE, |  |
|  | SEND AND RECEIVE MAIL | 18 |
| 5.5 | CLASS DIAGRAM | 19 |
| 7.1 | IMPORTING THE REQUIRED PACKAGES | 25 |
| 7.2 | LOADING THE DATASET | 25 |
| 7.3 | SPLITTING UP THE DATASET FOR |  |
|  | TESTING AND TRAINING | 26 |
| 7.4 | CALCULATION OF THE ACCURACY OF |  |
|  | RANDOM FOREST CLASSIFIERS | 26 |
| 7.5 | CALCULATION OF THE ACCURACY OF |  |
|  | NAÏVE BAYES CLASSIFIERS | 26 |
| 7.6 | COMPARISON OF ACCURACY OF |  |
|  | VARIOUS CLASSIFIER | 27 |
| 7.7 | RELATIONSHIP BETWEEN CLASSIFIERS | 27 |

### LIST OF ABBREVATIONS

|  |  |
| --- | --- |
| **ML** | MACHINE LEARNING |
| **RNN** | RECURRENT NEURAL NETWORK |
| **NLP** | NATURAL LANGUAGE PROCESSING |
| **DT** | DECISION TREE ALGORITHM |
| **RF** | RANDOM FOREST ALGORITHM |
| **GBM** | GRADIENT BOOSTING ALGORITHM |
| **GLM** | GENERALIZED LINEAR MODEL |
| **GAM** | GENERALIZED ADDICTIVE MODEL |
| **FTP** | FILE TRANSFER PROTOCOL |
| **IP** | INTERNET PROTOCOL |
| **HTTP** | HYPERTEXT TRANSPORT PROTOCOL |

**REQUIREMENTS**

SOFTWARE REQUIREMENTS:

• Python 3.0

• Pycharm

• Jupyter Notebook

### 

### HARDWARE REQUIREMENTS:

### • Operating system-windows 7,8,10,11

### • Processor –dual core 2.4 GHz

### • Ram – 4gb or more

### INSTALLING LIBRARIES

In this first step, we have to import the most common libraries used in python for machine learning such as

* Pandas
* Numpy
* Seaborn
* Matplotlib

### CHAPTER 1 INTRODUCTION

Today, the transportation industry is considered to be one of the backbones of the economy. Automobiles are referred to as the "Industry of Industries" in developed nations. According to industry professionals, the UAE's automotive industry has seen remarkable growth. Besides being the fastest-growing nation in the automobile industry, it represents its global presence. In Dubai, like most other countries, cars are gaining a great deal of popularity among the local population and the ex-pat community who work in the country. There are used cars for sale in the UAE of all makes and models, even cars from well-known brands (Rizvi, 2019). UAE's auto industry is experiencing constant growth, registered at 27%, with a total industry volume (TIV) of 310,403 cars. Approximately 1.49 million units were sold within the Gulf Cooperation Council (GCC). Compared to the global market, the Gulf Cooperation Council countries are growing at 10% in 2021 (Research, 2020). So far, the market in the UAE has grown by 19%. It is thus the world's largest market in terms of growth rate. Almost everyone wants their own car these days, but because of factors like affordability or economic conditions, many prefer to opt for pre-owned cars. Accurately predicting used car prices requires expert knowledge due to the nature of their dependence on a variety of factors and features. Used car prices are not constant in the market, both buyers and sellers need an intelligent system that will allow them to predict the correct price efficiently. In this intelligent system, the most difficult problem is the collection of the dataset which contains all important elements like the manufacturing year of the car, its gas type, its condition, miles driven, horsepower, doors, number of times a car has been painted, customer reviews, the weight of the car, etc. It is clear that the price of the product is affected by many factors, but unfortunately, information about these features is not always readily available. Since this project primarily focuses on the Dubai market, the benchmark dataset containing all key features is scraped. It is necessary to pre-process and transform collected data in the proper format prior to feeding it directly to the data mining model. As a first step, the dataset was statistically analyzed and plotted. Missing, duplicated, and null values were identified and dealt with. Features were chosen and extracted using

correlation matrices. To build an efficient model, the most correlated features were retained, and others were discarded. This prediction problem can be considered a regression problem since it belongs to the supervised learning domain. Three Regressor known as random forest, linear regression, and bagging regression were trained and compared. A random forest Regressor outperformed all others in this project, so it was chosen as the main algorithm model.

Statement of problem

The research objective of this study is to predict used cars prices in India using data mining techniques, by scraping data from websites that sell used cars, and analysing the different aspects and factors that lead to the actual used car price valuation. To enable consumers to know the actual worth of their car or desired car, by simply providing the program with a set of attributes from the desired car to predict the car price. The purpose of this study is to understand and evaluate used car prices in the India, and to develop a strategy that utilizes data mining techniques to predict used car prices.

Predicting the resale value of a car is not a simple task. It is trite knowledge that the value of used cars depends on a number of factors. The most important ones are usually the age of the car, its make (and model), the origin of the car (the original country of the manufacturer), its mileage (the number of kilometers it has run) and its horsepower. Due to rising fuel prices, fuel economy is also of prime importance. Unfortunately, in practice, most people do not know exactly how much fuel their car consumes for each km driven. Other factors such as the type of fuel it uses, the interior style, the braking system, acceleration, the volume of its cylinders (measured in cc), safety index, its size, number of doors, paint colour, weight of the car, consumer reviews, prestigious awards won by the car manufacturer, its physical state, whether it is a sports car, whether it has cruise control, whether it is automatic or manual transmission, whether it belonged to an individual or a company and other options such as air conditioner, sound system, power steering, cosmic wheels, GPS navigator all may influence the price as well. Some special factors which buyers attach importance in Mauritius is the local of previous owners, whether the car had been involved in serious accidents and whether it is a lady-driven car. The look and feel of the car certainly contributes a lot to the price.

# OBJECTIVES

# ▪ To make people decide whether a used car is worth the posted price when they see listings in online platforms.

# ▪ Based on the existing data, the aim is to use machine learning algorithms to develop models for predicting used car prices.

# ▪ It would help people to determine the best price by comparing the prices against different car vendors.

# ▪ The problem is solved using above mentioned machine learning techniques/models

* 1. **SCOPE**
     + To develop an unsupervised deep learning method to predict the car price.
     + The study can be extended in order to generate an outcome for a larger data using an application and protect the privacy of an individual.

**CHAPTER 2 LITERATURE SURVEY**

**PAPER 2.1:** PHISH-SAFE: URL Features-Based Phishing Detection System Using Machine Learning.

**Authors:** [Ankit Kumar Jain](https://link.springer.com/chapter/10.1007/978-981-10-8536-9_44#auth-Ankit_Kumar-Jain) **&** B.B.Gupta

**Abstract:**

Today, phishing is one of the most serious cyber-security threat in which attackers steal sensitive information such as personal identification number(PIN), credit card details, login, password, etc., from Internet users. In this paper, we proposed a machine learning based anti-phishing system (i.e., named as PHISH- SAFE) based on Uniform Resource Locator (URL) features. To evaluate the performance of our proposed system, we have taken 14 features from URL to detect a website as a phishing or non-phishing. The proposed system is trained using more than 33,000 phishing and legitimate URLs with SVM and Naïve Bayes classifiers.

Our experiment results show more than 90% accuracy in detecting phishing websites using SVM classifier.

**PAPER 2.2:** Detection of URL based phishing attacks using machine learning

**Authors:** Ms. Sophiya. Shikalgar, Dr. S. D. Sawarkar, Mrs. Swati Narwane

**Abstract:**

A fraud effort to get sensitive and personal information like password, username, and bank details like credit / debit card details by masking as a reliable organization in electronic communication. It most of the time redirects the users to similar looking website as legitimate website. The phishing website will appear same as the legitimate website and directs the user to a page to enter personal details of the user on the fake website. The system administration is very important these days as any failure can be detected and solved instantly. The system administration also need to define rules and set firewall settings to avoid phishing attacks through URL. Researchers have been studying various machine learning algorithm in lines to predict and avoid phishing attacks. Through machine learning algorithms one can improve the accuracy of the prediction. The machine learning, no one algorithm works best for every problem, and it’s especially relevant for supervised learning. Using a single machine learning algorithm will give us good accuracy to predict the phishing attacks but to get better accuracy we

need something more. The proposed system predicts the URL based phishing attacks with maximum accuracy. We shall talk about various machine learning, the algorithm which can help in decision making and prediction. We shall use more than one algorithm to get better accuracy of prediction. The algorithms namely the Naive Bayes and Random forest are used in the proposed system to detect URL based phishing attacks. The hybrid algorithm approach by combining

**PAPER 2.3:** An Ideal Approach for Detection and Prevention of Phishing Attacks

**Authors:** Narendra.M & Chaithali shah

### Abstract:

In this paper, we propose a phishing detection and prevention approach combining URL-based and Webpage similarity based detection. URL-based phishing detection involves extraction of actual URL (to which the website is actually directed) and the visual URL (which is visible to the user). LinkGuard Algorithm is used to analyze the two URLs and finally depending on the result produced by the algorithm the procedure proceeds to the next phase. If phishing is not detected or Phishing possibility is predicted in URL-based detection, the algorithm proceeds to the visual similarity based detection. A novel technique to visually compare a suspicious page with the legitimate one is presented.

**PAPER 2.4:** Phishing website detection based on effective machine learning approach

**Authors:** Lokesh.G & Gowtham.B

**Abstract:**

Phishing a form of cyber-attack, which has an adverse effect on people where the user is directed to fake websites and duped to reveal their sensitive and personal information which includes passwords of accounts, bank details, atm pin-card details etc. Hence protecting sensitive information from malwares or web phishing is difficult. Machine learning is a study of data analysis and scientific study of algorithms, which has shown results in recent times in opposing phishing pages when distinguished with visualization, legal solutions, including awareness workshops and classic anti-phishing approaches. This paper examines the applicability of ML techniques in identifying phishing attacks and report their positives and negatives. In specific, there are many ML algorithms that have been explored to declare the appropriate choice that serve as anti-phishing tools. We have designed a Phishing Classiﬁcation system which extracts features that are meant to defeat common phishing detection approaches. We also make use of numeric representation along with the comparative study of classical machine learning techniques like Random Forest, K nearest neighbours, Decision Tree, Linear SVC classifier, One class SVM classifier and wrapper-based features selection which contains the metadata of URLs and use the information to

determine if a website is legitimate or not.

**PAPER 2.5:** Machine Learning and Deep Learning Based Phishing Websites Detection: The Current Gaps and Next Directions

**Authors:** Kibreab Adane & Berhanu Beyene

### Abstract:

There are many phishing websites detection techniques in literature, namely white-listing, black-listing, visual-similarity, heuristic-based, and others.

However, detecting zero-hour or newly designed phishing website attacks is an inherent property of machine learning and deep

learning techniques. By considering a promising solution of machine learning and deep learningtechniques, researchers have made a great deal of effort to tackle the this problem, which persists due to attackers constantly devising novel strategies to exploit vulnerability or gaps in existing anti-phishing measures. In this study, an extensive effort has been made to rigorously review recent studies focusing on Machine Learning and Deep Learning Based Phishing Websites Detection to excavate the root cause of the aforementioned problems and offer suitable solutions. The study followed the significant criterion to search, download, and screen relevant studies, then to evaluate criterion-based selected studies. The findings show that significant research gaps are available in the rigorously reviewed studies. These gaps are mainlyrelated to imbalanced dataset usage, improper selection of dataset source(s), the unjustified reason for using specific train-test dataset split ratio, scientific disputes on website features inclusion and exclusion, lack of universal consensus on phishing website lifespans and on what is defining a small dataset size, and run-time analysis issues.

**PAPER 2.6:** Detection of phishing websites using an efficient feature-based machine learning framework.

**Authors:**Royhu Srinivas rao & sathvik

**Abstract:** In this paper, we propose a novel classification model, based on heuristic features that are extracted from URL, source code, and third-party services to overcome the disadvantages of existing anti-phishing techniques. Our model has been evaluated using eight different machine learning algorithms and out of which, the Random Forest (RF) algorithm performed thebest with an accuracy of 99.31%. The experiments were repeated with different (orthogonal and oblique) random forest classifiers to find the best classifier for the phishing website detection. Principal component analysis Random Forest (PCA-RF) performed the best out of all oblique Random Forests (oRFs) with anaccuracy of 99.55%. We have also tested our model with the third-party-based features and without third-party-based features to determine the effectiveness of third-party services in the classification of suspicious websites. We also compared our results with the baseline models (CANTINA and CANTINA+).

Our proposed technique outperformed these methods and also detected zero-day

**CHAPTER 3**

**METHODOLOGY**

In this section, we tend to study the various classifiers employed in system finding out to predict phishing. We will conjointly provide proof for our projected methodology to discover phishing internet sites and attacks.

In section 3.1 we shall provide proof for various classifiers and techniques which may be employed to check the phishing and legit web site. In section 3.2 we can justify our projected system.

Further, in section 3.3, we will explain about the feature extraction of the URL. We will utilize the extracted features for training and testing of the data sets.

### MACHINE LEARNING CLASSIFIERS AND METHODS TO DETECT THE PHISHING WEBSITE.

Detecting and identifying Phishing Websites is simply a complicated and dynamic task. Machine studying has been extensively utilized in many regions to create solutions. The phishing attacks can be carried out in many methods which include email, website, malware, SMS, etc. In this work, we concentrate on finding out website phishing (URL), which is finished by utilizing the Hybrid Algorithm Approach. Hybrid Algorithms Approach is a mixture of various classifiers algorithms running collectively which gives an amazing prediction rate and improves the accuracy of the system.

Contingent upon the application and furthermore the idea of the dataset utilized

we will utilize any grouping calculations referenced. As their square measure completely different applications, we have a tendency to cannot differentiate that of the algorithms square measure superior or not. Every of classifiers has its own manner of operating and classification.

Let us discuss each of them in details.

* **Naive Bayes Classifier:** This classifier can likewise be known as a Generative Learning Model. The characterization here depends on Bayes' Theorem, it expects autonomous indicators. In straightforward words, this classifier will expect that the presence of explicit highlights in a class isn't identified with the presence of some other component. On the off chance that there is any reliance among the highlights of one another or on the closeness of various features, these will be taken into consideration as a self-governing commitment to the likelihood of the yield. This arrangement calculations are a lot of value to huge datasets and are exceptionally simple to utilize.
* **Random Forest:** This classification set of rules is like gathering a learning approach of type. The regression and other tasks, work with the aid of forming a collection of decision bushes at training records level and for the duration of the output of the class, which can be the mode of class or prediction regression for character timber. This classifier accuracy for choice trees practice of overfitting the training facts set.
  + **XGBOOST: XGBoost** is an optimized distributed gradient boosting library designed to be highly **efficient**, **flexible** and **portable**. It implements machine learning algorithms under the Gradient boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way. The same code runs on major distributed environment (Hadoop, SGE, MPI) and can solve problems beyond billions of examples.

**GRADIENT BOOSTING CLASIFIER:**

GB builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions. In each stage n\_classes\_ regression trees are fit on the negative gradient of the loss function, e.g. binary or multiclass log loss. Binary classification is a special case where only a single regression tree is induced.

### DECISION TREE:

A tree has many analogies in real life, and turns out that it has influenced a wide area of **machine learning**, covering both **classification and regression**. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, its also widely used in machine learning, which will be the main focus of this article.

* **Support vector machine (SVM):** This is likewise one of the classification algorithms which is directed and is anything but difficult to utilize. In this calculation each point which is an information thing is plotted in a dimensional space, this space is additionally called an n-dimensional plane, where the 'n' speaks to the number of highlights of the information.

Once the version is trained it's terribly essential to gauge the classifier that we have a tendency to shall use and validate its capability. currently, within the higher than section, we've visible all the advantages and drawbacks of all the out their classifiers. Hence, we have a tendency to advocate employing a few classifiers that are we have a tendency to area unit ready to use an associate mixture of classifiers to enhance the accuracy equally of prediction. we have a tendency to shall value each of the classifiers and use Naive Bayes and Random forest, by the usage of the mixture explicit during this section we have a tendency to shall improve the accuracy and build it higher. once applying the classification, the outcomes area unit generated and also the URLs area unit categorized into phishing and valid URLs. The Phishing URLs area unit blacklisted within the info and also the valid maybe a white list in database.

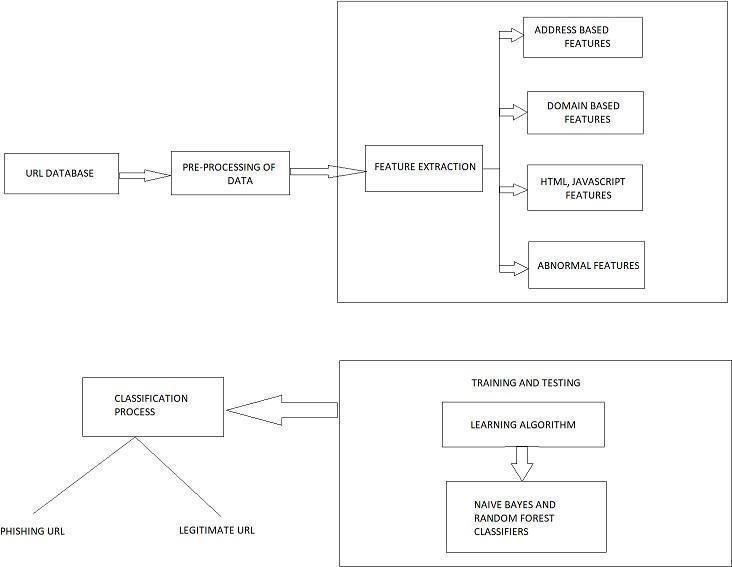
# PROPOSED SYSTEM

The dataset of phishing and legitimate URLs is provided within the application which is then pre-processed so that the facts are within the working format for analysis. The functions have round 30 traits of phishing websites that have used to distinguish them from legitimate ones. Each category has its very own traits of phishing attributes and values are defined. The specified traits are extracted for every URL and legitimate stages of inputs are identified. These values are then

assigned to every phishing internet site risk. The phishing properties esteems are spoken to with double no 0 and 1 which appears the characteristic is present or not.

In the education phase, we have to use the categorized statistics in which there are samples which include phish regions and legitimate areas. In the event that we attempt this, at that point type will never again be a difficulty for recognizing the phishing space. We should always use samples whose instructions are recognized to us, which shows the samples whom we label as phishing ought to be detected simplest as phishing. Similarly, the samples that are categorized as legitimate are detected as a valid URL.

The dataset which is for use for machine gaining knowledge of has to truly include these features. There is such a massive quantity of gadget learning algorithms and every set of rules has its own working mechanism which we’ve got already seen in the previous chapter. The triumphing system makes use of everybody of the acceptable system getting to know algorithms for the detection of phishing URL and predicts its accuracy.

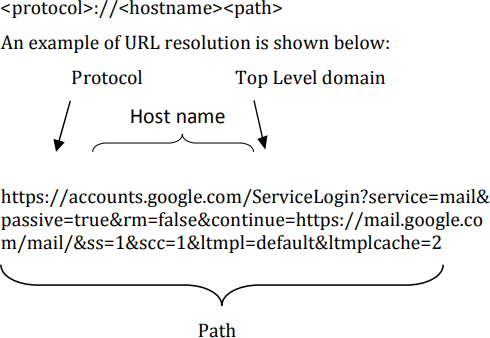


**Fig.3.1:** Proposed System block diagram

# LEXICAL FEATURES ANALYSIS

Lexical functions are the textual properties of the URL itself, now not the content of the page it factors too. URLs are human-readable textual content strings that can be parsed at some stage in a popular way by customer programs.

Through a multistep process, browsers will translate every URL into commands that discover the server web hosting the region and determine in which the location or useful resource is positioned on it. To facilitate this AI process, URLs have the subsequent widespread syntax.



The portion of the URL indicates which network protocol should be accustomed fetch the requested resource. The foremost common protocols in use are Hypertext Transport Protocol or HTTP (http), HTTP with Transport Layer Security (https), and File Transfer Protocol (ftp). The is that the identifier for the net server on the web. Sometimes it’s a machine-readable Internet Protocol (IP) address, but more often especially from the user’s perspective it is a humanreadable name. The of a URL is analogous to the trail name of a file on a neighborhood computer. The procedure which is used in our work to separate the lexical highlights from the URL list is as per the following: The URLs of genuine sites, gathered from alexa.com and dmoz.org, are composed into the scratchpad and along these lines, the record is spared inside the PC.

# LONG URL:

**CHAPTER 4 FEATURE EXTRACTION**

Long URL is used to shroud the Suspicious Part. If the length of the URL is bigger than or comparable to 54 characters then the URL is assigned to be phished.

# URL’s HAVING “@” SYMBOL:

Using “@” symbol within the URL leads the browser to ignore everything preceding the “@” symbol thus making it phished. Also, the real address often follows the “@” symbol.

IF {URL Having @ Symbol→ Phishing URL Otherwise→ Legitimate}.

# REDIRECTING USING “//”:

The existence of “//” within the URL path implies that the user is going to be redirected to a different website. An example of such URL’s is:

[“h](http://www.legitimate.com/)t[tp://www.legitimate.com//](http://www.legitimate.com/)[http://www.phishing.com](http://www.phishing.com/)”.

We investigate the condition where the "//" appears. We find that if the URL begins with "HTTP", which means the "//" ought to show up inside the 6th position. Regardless, if the URL uses "HTTPS" by then the "//" must show up in the seventh position.

IF {The Position of the Last Occurrence of "//" in the URL > 7→ Phishing URL Otherwise→ Legitimate}.

# ADDING PREFIX OR SUFFIX SEPERATED BY (-) TO THE DOMAIN:

The dash symbol isn’t employed in legitimate URLs. Phishers tend to use prefixes or suffixes separated by (-) to the URL so that the users feel that they are visiting a secure webpage

For example, [http://www.Confirme-paypal.com/.](http://www.confirme-paypal.com/)

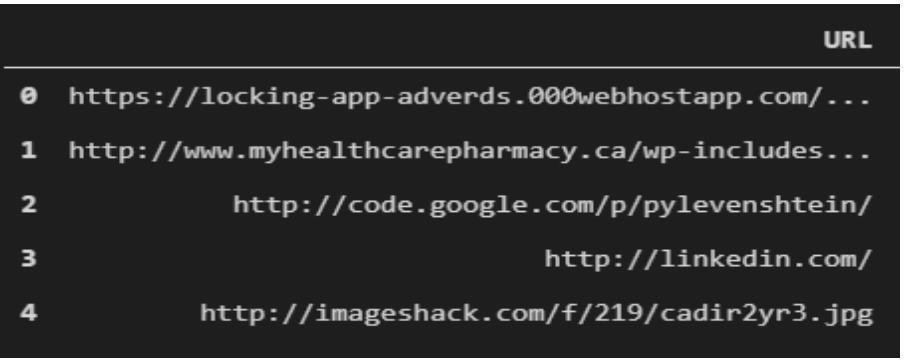
IF {Domain has (−) Symbol → Phishing URL Otherwise → Legitimate}.

# SUB-DOMAINS AND MULTI SUB-DOMAINS:

The legitimate URL link has two dots within the URL since we will ignore typing [“w](http://www/)w[w.”.](http://www/) If the number of dots is comparable to three then the website is evaluated as “Suspicious”.

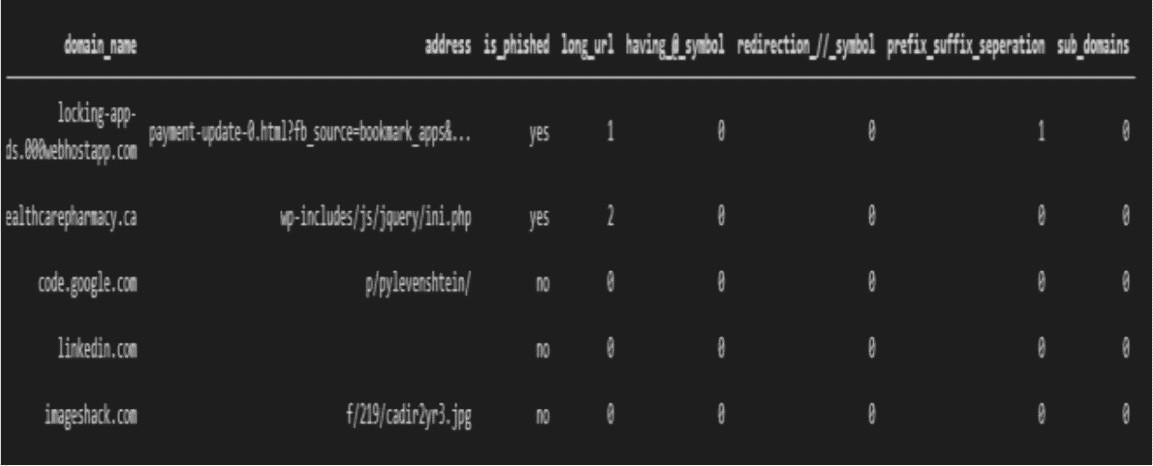
However, if the dots are larger than three, then it will be categorized as “Phishy”.

**Data set:** The information of URLs is gotten from the Phishtank site, where Phishtank is an enemy of the phishing site. It contains 2905 URLs which is in an unstructured structure. Our primary target is to identify whether the URL is phishing or authentic dependent on the highlights removed.



**Fig.4.1:** Unstructured Data

In Pre-processing, we have performed the component extraction where The URLs are transmitted to the element extractor, which concentrates values through the predefined URL-based highlights. The highlights have allocated twofold qualities 0 and 1 which demonstrates that component is available or not as appeared in the figure beneath. A structured dataset is given to the classifiers.



**Fig.4.2:** Loading the data in our program

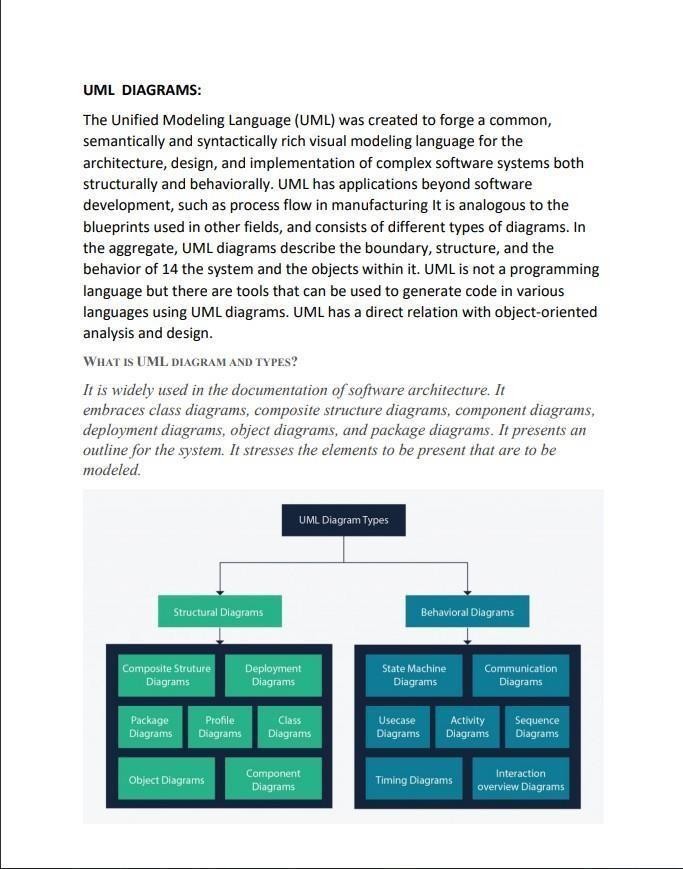
**Table -1:** URL Features

### URL Features:

Referring to Table 1., features from 1 to 4 are associated with suspicious Characters such as ‗@ ‘and ‗// ‘rarely appear in a URL. At present, to keep a client from distinguishing that a site isn't authentic, phishing destinations ordinarily conceal the essential area; the URLs of these phishing locales have curiously long subdomains.

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Feature name** | **Description** |
| **1** | IP address | Whether Domain is in the form of an IP address |
| **2** | Length of URL | Length of URL |
| **3** | Suspicious character | Whether URL has  ‗@ ‘, ‗//‘ |
| **4** | Prefix and suffix | Whether URL has ‘-‘ |
| **5** | HTTPS protocol | Whether URL use https. |
| **6** | Phishing words in URL | URL has phishing terms |
| **7** | Number of ‘.’ | Number of dots '.’ in URL |

### CHAPTER 5



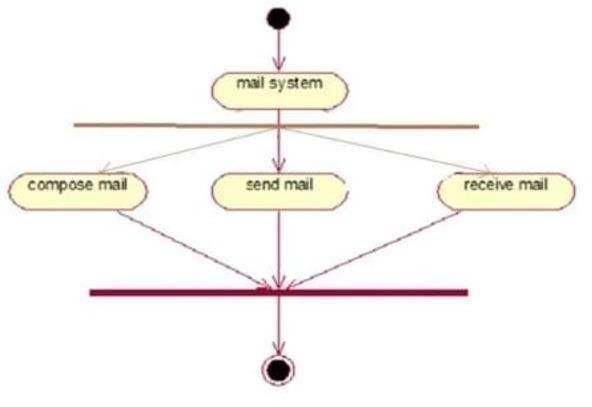
**SYSTEM DESIGN**

### USE CASE DIAGRAM

**Fig.5.1:** Use case Diagram

* + - A use case is a list of steps that define interaction between an actor (a human who interacts with the system or an external system) and the system itself. Use case diagrams depict the specifications of a use case and model the functional units of a system.
    - These diagrams help development teams understand the requirements of their system, including the role of human interaction therein and the differences between various use cases.

### ACTIVITY DIAGRAM



**Fig.5.2:** Activity Diagram for Compose, Send and Receive Mail

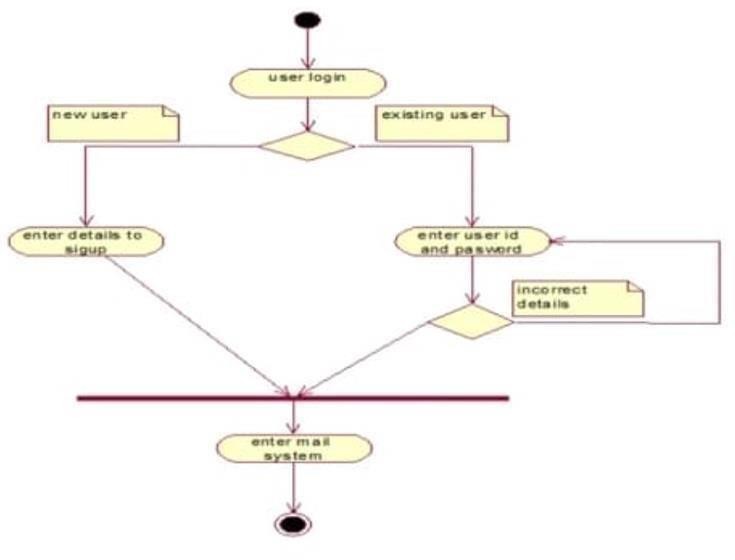
* Activity diagrams show the procedural flow of control between class objects, along with organizational processes like business workflows.
* These diagram are made of specialized shapes, then connected with arrows. The notation set for activity diagrams is similar to those for state diagrams.

**What is purpose of activity diagram?**

An activity diagram shows business and software processes as a progression of actions. These actions can be carried out by people, software components or computers. Activity diagrams are used to describe business processes and use cases as well as to document the implementation of system processes.

**Benefits of activity diagrams**

* Demonstrate the logic of an algorithm.
* Describe the steps performed in a UML use case.
* Illustrate a business process or workflow between users and the system.
* Simplify and improve any process by clarifying complicated use cases.



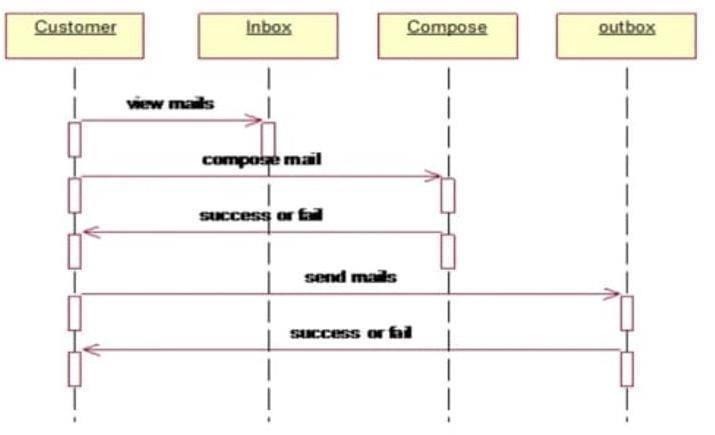
**Fig.5.3:** Activity Diagram for Mail System

### SEQUENCE DIAGRAM

* + - Sequence diagrams, also known as event diagrams or event scenarios,illustrate how processes interact with each other by showing calls between different objects in a sequence.
    - These diagrams have two dimensions: vertical and horizontal. The vertical lines show the sequence of messages and calls in chronological order, and the horizontal elements show object instances where the messages are relayed.
    - A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works

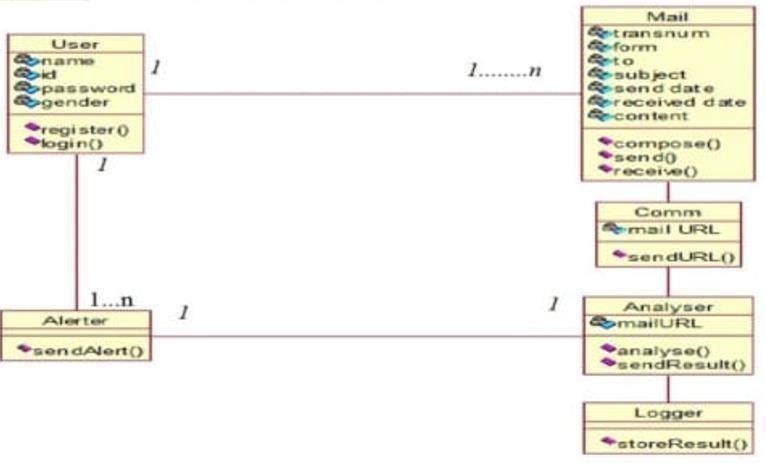
together. These diagrams are used by software developers and business professionals to understand requirements for a new systemor to document an existing process.

* + - A sequence diagram shows the sequence of messages passed between objects.Sequence diagrams can also show the control structures between objects. Forexample, lifelines in a sequence diagram for a banking scenario can represent a customer, bank teller, or bank manager.



**Fig.5.4:** Sequence Diagram for Compose, Send and Receive Mail

### CLASS DIAGRAM



**Fig.5.5:** Class Diagram

* Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.
* Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of objectoriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.
* Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

### CHAPTER 6 MODEL BUILDING

### 6.1 BUILDING A MODEL :

This is most important phase which includes model building for Phishing website detection.

Procedure of Proposed Methodology

Step1: Import required libraries, Import dataset. Step2: Pre-process data to remove missing data.

Step3: Perform percentage split of 80% to divide dataset as Training set and 20% to Test set, both the data sets are used in various parts of the process of the detection and usage of the software.

The splitting process is done for the enhancement of the outcome and easy usage of the required data.

Step4: Select the machine learning algorithm i.e. Decision Tree (DT), Random Forest (RF), Gradient Boosting (GBM), Generalized Linear Model (GLM) and Generalized Additive Model (GAM) all these algorithms are used and tested for accuracy metric and the result is generated.

We have taken accuracy as our metric for the generation of the output and later usage of the result of the process.

Step5: Build the classifier model for the mentioned machine learning algorithm based on training set, this model is used in the core of the problem functioning and result optimization.

The classifier model is built in such a way that it optimizes the selected machine learning model and uses it accordingly.

Step6: Test the Classifier model for the mentioned machine learning algorithm based on test set, Tested model is used for the futhur functioning of the process and algorithm authentication.

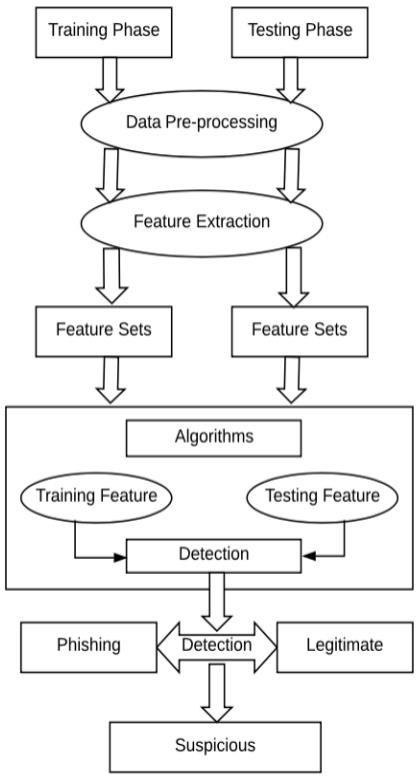
Testing is very much essential for the smooth functioning of the model and the given selected algorithm thus it’s done in an authentic way of test process.

Step7: Perform Comparison Evaluation of the experimental performance results obtained for each classifier, This process gives the output for the best approach and algorithm that is to be used in the entire process.

Comparing and extracting the results obtained from the experimental classifier algorithms is done in this step.

Step8: After analysing based on various measures conclude the best performing algorithm, this performance generate the process modules and generate the required outcome and the process defined for the outcome which has to be generated regularly through out the process.

The main outcome of the modules responsible here are given in this step and this process can be reused and redone multiple times as long as the URL is tested for the correctness and proximity.



### CHAPTER 7 IMPLEMENTATION AND TESTING

This segment gives data about the execution condition and illuminates the real strides for the usage of the dataset to show signs of improvement exactness to anticipate phishing by utilizing various classifiers mixes.

### HARDWARE REQUIREMENTS

The following hardware was used for the implementation of the system:

* + - 4 GB RAM
    - 10GB HDD
    - Intel 1.66 GHz Processor Pentium 4

### SOFTWARE REQUIREMENTS

The following software was used for the implementation of the system:

* + - Windows 7
    - Python 3.6.0
    - Visual Studio Code

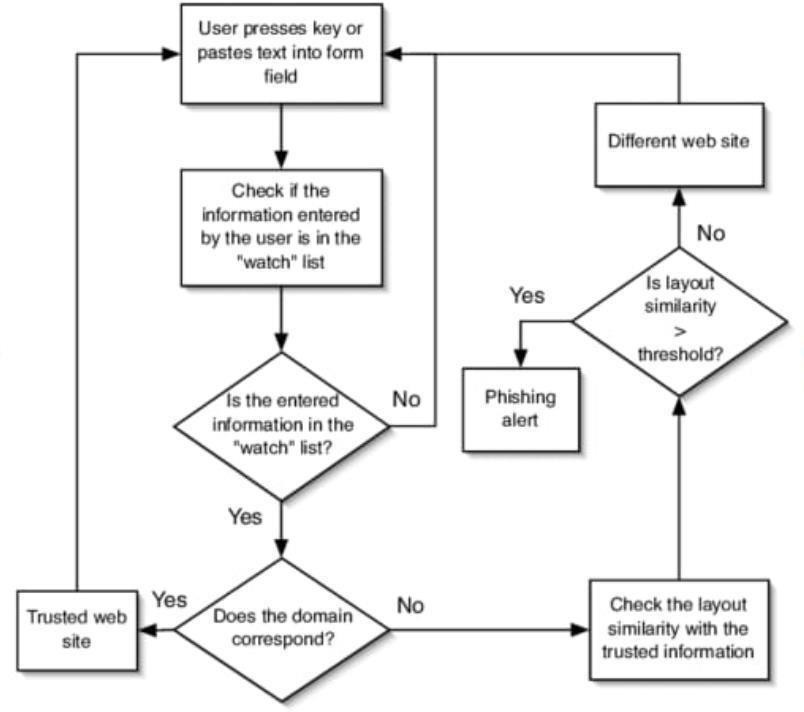
### IMPLEMENTATION STEPS

In this section, we will talk about the means which were actualized while doing the examination. We will provide a piece of evidence for the stepwise method

accustomed to split the knowledge and to foresee the phishing. We have utilized unstructured information that comprises just URLs. There are 2905 URLs gotten from the Phishtank site which comprises of both phishing and genuine URL where the majority of the URLs got are phishing.

1. We have collected unstructured data of URLs from Phishtank website.
2. In pre-processing, feature generation is done where nine features are generated from unstructured data. These features are length of an URL, URL has HTTP, URL has suspicious character, prefix/suffix, number of dots, number of slashes, URL has phishing term, length of subdomain, URL contains IP address.
3. After this, an organized dataset is made in which each detail incorporates the paired (0,1) which is then passed to the various classifiers.
4. Next, we train the three unique classifiers and analyse their presentation based on exactness three classifiers utilized are SVM, Naive Bayes and Random Forest.
5. At that point, the classifier identifies the given URL dependent on the preparation information that is if the site is phishing it prompts the user that the website is phished and if genuine, it prompts the user that the website is legitimate.
6. We look at the exactness of various classifiers and discovered Random Forest as the best classifiers which gives the most extreme precision.

### FLOW CHART

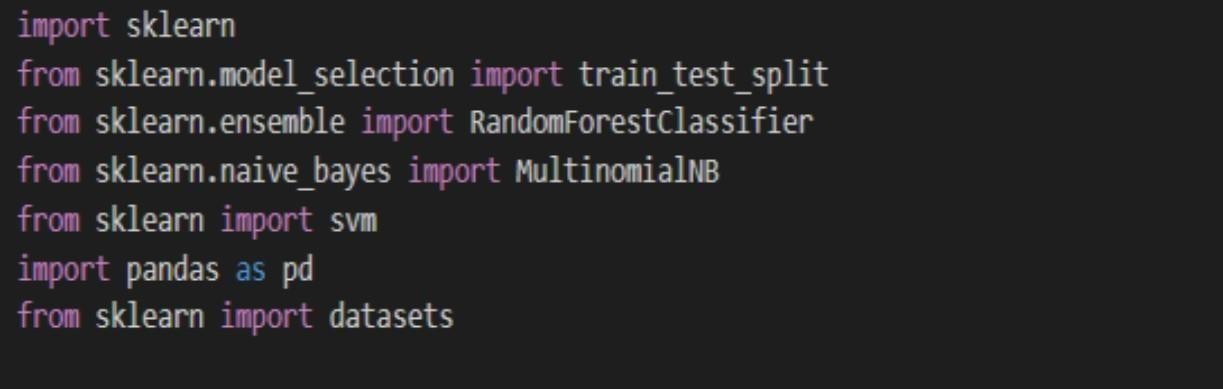


# CHAPTER 8 RESULTS

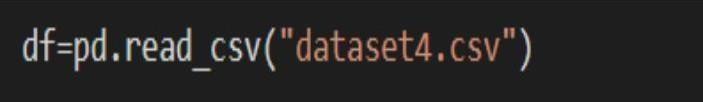
We have efficiently calculated the consequences of numerous classifiers which might be SVM, Naïve Bayes, Random Forest.

On comparison of resultant values, we chose to put into effect the Random Forest classifier in our datasets. Steps to obtain the accuracy of various classifiers:

* + Initially, we import all the packages which can be implemented in our project.

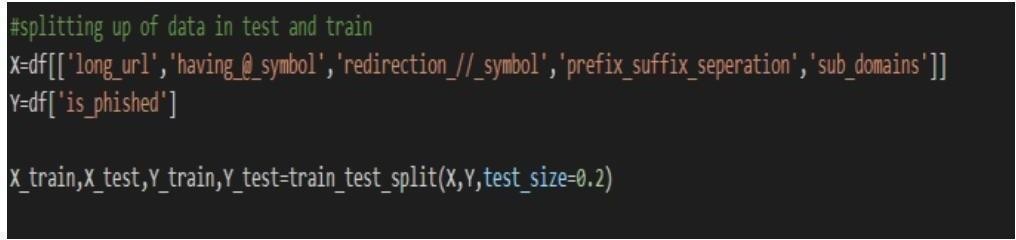


**Fig.7.1:** Importing the required packages

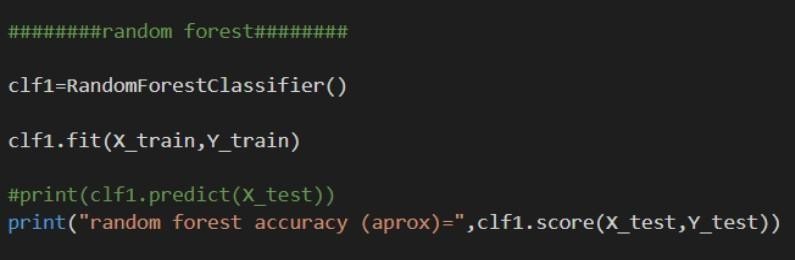
* + We will load the data sets for testing and training.

**Fig.7.2:** Loading the data set

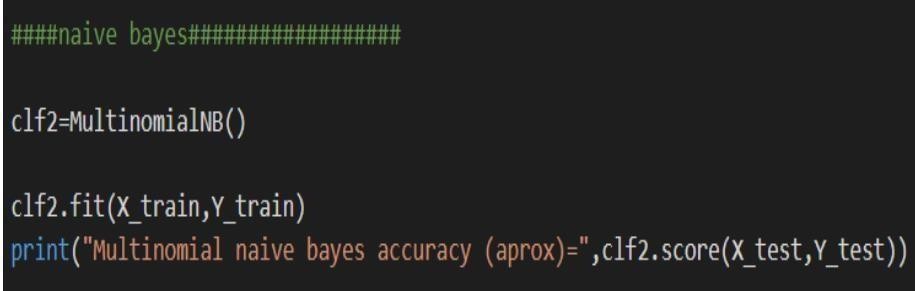
* + Now, we will do splitting up of data for training and testing. We will use 20% of data set for testing.



**Fig.7.3:** Splitting up of Data Set for testing and training

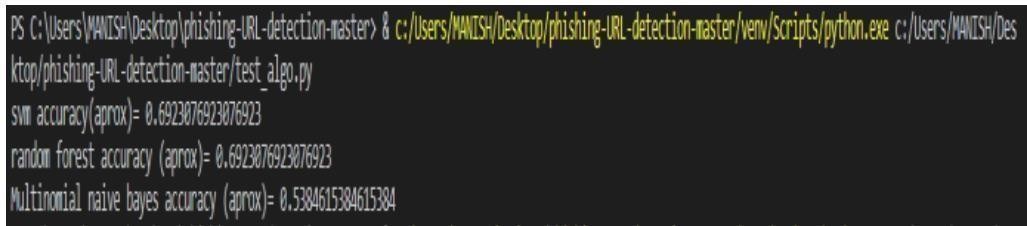
* + We will calculate the accuracy of Random Forest classifier.

**Fig.7.4:** Calculation of the accuracy of Random Forest classifier

* + We will calculate the accuracy of Naïve Bayes classifier.

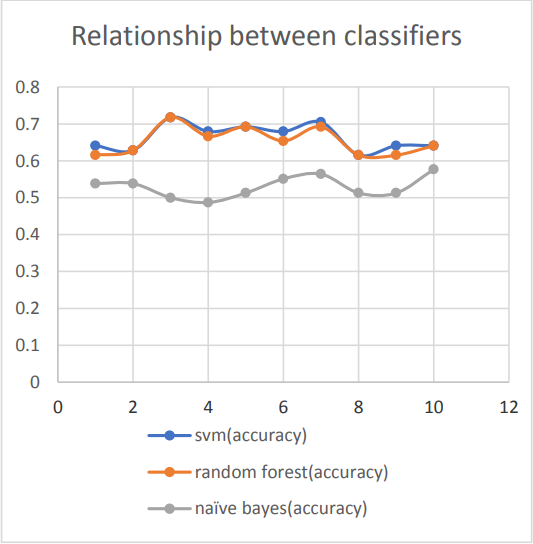
**Fig.7.5:** Calculation of the accuracy of Naïve Baye’s classifier

26

* + Now, we will compare the results obtained after calculating the accuracy of various classifiers.

**Fig.7.6:** Comparison of accuracy of various classifiers

* + Upon comparison, we found that accuracy of Random Forest Algorithm is highest and is considered best for our data set.



**Fig.7.7:** Relationship between the classifiers

# CHAPTER 9 CONCLUSIONS

It is discovered that phishing assaults are unbelievably essential and it's significant for us to invite an instrument to distinguish it. As fundamental and private data of the client is spilled through phishing sites, it turns out to be progressively basic to require care of this issue. This issue is handily understood by utilizing any of the AI calculations with the classifier. We have just got classifiers that give a decent expectation pace of phishing additionally, yet after our overview that it'll be smarter to utilize a half breed approach for the forecast and further improvement of the exactness expectation pace of phishing sites. We've seen that the current framework gives less precision so we proposed a fresh out of the box new phishing strategy that utilizes URL based highlights and furthermore, we created classifiers through a few AI calculations.

The main findings of our preliminary work include:

* + Phishing URLs and domains show some characteristics that are different from other URLs and domains.
  + Phishing URLs and domain names have altogether different lengths contrasted with different URLs and domain names inside the Internet.
  + A large number of the phishing URLs contained the name of the brand they focused on.

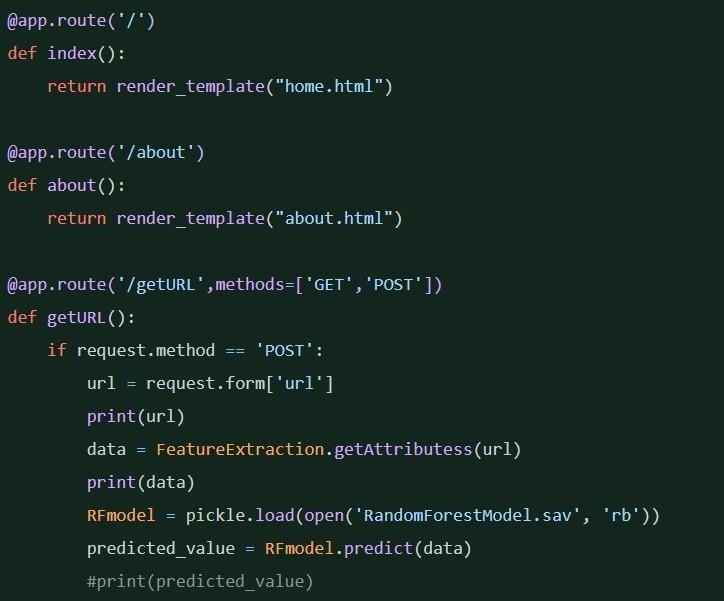
### APPENDIX-1(RESULTS)

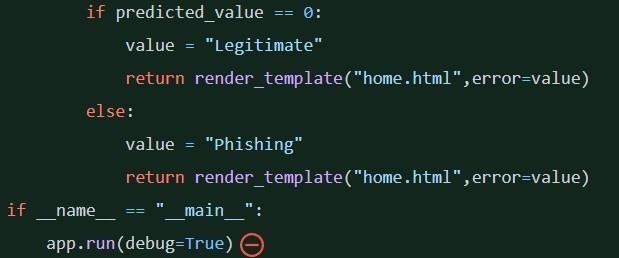




**APPENDIX-2(SAMPLE CODE)**

SOURCE CODE





### SAMPLE CODE OF MODELS:

{

"nbformat": 4,

"nbformat\_minor": 0, "metadata": {

"colab": {

"name": "Phishing Website Detection.ipynb", "provenance": [],

"collapsed\_sections": [], "toc\_visible": true

},

"kernelspec": { "name": "python3",

"display\_name": "Python 3"

}

},

"cells": [

{

"cell\_type": "markdown", "metadata": {

"id": "WTVY5lz4vJQM",

"colab\_type": "text"

},

"source": [

"# \*\*Phishing Website Detection by Machine Learning Techniques\*\*\n", "\n",

"\*Final project of AI & Cybersecurity Course\*"

]

},

{

"cell\_type": "markdown", "metadata": {

"id": "\_cJA-yD-vNcS", "colab\_type": "text"

},

"source": [

"## \*\*1. Objective:\*\*\n",

"A phishing website is a common social engineering method that mimics trustful uniform resource locators (URLs) and webpages. The objective of this project is to train machine learning models and deep neural nets on the dataset created to predict phishing websites. Both phishing and benign URLs of websites are gathered to form a dataset and from them required

URL and website content-based features are extracted. The performance level of each model is measures and compared.\n",

"\n",

"\*This project is worked on Google Collaboratory.\*<br>\n",

"\*The required packages for this notebook are imported when needed.\*"

]

},

{

"cell\_type": "markdown", "metadata": {

"id": "NOq598WCqIol", "colab\_type": "text"

},

"source": [

"## \*\*2. Loading Data:\*\*\n", "\n",

"The features are extracted and store in the csv file. The working of this can be seen in the 'Phishing Website Detection\_Feature Extraction.ipynb' file.\n",

"\n",

"The reulted csv file is uploaded to this notebook and stored in the dataframe."

]

},

{

"cell\_type": "code", "metadata": {

"id": "C297HhYulXcb", "colab\_type": "code",

"outputId": "d6e2a9df-586e-4192-b8ec-1e7b7025c0c3", "colab": {

"base\_uri": "https://localhost:8080/", "height": 70

}

},

"source": [

"#importing basic packages\n", "import pandas as pd\n", "import numpy as np\n", "import seaborn as sns\n", "import matplotlib.pyplot as plt"

],

"execution\_count": 0, "outputs": [

{

"output\_type": "stream", "text": [

"/usr/local/lib/python3.6/dist-packages/statsmodels/tools/\_testing.py:19:

FutureWarning: pandas.util.testing is deprecated. Use the functions in the public API at pandas.testing instead.\n",

" import pandas.util.testing as tm\n"

],

"name": "stderr"

}

]

},

{

"cell\_type": "code", "metadata": {

"id": "fVPglpaf4REa", "colab\_type": "code",

"outputId": "eef4a4ca-e12d-4cd3-e011-20376fc752a2", "colab": {

"base\_uri": "https://localhost:8080/", "height": 217

}

},

"source": [

"#Loading the data\n",

"data0 = pd.read\_csv('5.urldata.csv')\n", "data0.head()"

],

"execution\_count": 0, "outputs": [

{

"output\_type": "execute\_result", "data": {

"text/html": [ "<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n", " vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n", " vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n", " text-align: right;\n",

" }\n", "</style>\n",

"<table border=\"1\" class=\"dataframe\">\n", " <thead>\n",

" <tr style=\"text-align: right;\">\n", " <th></th>\n",

" <th>Domain</th>\n", " <th>Have\_IP</th>\n", " <th>Have\_At</th>\n",

" <th>URL\_Length</th>\n", " <th>URL\_Depth</th>\n", " <th>Redirection</th>\n",

" <th>https\_Domain</th>\n", " <th>TinyURL</th>\n",

" <th>Prefix/Suffix</th>\n", " <th>DNS\_Record</th>\n", " <th>Web\_Traffic</th>\n", " <th>Domain\_Age</th>\n", " <th>Domain\_End</th>\n", " <th>iFrame</th>\n",

" <th>Mouse\_Over</th>\n", " <th>Right\_Click</th>\n",

" <th>Web\_Forwards</th>\n", " <th>Label</th>\n",

" </tr>\n",

" </thead>\n", " <tbody>\n", " <tr>\n",

" <th>0</th>\n",

" <td>graphicriver.net</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>ecnavi.jp</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>hubpages.com</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>extratorrent.cc</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>3</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>icicibank.com</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>3</td>\n", " <td>0</td>\n", " <td>0</td>\n",

" <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " <td>1</td>\n", " <td>0</td>\n", " <td>0</td>\n", " </tr>\n",

" </tbody>\n", "</table>\n", "</div>"

],

"text/plain": [

" Domain Have\_IP Have\_At ... Right\_Click Web\_Forwards Label\n",

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| "0 | graphicriver.net | 0 | 0 ... | 1 | 0 0\n", |
| "1 | ecnavi.jp | 0 | 0 ... | 1 | 0 0\n", |
| "2 | hubpages.com | 0 | 0 ... | 1 | 0 0\n", |
| "3 | extratorrent.cc | 0 | 0 ... | 1 | 0 0\n", |
| "4 | icicibank.com | 0 | 0 ... | 1 | 0 0\n", |

"\n",

"[5 rows x 18 columns]"

]

},

"metadata": {

"tags": []

},

"execution\_count": 2

}

]

},

# REFERENCES

1. S.Sailesh & E. Varsha “Web Phishing Detection Using a Deep Learning Framework”. Hindawi Wireless Communications and Mobile Computing Volume 2018, Article ID 4678746, 9 pages.
2. D. Akila & K.Karthiga, Phishing Websites Detection Using Machine Learning. International Journal of Recent Technology and Engineering (IJRTE)

ISSN: 2277-3878, Volume-8, Issue-2S11, September 2019

1. J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran and B. S. Bindhumadhava, “Phishing Website Classification and Detection Using Machine Learning,” 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1–6, 10.1109/ICCCI48352.2020.9104161.
2. T. Peng, I. Harris, and Y. Sawa, “Detecting Phishing Attacks Using Natural Language Processing and Machine Learning,” Proc. - 12th IEEE Int. Conf. Semant. Comput. ICSC 2018, vol. 2018–Janua, pp. 300– 301, 2018.
3. Shad and S. Sharma, “A Novel Machine Learning Approach to Detect Phishing Websites Jaypee Institute of Information Technology,” pp. 425–430, 2018.
4. 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. 949– 952.
5. M. Aydin and N. Baykal, “Feature extraction and classification phishing websites based on URL,” 2015 IEEE Conf. Commun. NetworkSecurity, CNS 2015, pp. 769–770, 2015.
6. Shekokar, N. M., Shah, C., Mahajan, M., & Rachh, S. (2015). An ideal approach for detection and prevention of phishing attacks. Procedia Computer Science, 49, 82-91.
7. Lakshmi, V. S., & Vijaya, M. S. (2012). Efficient prediction of phishing websites using supervised learning algorithms. Procedia Engineering, 30, 798- 805.
8. P. Yi, T. Zhu, Q. Zhang, Y. Wu, and L. Pan, “Puppet attack: A denial of service attack in advanced metering infrastructure network,” Journal of Network and Computer Applications, vol. 59, no. 1, pp. 325–332, 2016.
9. P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, “A denial of service attack in advanced metering infrastructure network,” in Proceedings of the 2014 IEEE International Conference on Communications (IEEE ICC 2014), pp. 1029–1034, IEEE, Sydney, Australia, June 2014.
10. S. Xiao,W. Gong,D. Towsley,Q. Zhang, andT. Zhu, “Reliability analysis for cryptographic key management,” in Proceedings of the IEEE International Conference on Communications (IEEE ICC 2014), Sydney, Austrailia, June 2014.
11. D. Jiang, Z. Yuan, P. Zhang, L. Miao, and T. Zhu, “A traffic anomaly detection approach in communication networks for applications of multimedia medical devices,” Multimedia Tools and Applications, vol. 75, no. 22, pp. 14281– 14305, 2016
12. Aljofey A, Jiang Q, Qu Q, Huang M, Niyigena JP. An effective phishing detection model based on character level convolutional neural network from URL. Electronics. 2020 Sep;9(9):1514.