**“DETECTION OF PHISHING SITE”**

A

*Project report*

*Submitted for minor project work*

**Department of Computer science & Engineering**

**CET,BBSR.**

****

**(**Academic year 2021-22)

**Guided by Submitted By**

Ms.Arunima Hota Sridhar Pra. Dash

Ashish Kumar Sahu

Srijeet Panda

Debashish Behera

**CERTIFICATE**

This is to certify that the project entitled “Detection of Phishing site” has been submitted to Department of Computer Science and Engineering,College of engineering and technology for the fulfillment of the requirement for the minor project by following student of 3rd year B.Tech(CSE)

**Student Name(with regd no.)**

1801106536 Sridhar Prasad Dash

1801106122 Ashish Kumar Sahu

1801106537 Srijeet Panda

1921106026 Debashish Behera

**Ms. Arunima Hota**

(Project Guide)

**ACKNOWLEDGEMENT**

We like to share our sincere gratitude to all those help us in completion of this project.During the work we faced many challenges due to our lack of knowledge and experience but

These people help us to get over from all the difficulties and final completion of our project.

We would like to thank **Ms. Arunima hota** ma’am for her governance,because of which tour whole team was able to learn the minute aspect of the project.

All of our team is thankful to all the faculties and staff of the department for their help and support towards this project.

THANK YOU ALL.

1801106536 Sridhar Prasad Dash

1801106122 Ashish Kumar Sahu

1801106537 Srijeet Panda

1921106026 Debashish Behera

**DECLARATION**

We,hereby declare that the discussion entitled “Detection of phishing site” being submitted by us towards the fulfillment of minor project work is carried out by us under supervision of **Ms. Arunima Hota**, and have not submitted anywhere else.

We will be solely responsible if any kind of plagiarism is found.

**Date: 2nd June 2021**

1801106536 Sridhar Prasad Dash

1801106122 Ashish Kumar Sahu

1801106537 Srijeet Panda

1921106026 Debashish Behera

**ABSTRACT**

Tremendous resources are spent by organizations guarding against and recovering from *cyber security attacks* by online hackers who gain access to sensitive and valuable user data. Many *cyber infiltrations* are accomplished through *phishing attacks* where users are tricked into interacting with web pages that appear to be legitimate. In order to successfully fool a human user, these pages are designed to look like legitimate ones. Since humans are so susceptible to being tricked, automated methods of differentiating between phishing websites and their authentic counterparts are needed as an extra line of defense. The aim of this project is to develop these methods of defense utilizing various approaches to categorize websites. Specifically, we have developed a system that uses machine learning techniques to classify websites based on their URL. We used two classifiers: *logistic regression* and *Naive bayes*. The classifiers were tested with a data set containing real world URLs where each could be categorized as good or bad site. The results of the experiments show that the classifiers were successful in distinguishing real websites from fake ones over 98% of the time.

KEYWORD:Cyber security attack, Cyber infiltration, Phishing attack, Logistic regression, Naive bayes

**CONTENT**

**CERTIFICATION……………………………………………………..A**

**ACKNOWLEDGEMENT…………………………………………….B**

**DECLARATION………………………………………………………C**

**ABSTRACT……………………………………………………………D**

**CHAPTER-1……………………………………………………………1**

Introduction…………………………………………………………….1

1.1.What is phishing ?………………………………………………..1

1.2.How Phishing attack is done by attacker?……………………..2

1.3.Status of phishing in INDIA and other country……………….3

1.4.Prevention…………………………………………………………4

**CHAPTER-2……………………………………………………………6**

Phishing site detection using ML…………………………………….6

2.1.Data collection and structuring…………………………………6

2.2.Using Logistic regresssion………………………………………6

2.2.1.Maximun likelihood estimation…………………………..7

2.2.2.Advantage…………………………………………………..8

2.2.3.Disadvantage……………………………………………….8

2.3.Using Multinomial naive bayes………………………………..10

2.3.1.Advantage………………………………………………….11

2.3.2.Disadvantage………………………………………………11

2.4.Using Pipelined logistic regression……………………………12

**CHAPTER-3…………………………………………………………..14**

Deployment of the model using Fast API………………………….14

3.1.Dumping the model into .pkl file……………………………...14

3.2.Deployment using Fast API & uvicorn………………………..14

**CHAPTER-4……………………………………………………………15**

4.1.Step by step procedure to use the product…………………….15

4.2.Conclusion………………………………………………………...16

4.3.Reference…………………………………………………………..17

**LIST OF FIGURES**

Fig:1- Process of phishing…………………………………………………….2

Fig:2-Cofusion matrix for logistic regression…………………………9

Fig:3-Cofusion matrix for multinomial naivebayes………………11

Fig:4-Cofusion matrix for pipelined model……………………….…13

**CHAPTER:1**

**INTRODUCTION**

* 1. **What is Phishing?**[ref:10]

Phishing is a network type attack where the attacker creates the fake of an existing webpage to fool an online user into elicit personal Information. Phishing is the combination of social engineering and technical methods to convince the user to reveal their personal data. This report discusses about the Phishing and social engineering attack theoretically and they’re in the life of human beings. Phishing is typically carried out by Email spoofing or instant messaging. It targets the user who has no knowledge about social engineering attacks, and internet security, like persons who do not take care of privacy of their accounts details such as Facebook, Gmail, credit banks accounts and other financial accounts.

In another sense ,Phishing is an act of attempt to acquire information such as usernames,passwords, and credit card details,etc of a person or organization illegaly in an electronic communication. Phishing is committed so that the criminal may obtain sensitive and valuable information about a consumer, usually with the goal of fraud to obtain the customer bank and other financial information. Phishing are typically carried out by e-mail spoofing or instant messaging.

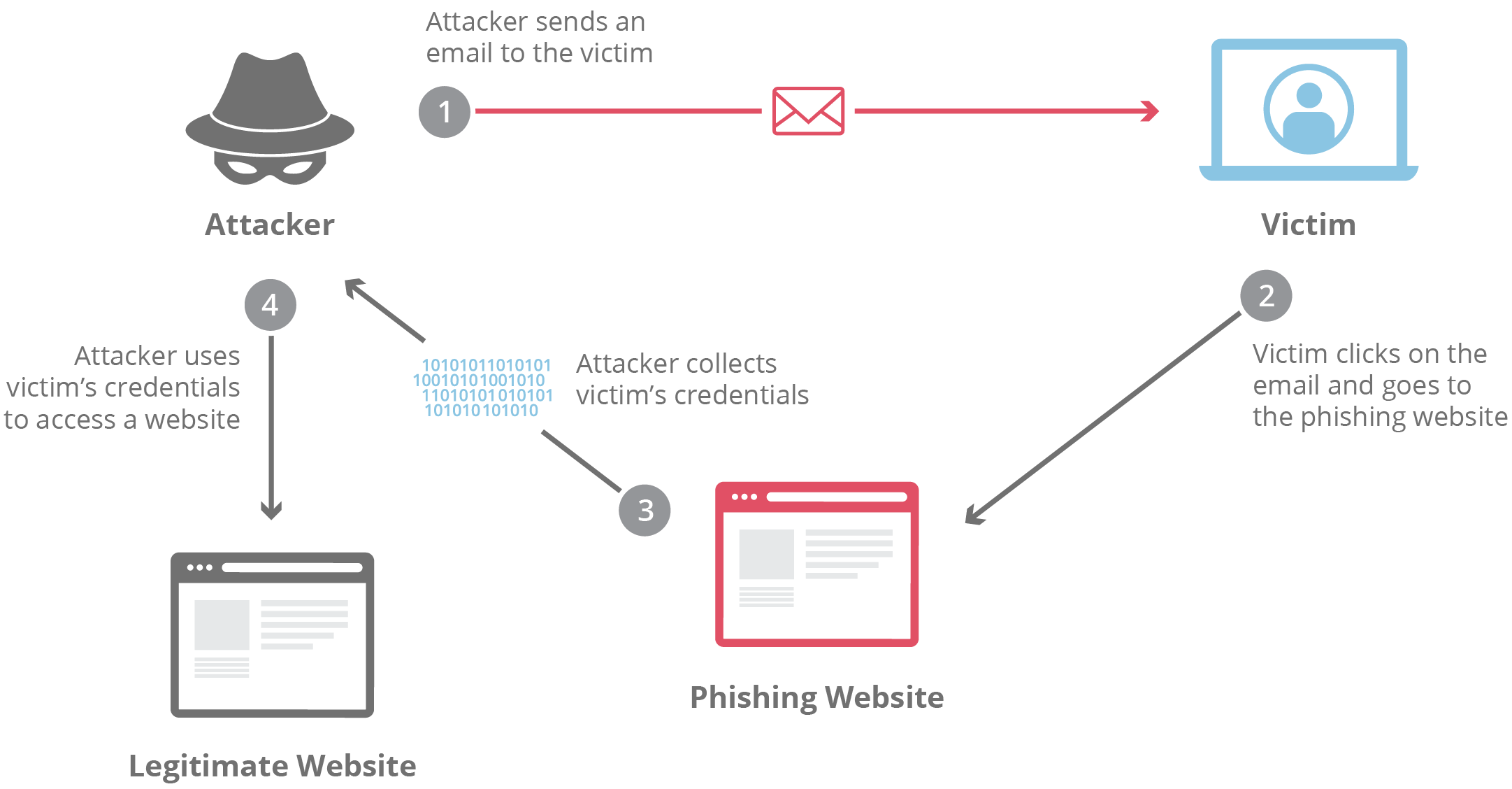
* 1. **How Phishing attack is done by attacker?**

Step1: Attacker sends an email to the victim.

Step2: Victim clicks on the email and goes to the phishing website.

Step3: Attacker collects the victim’s credentials through browser cache or by other medium.

Step4: Attacker uses victim’s credentials to access a website as if he is the real user.



(Fig:1)

* 1. **Status of phishing in INDIA and other country**
* As per report of The Economic Times, there is a increase of 37% of phishing attack in first quater of 2021 than last years last quater.
* A record of 3.5 lakh cases of phishing attack has been registered only in the month of April 2021.
* INDIA stood second to US to host phishing attacks.
* 29% of whole worlds phishing attack cases are from INDIA.
* In 2020 the finance loss due to phishing attack is about 10 billion USD.
* The financial sector was the most heavily targeted for phishing schemes, constituting 92.6 percent of all phishing attacks
* The growth of phishing attacks are exponential. And also, due to internet boom in 2015 in india, accelates this trend more rapidly.
  1. **Prevention**

1. ****Keep Your Firewall Turned On:****

 A firewall helps protect your computer from hackers who might try to gain access to crash it, delete information, or even steal passwords or other sensitive information. Software firewalls are widely recommended for single computers. The software is prepackaged on some operating systems or can be purchased for individual computers. For multiple networked computers, hardware routers typically provide firewall protection.

1. ****Install or Update Your Antivirus Software:****

 Antivirus software is designed to prevent malicious software programs from embedding on your computer. If it detects malicious code, like a virus or a worm, it works to disarm or remove it. Viruses can infect computers without users’ knowledge. Most types of antivirus software can be set up to update automatically.

1. ****Install or Update Your Antispyware Technology:****

Spyware is just what it sounds like—software that is surreptitiously installed on your computer to let others peer into your activities on the computer. Some spyware collects information about you without your consent or produces unwanted pop-up ads on your web browser. Some operating systems offer free spyware protection, and inexpensive software is readily available for download on the Internet or at your local computer store. Be wary of ads on the Internet offering downloadable antispyware—in some cases these products may be fake and may actually contain spyware or other malicious code. It’s like buying groceries—shop where you trust.

1. ****Keep Your Operating System Up to Date:****

Computer operating systems are periodically updated to stay in tune with technology requirements and to fix security holes. Be sure to install the updates to ensure your computer has the latest protection.

1. ****BeCareful What You Download:****

Carelessly downloading e-mail attachments can circumvent even the most vigilant anti-virus software. Never open an e-mail attachment from someone you don’t know, and be wary of forwarded attachments from people you do know. They may have unwittingly advanced malicious code.

CHAPTER:2

**PHISING SITE DETECTION USING MACHINE LEARNING**

**2.1. Data collection and data structuring**

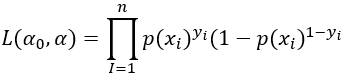
1. We collected our training data set from **Kaggle.com**.The dataset is in csv file format containg 2 columns and 549346 rows. The first column contains URLs and the second column indicates good or bad website.From plotting it was known that the dataset contains approximately 4lakhs of good URLs and 1.5lakhs of bad URLs.
2. Then **RegexpTokenizer** is applied in each URL to get the tokens.
3. Each token is then stemmed using stemmer to get all the related words in english language.
4. On each token **CountVectorizer** is used to transform a corpoa of text to a vector of token counts.
5. The data set is then sliced into training and test data set as the algorithms used in project are supervised.

**2.2. Using Logistic regression ML algo** [ref:1]

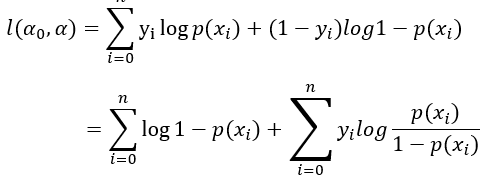
logistic regression IS a regression model. The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. Just like Linear regression assumes that the data follows a linear function, Logistic regression models the data using the sigmoid function.

lr

Since Logistic regression predicts probabilities, we can fit it using likelihood. Therefore, for each training data point x, the predicted class is y. Probability of y is either p if y=1 or 1-p if y=0. Now, the likelihood can be written as:

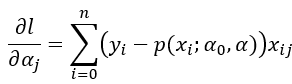


The multiplication can be transformed into a sum by taking the log:



## 2.2.1. Maximum Likelihood Estimation (MLE)

A method of estimating the parameters of probability distribution by maximizing a likelihood function, in order to increase the probability of occurring the observed data. We can find MLE by differentiating the above equation with respect to different parameters and setting it to be zero. For example, the derivative with respect to one of the component of parameter alpha i.e. a\_j is given by:



## **2.2.2. Advantages**[ref:5]

* Logistic Regression is ****one of the simplest machine learning algorithms**** and is easy to implement yet provides great training efficiency in some cases. Also due to these reasons, training a model with this algorithm doesn't require high computation power.
* The predicted parameters (trained weights) give ****inference about the importance of each feature****. The direction of association i.e. positive or negative is also given. So we can use logistic regression to find out the relationship between the features.
* This algorithm allows models to be ****updated easily to reflect new data****, unlike decision trees or support vector machines. The update can be done using stochastic gradient descent.
* Logistic Regression ****outputs well-calibrated probabilities**** along with classification results.
* In a ****low dimensional dataset**** having a sufficient number of training examples, logistic regression is ****less prone to over-fitting****.

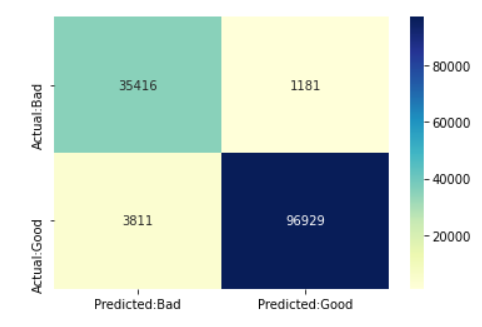
## **2.2.3. Disadvantages** [ref:5]

* Logistic Regression is a statistical analysis model that attempts to predict precise probabilistic outcomes based on independent features. On ****high dimensional datasets****, this may lead to the model being ****over-fit on the training set****, which means overstating the accuracy of predictions on the training set and thus the model ****may not be able to predict accurate results on the test set****.
* ****Non linear problems can't be solved**** with logistic regression ****since it has a linear decision surface****.

Here,We used logistic regression instead of linear regression because it is a classification problem(whether the URL is good or bad?).

A object of logistic regression is created and applied to the training data set to train the model and then test data set is applied and the accuracy and confusion matrix were drawn to analyse the model.Here is the result:

* ****Logistic Regression is giving 96% accuracy.****
* **Training Accuracy : 0.9782480479795345**
* **Testing Accuracy : 0.9636514559077306**
* **Confusion matrix:**



(Fig:2)

**2.3. Using Multinomial Naivebayes ML algo**

**Naive Bayes Algorithm** is a family of probabilistic algorithms based on applying Bayes’ theorem with the “naive” assumption of conditional independence between every pair of a feature.  
Bayes theorem calculates probability P(c|x) where c is the class of the possible outcomes and x is the given instance which has to be classified, representing some certain features.

Naive Bayes are mostly used in natural language processing (NLP) problems. Naive Bayes predict the tag of a text. They calculate the probability of each tag for a given text and then output the tag with the highest one.

### **Bayes’ Theorem for Naive Bayes Algorithm** [ref:2]

In a machine learning classification problem, there are multiple features and classes, say, C1,C2,…,Ck. The main aim in the Naive Bayes algorithm is to calculate the conditional probability of an object with a feature vector x1,x2,…,xn belongs to a particular class Ci,

P(Ci|x1,x2,…,xn)=P(x1,x2,…,xn|Ci).P(Ci)P(x1,x2,…,xn) for 1≤i≤k

Now, the numerator of the fraction on right-hand side of the equation above is

 P(x1,x2,…,xn|Ci).P(Ci)=P(x1,x2,…,xn,Ci)

From the calculation above and the independence assumption, the Bayes theorem boils down to the following easy expression:

P(Ci|x1,x2,…,xn)=(∏j=1j=nP(xj|Ci)).P(Ci)P(x1,x2,…,xn) for 1≤i≤k

The expression P(x1,x2,…,xn) is constant for all the classes, we can simply say that

P(Ci|x1,x2,…,xn)∝(∏j=1j=nP(xj|Ci)).P(Ci) for 1≤i≤k

### ****2.3.1. Advantages**[ref:6]**

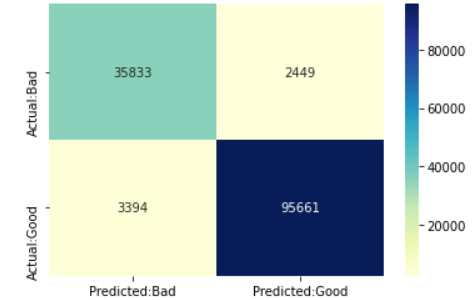
* This algorithm works quickly and can **save a lot of time**.
* Naive Bayes is suitable for solving **multi-class prediction** problems.
* If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.

### ****2.3.2.Disadvantages**[ref:6]**

* Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. **This limits the applicability of this algorithm in real-world use case**s.

A object of MultinomialNB is created and applied to the training data set to train the model and then test data set is applied and the accuracy and confusion matrix were drawn to analyse the model.Here is the result:

* ****MultinomialNB gives us 95% accuracy.****
* **Training Accuracy : 0.9741437687040817**
* **Testing Accuracy : 0.9574550194048217**
* **Confusion matrix:**

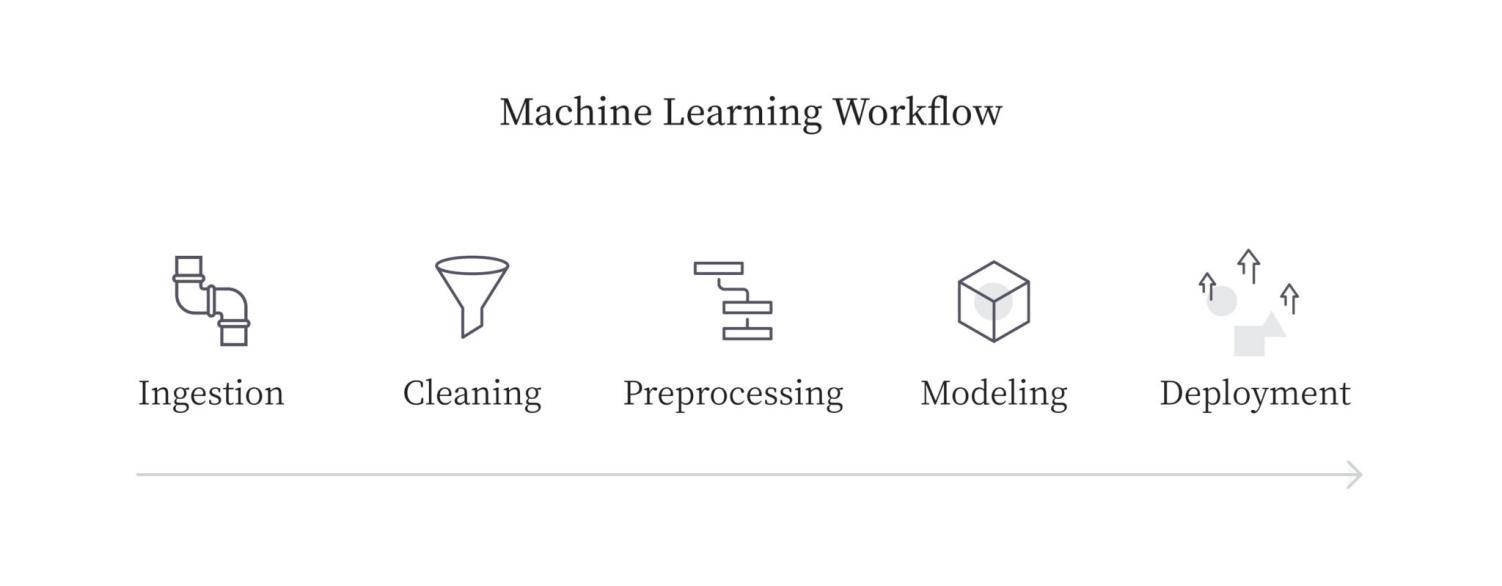
****

(Fig:3)

**2.4. Pipeline using Logistic regression**[ref:8]

One definition of an ML pipeline is a means of automating the machine learning workflow by enabling data to be transformed and correlated into a model that can then be analyzed to achieve outputs. This type of ML pipeline makes the process of inputting data into the ML model fully automated.

Machine learning pipelines consist of several steps to train a model. Machine learning pipelines are iterative as every step is repeated to continuously improve the accuracy of the model and achieve a successful algorithm. To build better machine learning models, and get the most value from them, accessible, scalable and durable storage solutions are imperative, paving the way for on-premises object storage.

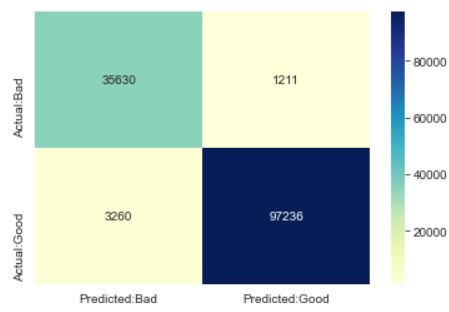
****

In a mainstream system design, all of these tasks would be run together in a monolith. This means the same script will extract the data, clean and prepare it, model it, and deploy it. Since machine learning models usually consist of far less code than other software applications, the approach to keep all of the assets in one place makes sense.

A object of logistic regression with pipeline is applied on training data set and the trained model is applied on the test data set and

the accuracy and confusion matrix were drawn to analyse the model.Here is the result:

* **Pipelined logistic regression gives us 98% accuracy.**
* **Training Accuracy : 0.9808911941244002**
* **Testing Accuracy : 0.9674450439430016**
* **Confusion matrix:**

****

(Fig:3)

CHAPTER-3

**DEPLOYMENT OF THE MODEL USING FAST API**

**3.1.Dumping the model into .pkl file**[ref:9]

In machine learning, while working with scikit library, we need to save the trained models in a file and restore them in order to reuse it.Also, we deal with different types and sizes of data. Some datasets are easily trained i.e- they take less time to train but the datasets whose size is large (more than 1GB) can take very large time to train on a local machine even with GPU. When we need the same trained data in some different project or later sometime, to avoid the wastage of the training time, store trained model so that it can be used anytime in the future.

So we dumped the pipelined logistic regression model into a file called “**phishing.pkl**”

**3.2.Creating a FastAPI object and running in localhost using uvicorn library**[ref:3,4]

FastAPI is a modern, fast (high-performance), web framework for building APIs with Python 3.6+ based on standard Python type hints. Uvicorn is a lightning-fast ASGI server implementation.

We craeted a python file “main.py” and created object of fastapi called ‘app’ init and passed the object through uvicorn to run in the local having URL: “**127.0.0.1:8000/docs**”

CHAPTER-4

**STEP BY STEP PROCEDURE TO USE THE PRODUCT**

1. Locate the **“main.py”** in your respective computer.
2. Install python and and set the path varible.(only for the first time)
3. Open the command prompt and these commands(only for the first time)

**>>pip install scikit-learn**

**>>pip install fastapi**

**>>pip install uvicorn**

**>>pip install joblib**

**>>pip install nest-asyncio**

1. Open command prompt and goto the main.py folder and run the command

**>>uvicorn main:app --reload**

1. Goto chrome browser and search for local host “**127.0.0.1:8000/docs**”.
2. Paste the url on the feature and click execute. It will show you website is good or bad.
3. To quit from program you simply goto the command prompt

And press **“ctrl+c**” .

**CONCLUSION**

1. This project is deployed using Fast API ans uvicorn unlike

Flask or other web technologies so it can be used offline. And it will far more faster than other web applications.

1. The model used to classify in this project is pipelined,so its accuracy is far more greater than the other unpipelined model.
2. This project can be deployed easily if your computer has python and chrome browser. It does not require any high end device or applications to be installed.

**REFERENCES**

1. **<https://medium.com/analytics-vidhya/the-math-behind-logistic-regression-c2f04ca27bca>**
2. **<https://www.hackerearth.com/blog/developers/introduction-naive-bayes-algorithm-codes-python-r/>**
3. **<https://fastapi.tiangolo.com/>**
4. **<https://www.uvicorn.org/>**
5. **<https://iq.opengenus.org/advantages-and-disadvantages-of-logistic-regression/>**
6. **<https://www.upgrad.com/blog/naive-bayes-explained/>**
7. **<https://www.researchgate.net/publication/328541785_Phishing_Website_Detection_using_Machine_Learning_Algorithms>**
8. **<https://algorithmia.com/blog/ml-pipeline>**
9. **<https://www.geeksforgeeks.org/saving-a-machine-learning-model/>**
10. **https://www.imperva.com/learn/application-security/phishing-attack-scam/**