**Chapter I: Introduction**

**1.1 Problem Statement:**

Phishing links often results to a deadly outcome in recent days and for the security researcher it has been a main area of concern. Because it is easy to create a fake website which looks so close to the legitimate one. Generally, the attacker sends a link to the victim pretending to be an authorized person. When the user clicks on to the link he got victimized. Sometimes the phisher push malware to victim’s computer and gain full control of it. The main goal of the phisher is to steal bank account credentials or personal documents.

In 2016 countries biggest cyberattack “The Bangladesh Bank Money Heist” stealing $1 billion US dollar from the reserve also started from a phishing email containing malware [1]. 2022 was a record year for phishing, with the Anti-Phishing Working Group (APWG) logging more than 4.7 million attacks. Since the beginning of 2019, the number of phishing attacks has grown by more than 150% per year. Attacks against the financial sector represented 27.7% of all phishing attacks. The average Business Email Compromise (BEC) attack attempted to steal $132,559 [2]. That shows the urgency of effective phishing detection techniques.

**1.2 Existing Approaches:**

Detecting and preventing phishing activities always been a challenge for researcher due to the way phishers carry out the attack to bypass the existing anti-phishing techniques. Mostly available Anti-phishing techniques are Black listing, Visual Similarity, heuristic etc.

Black listing method used widely for detecting phishing sites by adding black listed websites and IP addresses to the antivirus databases. Now the attackers became so clever to follow the techniques like “Obfuscation”, “fust-flux” or other techniques to automatically build proxies to host webpages, algorithmically produce new URLs and so on. It also cannot detect the 0-hour phishing attacks.

Visual similarity compares the new website with prestored websites signatures. The website signature includes screenshots, font styles, images, page layouts, logos, etc. Thus, these techniques cannot identify the fresh phishing websites and generate a high false-negative rate (phishing to benign).

Heuristic based detection which includes characteristics that are found to exist in phishing attacks in reality and can detect 0-hour phishing attack, but the characteristics are not guaranteed to always exist in such attacks and false positive rate in detection is very high [4].

**1.3 Objective of the study:**

* Finding out an easy solution for detecting Phishing URL
* Extracting the key feature of good and bad URL for analysis.
* To enhance the accuracy and precision of the existing methods
* Explore the best classification algorithm for building the model
* Develop and deploy phishing URL detection system for general users.
* Building an efficient model that will help user from cyber attacks or loss of personal data and properties.

**1.4 Resolving technology:**

Other Phishing detection approach are depending on different types of APIs or websites, which need to be update time to time. In this research NLP technology is used to detect phishing URL by analyzing the keywords of the URL. NLP (Natural Language Processing) based phishing link detection refers to using NLP techniques to analyze and classify URLs or text data with the goal of identifying phishing links. Phishing links are URLs that lead to deceptive or malicious websites designed to trick users into revealing sensitive information or performing harmful actions. In NLP-based phishing link detection, the focus is on understanding the content and language used in URLs to determine whether they are legitimate or phishing attempts. Data Urls are tokenized into words. Out of these words the root words were founded. Then a parse matrix created using count vectorizer that also takes the most frequently used words. Applying different machine learning algorithm to find out a best algorithm for detecting phishing site.

This study will help to carry out a simple but effective phishing detection technique using Machine Learning algorithms. At the end the model is deployed using FastApi. This will return a user interface where user can easily search with any URL to find out it is phishing or not.

**1.5 Research Question:**

1. Can NLP be used in finding Phishing site by analyzing the features of the URL in machine learning model?
2. Is it possible to build a simple but efficient phishing detection system by using this model?
3. Can this model help user to get rid of cyber attacks by giving better accuracy in phishing detection?

**1.6 Hypothesis:**

Machine learning can be use to predict a URL Phishing or not. The use of NLP makes the detection model independent from third party APIs and add more efficiency.

The keywords used in URLs can be used for detecting an URL. Classification algorithms like Logistic Regression gives better accuracy in dataset training and testing.

**Chapter II: Literature Review**

Some of the research papers that had been followed throughout the study are mentioned here. The papers deployed with same type of Machine learning algorithm are reviewed and their results are summarized.

1. This paper entitled with “Phishing Website Detection Using Machine Learning Algorithms” by Parvathy R. analyzed the lexical features and symbol used in the URL as features. They used classification algorithms KNN, Logistic Regression, Decision tree, SVM for comparing the performance and select the best algorithm for classifying phishing URL. The score obtained with Logistic Regression is 0.93 accuracy with 0.90 precision and 0.94 recall. They used different APIs and Indexes from internet which need to be updated over time. [8]
2. The paper titled with “NLP Based Phishing Attack Detection from URLs” authors have done a hybrid testing with NLP features and word vector. 7,357 URLs were used for the test, 10,572 features were extracted after the vectorization process. Later they dropped the number of features to 238 using feature selection algorithm. The achieved a test result of 97.2 for hybrid testing with Random Forest algorithm. A limitation of their study is Depending on few features and used a less amount of data. [10]
3. This study “Phishing Website Detection using Machine Learning Algorithms” deals with machine learning technology for detection of phishing URLs by extracting and analyzing various features of legitimate and phishing URLs. They extract features loke the IP address, number of dots, @ symbols etc. by using Python program. Later they make a comparison among the machine learning algorithms and found Random Forest with 97.4 accuracy with less False Postive and False Negative rate. They used a dataset of 36,711 URLs. Increasing the size of data set cause huge delay in deploying Random Forest algorithm. [10]
4. The paper Titled “Detecting phishing websites using machine learning technique” proposed a method LURL to compare with existing Detection methods. The Researcher evaluated the proposed method with 7900 malicious and 5800 legitimate sites. The LURL produce 94.3% accuracy. Here they build their own model with smaller size of data.[11]

**Chapter III: Research Methodology**

**3.1 Proposed Methodology:**

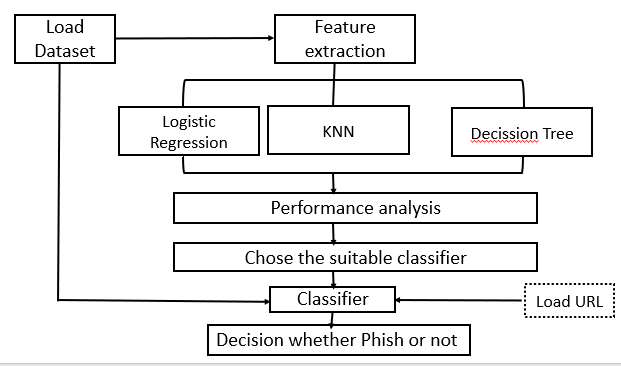
****

Fig 3.1.1: Proposed Methodology

**3.2 Data set Overview:**

To detect a phishing link we need two types of URL. These types are Good URL and Bad or malicious URL. The dataset I used throughout the research collected from internet. After removing all the duplicate URLs we have got 5,07,195 unique URL in two column with ‘good’ and ‘bad’ label.

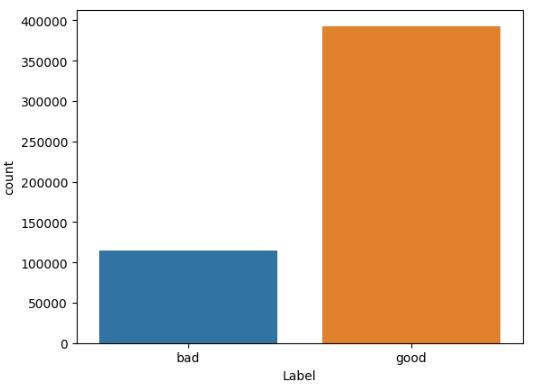


Fig 3.2.1: Dataset overview

**3.3 Tokenization:**

NLP technology deals with the natural languages. A phishing link is nothing but a combination of some words. So , we can acknowledge a phishing link as a sentence. For this purpose we need to gather all the proper words from the URLs. I use RegexpTokenizer() method from the nltk method. I pass "r'[A-Za-z]+'" to consider only alphabets for forming tokens.

**tokenizer = RegexpTokenizer(r'[A-Za-z]+')**

This will tokenize the url and provide the tokenized text. Here is an example-

|  |  |
| --- | --- |
| URL | Tokenized\_URL |
| mail.printakid.com/www.online.americanexpress | [mail, printakid, com, www, online, americanexpress] |

Fig 3.3.1 Tokenization

**3.4 Find root words:**

From the word token root words are extracted by using **SnowballStemmer.** Snowball is a small string processing language, gives root words. Snowball stemmer, also known as the Porter2 stemmer, is a widely used algorithm for stemming words in natural language processing (NLP). The main goal of stemming is to reduce words to their base or root form, so that different inflections or derivations of a word can be treated as the same word. This is particularly useful for tasks like text analysis, search engines, and information retrieval **[5].**

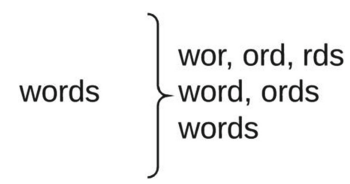


Fig 3.4.1: Root words finding

Difference Between Porter Stemmer and Snowball Stemmer:

* Snowball Stemmer is more aggressive than Porter Stemmer.
* Some issues in Porter Stemmer were fixed in Snowball Stemmer.
* There is only a little difference in the working of these two.

**3.5 Count Vectorizer:**

A Count Vectorizer is a text processing technique commonly used in natural language processing (NLP) and machine learning to convert a collection of text documents into a numerical format that can be used for various computational tasks. It is a type of feature extraction method that represents text data as a matrix of token counts. Initially tokenize the data and perform some basic preprocessing like converting the word into lowercase, remove punctuation etc. Next, it counts the frequency of each term (word) in the document. Each unique term in the document becomes a column in the resulting matrix, and the values in each column represent the count of that term in the respective document. The result is a matrix where each row represents a document, and each column represents a unique term in the entire collection of documents. The values in the matrix are the counts of the terms in each document. Since most documents contain only a subset of the entire vocabulary of terms, the resulting matrix is often sparse, meaning that most of its entries are zero. This sparse representation is memory-efficient. After vectorizing these data deployed to machine learning algorithm.

**3.6 Machine Learning algorithms:**

Four machine learning classification model algorithms are used for comparing the performance for detecting phishing URL.

**K Nearest Neighbors (KNN)**: is a straightforward algorithm, in machine learning for classification and regression tasks. It falls under the category of instance based or lazy learning algorithms, which means it doesn't create a model during training but memorizes the training data to make predictions on new unseen data points. During training KNN stores the feature vectors and their corresponding class labels. When classifying a data point KNN identifies the K neighbors in the training data based on a distance metric like Euclidean distance. The value of "K" is determined by the user. Represents the number of neighbors taken into consideration.

**Multinomial Naive Bayes:** MultinomialNB is an used text classification algorithm for scenarios where features represent word counts or frequencies. It builds upon the principle of Naive Bayes. Extends it from Gaussian Naive Bayes, which's simpler, in nature.

**Decision Tree:** Decision tree is a supervised learning algorithm. Used both for classification and regression task. As per the name it builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. It is constructed by recursively splitting the training data into subsets based on the values of the attributes until a stopping criterion is met, such as the maximum depth of the tree or the minimum number of samples required to split a node.[6]

**Logistic Regression:** It is a supervised algorithm that mainly used for classification task. It's a technique for predicting a binary outcome from a series of independent variables. A binary outcome is one in which there are only two options: the occurrence occurs (1) or it does not occur (0). Independent variables are variables or factors that have the ability to affect the result (or dependent variable). When dealing with binary data, the best method of analysis to use is logistic regression. When the performance or dependent variable is dichotomous or categorical in nature (e.g., "yes" or "no," "pass" or "fail," and so on. The goal is to predict the probability that an instance of belonging to a given class. This algorithm referred to as regression because it takes the output of the linear function as input and uses a sigmoid function to estimate the probability for the given class. It predicts the output of categorical independent variable. The output can be yes or no, true or false, 0 or 1.[7]

**3.7 Work Flow:**

Dataset

Remove duplicate urls

Tokenization

URLs

regular expression

Make sentence

Finding root words

Snowball stemmer

Count vectorizer

Data labeling

Split Data

Good

Bad

Apply ML Algo.

Fig 3.7.1: Work Flow diagram

**Chapter IV: Result Discussion**

Four classification algorithms – KNN, MultinomialNB, Decision tree and Logistic Regression are used for training the model. 1,21,121 benign or good URL and 31,038 malicious or bad URL are taken for testing the model. The testing score for each of the algorithm analyzed with confusion matrix.

TP= Predicts good URL as good

FN= Predicts bad URL as bad

FP= Predicts bad URL as bad

FN= Predicts bad URL as good

**4.1 Confusion Matrix:**

KNN: The KNN testing score was 0.90 by predicting 1,07,866 good url as good and 30,523 bad URL as bad. But FP and FN is a big concern for this algorithm.

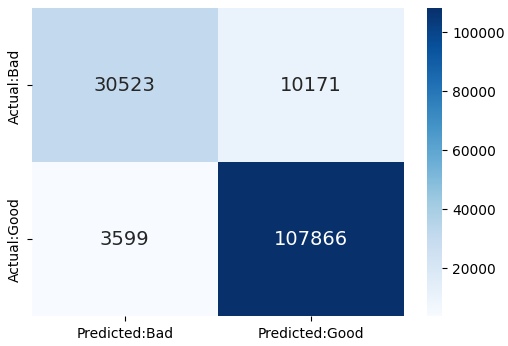


Fig 4.1.1: confusion matrix for KNN

MultinomialNB:

Testing Score for MultinomialNB was 0.95. Which is considerably good score. While successfully predicted 1,11870 URL good URL as good and 29,871 bad URL as bad.

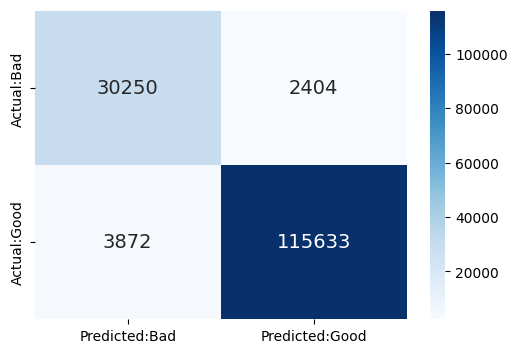


Fig 4.1.2: confusion matrix for MultinomialNB

Decision tree:

Testing score found for Decision tree was 0.90. where 1,15,469 good URL predicted as good and 22,983 bad URL predicted as bad. But here the FP and FN rate is high.

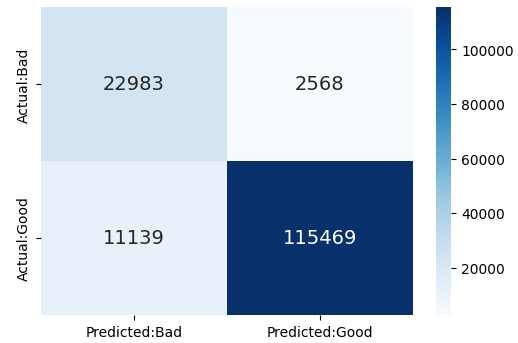


Fig 4.1.3: confusion matrix for Decision tree

Logistic Regression:

The testing score got in Logistic Regression is 0.96 which is highest compared to other algorithms. The Logistic Regression predicts 1,16,870 good URL as good and 29,871 URL as bad.

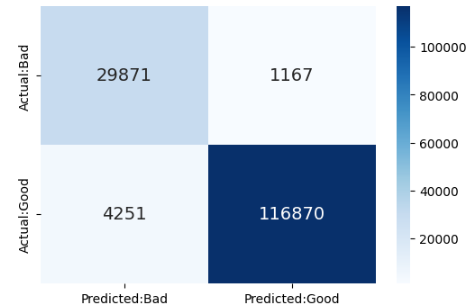


Fig 4.1.4: Confusion Matrix for LR

After comparing the confusion matrix of four algorithms its clear that the Logistic Regression gives Lower rate of FP and FN along with higher rate of TP and TN.

**4.2 Performance Comparison:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Classifier | Precision | Recall | f-1-score | Accuracy |
| KNN | 0.91 | 0.91 | 0.91 | 0.91 |
| MultinomialNB | 0.96 | 0.96 | 0.96 | 0.96 |
| Decision Tree | 0.83 | 0.91 | 0.91 | 0.91 |
| Logistic Regression | 0.97 | 0.96 | 0.97 | 0.97 |

Table 4.2.1: Comparison Table

The comparison table shows Logistic Regression algorithm gives better accuracy rate up to 97% with good precision and recall.

**Chapter V: Conclusion**

In this study a system has been developed for detecting a phishing site. The process followed here is simple but efficient. In feature Extraction, simply store and labeling the most used words in the phishing and benign URL. The Logistic regression provides the best accuracy and selected for building the model. Finally the Machine Learning Model was deployed to Fast API using uvicorn server. Where user can directly interact with the model and by inputting an URL can check whether it is a phishing site or not.

We also tried to use random forest algorithm for testing. But it took longer time to response, as per the big dataset are used for training. In future it is recommended to reduce no. of features in dataset and try other algorithms for better performance.

Reference:

1. Bangladesh Bank money heist: <https://www.thedailystar.net/backpage/news/bb-cyber-heist-malware-sent-bb-network-thru-emails-1694299>
2. Anti-Phishing Working Group (APWG): Phishing activity trends report—last quarter- <https://apwg.org/trendsreports/>
3. An effective detection approach for phishing websites using URL and HTML features- <https://www.nature.com/articles/s41598-022-10841-5>
4. Mahmoud Khonji, Youssef Iraqi, "Phishing Detection: A Literature Survey IEEE, and Andrew Jones, 2013
5. About Snowball stemmer: A string processing language- <https://www.geeksforgeeks.org/snowball-stemmer-nlp/>
6. About decision tree: A classification algorithm- <https://www.geeksforgeeks.org/decision-tree/>
7. About Logistic regression: <https://www.ibm.com/topics/logistic-regression>
8. Parvathy R. “Phishing Website Detection Using Machine Learning Algorithms” Proceedings of the National Conference on Emerging Computer Applications (NCECA)-2022
9. Ebubekir Buber, Banu Diri, Ozgur Koray Sahingoz et Al. “LP Based Phishing Attack Detection from URLs” Springer International Publishing AG, part of Springer Nature 2018 A. Abraham et al. (Eds.): ISDA 2017, AISC 736, pp. 608–618, 2018
10. Rishikesh Mahajan, Irfan Siddavatam. “Phishing Website Detection using Machine Learning Algorithms” International Journal of Computer Applications (0975 – 8887)
11. Ashit Kumar Dutta. “Detecting phishing websites using machine learning techniqu” Department of Computer Science and Information System, College of Applied Sciences, Almaarefa University, Riyadh, Saudi Arabia- October 11, 2021