Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

DETECTING PHISHING WEBSITES BY IMPLEMENTING MACHINE LEARNING ALGORITHMS

Vishal Jha[1], Vaibhav Dixit[2], Viji D[3]
[1,2,3]Department of Computer Science and Engineering,
[1,2,3]SRM Institute of Science and Technology Chennai, India
vu6770@srmist.edu.in, vr1184@srmist.edu.in. vijid@srmist.edu.in

Abstract —Now-a-days we are completely or say almost completely dependent on various websites for our day-to-day activities like e-banking, shopping, e-services etc. Most of the sites involve making online transactions which require our confidential information like e-PINs, username, passwords etc. A lot of fraudulent websites have come up now-a-days which are similar in appearance to the original websites but contain malicious links or ask for confidential information which are sensitive in nature and can cause financial losses or data theft to the user. These websites which contain such malicious links or ask for sensitive information are called Phishing Websites. With extremely large number of such websites, it becomes a very difficult and tedious job to detect the fraudulent ones. For the prediction and detection of these phishing websites, we propose a system that works on classification techniques and algorithms and classifies the data sets as phishing/legitimate. It is detected on various characteristics like URL(Uniform Resource Locator), Domain Name, Domain Entity, DNS etc. The algorithms used in this model are decision tree and logistic regression which provided the maximum accuracy among all the existing models We have used exhaustive datasets to train the model so that maximum accuracy can be achieved.

Keywords — Phishing, URL, DNS, Decision Tree MANET, Multihop.

I. INTRODUCTION

Phishing stands for an illegitimate/fraudulent attempt to gain access to are make misuse of sensitive and confidential information like password, userids, banking details etc. The most common practice used by phishers is developing a website which is almost similar to the original one and trap the user to disclose their confidential information on that website. Phishing attacks are also carried out by E-mails and by sending links to the victims hidden in the message content of an e-mail or SMS. These messages contain attractive discount offers or any other schemes to lure the user into visiting that link and disclose their details. The attackers also pretend to be IT administrators of various banking organizations and ask for E-banking details of customers via E-mails, Links sent through SMS. Prevention of such phishing attacks have become a very crucial task today and one way to do it is to make the users capable enough to differentiate between original/legitimate websites and fake ones. One more challenge that is faced in the detection of phishing websites is the use of images instead of texts by the phishers since detection of images is rather difficult than detection of texts. The easiest way to prevent phishing is blacklisting, but this method is not very practical and helpful as it can only detect and prevent the old and existing websites, for detection of old as well as new ones, Machine learning classification and data mining algorithms are very useful and effective. In this paper we have used improved Decision Tree and Logistic Regression model with accuracy level of 97.529%.

II. LITERATURE SURVEY

In this section we will discuss about the existing work done related to this model and will have a brief look at their advantages and drawbacks, functioning, proposals etc.

In the paper “Data Mining based on Phishing detection and classification” [1], aggression analysis using KNN algorithm is discussed. The dataset is obtained from Phishtank. The accuracy obtained using

KNN is 95.35% which is quite effective however there are some drawbacks in this model, the most evident one is that the model is sensitive to the scale for the data and irrelevant features which makes it prone to overfitting.

In "Empirical Analysis of blacklisting and whitelisting methodology to detect phishing websites" [2] a new are comparative fast boosting algorithm called Adaboost whose major advantage was its flexibility to combine with any algorithm for different kinds of analysis on varied data sets but the major drawback was its stand-alone efficiency of 91.00 % because of the weak classifiers.

In the paper titled "Image consumption for detection of content based phishing techniques" [3], a method which is termed as Feature selection (Wrappers) is used for the prediction of salient features of the websites. Since it is not possible and also very impractical to include all the features while training the model/classifier in predicting the websites. In this model, a classifier of inductive type is used. The main motive behind using this model was to eliminate redundant features and assign a score to each subset which is based on the performance and the error rate of the classifier. By implanting this methodology, a more filtered feature set is obtained and a significant improvement is achieved in the performance of the classifier. The final accuracy obtained after training the classifier is 97.20% and the major advantage of using this method is that it provides the important and essential features required for the classification and thus improves the accuracy. However, there is a major drawback in this method since it is a complex task as well as time consuming and also involves extra computational overhead.

"Implementation of NLP and ML in the prevention of phishing attacks" [4] was a survey published in IEEE in the year 2019 throws light on a very important yet less discussed approach towards detection of phishing websites. Majority of the existing anti-phishing models are not significantly effective against DNS based poisoning method of conducting phishing attacks. This method primarily focuses on this method of phishing. Datasets for training the model is taken from performance characteristics of various websites. To demonstrate the efficiency of this model, performance of four classification algorithms are measured namely Naïve Bayes, K-Nearest Neighbor, Linear Discriminant Analysis and Support Vector Machine. The data sets comprised of over 10,000 real world logs of routing information of various websites which were observed over a week. Although this model's effectiveness was only limited to DNS Poisoning based approach, it achieved very high accuracy of 98.70%.

In the paper titled "A feature-based framework based on Machine learning and data mining techniques" [5], web mining method is used to extract the features and train the classifier. Bag of words (BOG) technique is used to extract data from web content. Web mining is a term that is used to refer to the data mining techniques performed primarily on web content. The data is extracted and put into the hierarchy of similar data. Therefore, the data set preparation in this method is done using web mining. Various machine learning algorithms like Random Forest, Naïve bayes, KNN, Support vector machine are taken and the data sets is trained on them and the accuracy of the output is observed. Using Naïve Bayes classifier around 92.25% of phishing instances were predicted accurately, random forest method showed 97.592% and SVM gave 93.88% of correct classification. Hence this paper gave a conclusion that random forest achieves greater accuracy when used on a pre-processed data set.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

Classification Algorithm	Maximum accuracy achieved
Random Forest	97.592%
Naïve Bayes Classifier	92.250%
Support Vector Machine	93.880%
Decision Tree	96.750%
KNN	95.368%

Table 1 Accuracy level of Different Algorithms

Title	Method Used	Advantage	Limitation	Accuracy
Detecting Phishing websites by Aggression analysis of page layouts (2019) [6]	KNN	Robust in search space, cost of updating the model is very less	Sensitive to scale for the data and irrelevant features	95.352%
Comparison of Adaboost with Multiboosting for phishing techniques (2020) [7]	Adaboost	Flexible to combine with any algorithms for diverse datasets	Increased risk of overfitting because of weak classifiers.	91.00%
Detection and Prevention of Phishing website using Machine Learning Approach (2020) [8]	Decision Tree with data sets stored in stacks	Requires less effort in data preparation and pre processing	Small change in data leads to large change in tree structure	90.05%
A methodological overview on Phishing detection using framework (2020) [9]	SVM and KNN	Works well with even and semi structured data like texts, images.	Extensive training required, difficult to perform small calibrations	93.552%
Web crawling based phishing attack detection (2020) [10]	Random forest with back propagation algorithm	Extremely versatile model, produces high accuracy and counters overfitting	Large number of trees can make the algorithm ineffective	97.525%
Detecting phishing websites using machine learning (2020) [11]	Linear Regression and SVM	Simple to implement and interpret the output coefficient	Oversimplification of dataset by assuming a linear relationship	92.34%

Malicious website detection using SVM and backpropagation (2020) [12]	Adaboost SVM	Versatile model, can be used with text or numeric data	Vulnerable to uniform noise	88.20%
Phishing website detection – A survey (2019) [13]	Decision tree, SVM along with backpropagation algorithm	Fast and easy calibration, minimal parameter required	Extremely sensitive to noise	90.25%
Phishing detection: Analysis of visual based approaches (2020) [14]	Naive bayes classifier	Suitable for classifying multiclass prediction problems	Faces the 0-frequency problem, estimation can be wrong	84.95%
Detection of phishing websites using data mining techniques (2019) [15]	Link guard algorithm	End host-based algorithm which is efficient in known and obscure datasets	Requires excessive training and increased risk of overfitting	91.25%

Table 2 Study of Different methodologies

III. METHODOLOGY & APPROACH

In this section, we will discuss about the proposed approach to implement the model for phishing detection. Various features can be used to differentiate between original and fake web pages such as domain entity, URLs, source code, encryption, web content, graphics and multimedia content etc. Our study will have a prime focus on domain name characteristics and URL based features. These features can be checked using several parameters such as IP address length(whether it is too long or shorter than the usual), whether // symbol is being used to redirect the user to a different location instead of the web page, it will also check for any prefix or suffix added to the URL.

Dataset Collection

The dataset to be used for training the model is taken from publicly available UCI Machine Learning Repository [16] and Phishtank [17]. It consists of 420,000 URLs with around 50000 phishing instances and 350000 safe/legitimate instances. The URLs with phishing instances are labelled as “bad” and those which are legitimate are labelled as “good”. A particular instance holds nearly 35 features and the values displayed as result are “good” for legitimate ones and “bad” for phishing instances.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

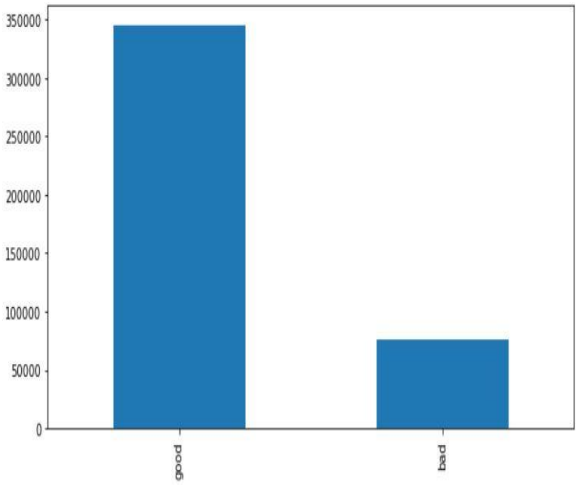

Figure 1 Dataset Description

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

Figure 2 Data labelling

Feature Extraction

After collection of the dataset the next step involved in our model is the extraction of features of the dataset. There is a total of 35 features that are taken into consideration for performing the classification on the dataset. These 35 features are then broadly classified under three major categories namely:

- Extraction from URL
- Extraction from content
- Extraction from rank

For feature extraction python library sklearn will be used and the commands for feature extraction are as follows:

- From sklearn. feature_extraction.text import CountVectorizer.
- from sklearn.feature_extraction.text import TfidfVectorizer

TF-IDF stands for term frequency inverse document frequency [18], it is a widely used algorithm whose main job is to transform the text data into a meaningful cluster of numbers and this cluster is then used to fit the machine learning algorithm for further prediction.

CountVectorSizer is a feature extraction tool provided in the SciKit python library which counts the frequency of a text appearing in the dataset and transforms the result as a vector quantity which can then be easily used in various models as vector type has more versatility and flexibility.[19]

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

List of Features		
URL based	URL length	Length of URL hostname
	URL Path length	Frequency of (.)
	(.) frequency in Hostname	Frequency of (/) in hostname
	Frequency of (-) in hostname	Presence of special characters (: ; % & +)
	Frequency of (@) in URL	Digit count in hostname
	(_) symbol in hostname	(_) frequency in pathname
	Frequency of certain keyword	Hexadecimal with % symbol
	Transport layer encryption	IP address
	www	Redirect to other page
	Unicode	Hexadecimal symbols

Table 3 Features extracted from dataset.

Methodology

In this section we will discuss about the methodology applied on the dataset after the feature extraction is over and the data is in pre-processed stage. Two machine learning algorithms are used to implement the classification Decision tree classifier and logistic regression.

Initial accuracy of the classifier model with the pre-processed data was found to be 85.88%. Python libraries such as pandas and NumPy were imported for the pre-processing task and feature extraction was done using Sklearn. Once the relevant features were picked out then the data set was implemented into the algorithms and the algorithms were implemented using sklearn library.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

Following commands were used to import the algorithms:

- From sklearn.linear_model import LogisticRegression
- From sklearn.tree import DecisionTreeClassifier

Various python libraries are used in the development of the model. They are Pandas [20], NumPy, Scikit and other tools such as Tkinter for developing and designing the output window that is the user interface of the classifier.

Pandas is used since it is a library which is specifically developed for data processing and comes with many in-built functions and tools for grouping, combining and extracting the data.

After the data collection process is over, pre-processing task is executed using tools and methodologies provided by python libraries such as pandas and sklearn.[20].Then after the pre-processing and feature extraction a proper dataset with relevant features is obtained which comprises of 80% of training data and 20% test data which is a exhaustive dataset which makes our model a self-learning model.

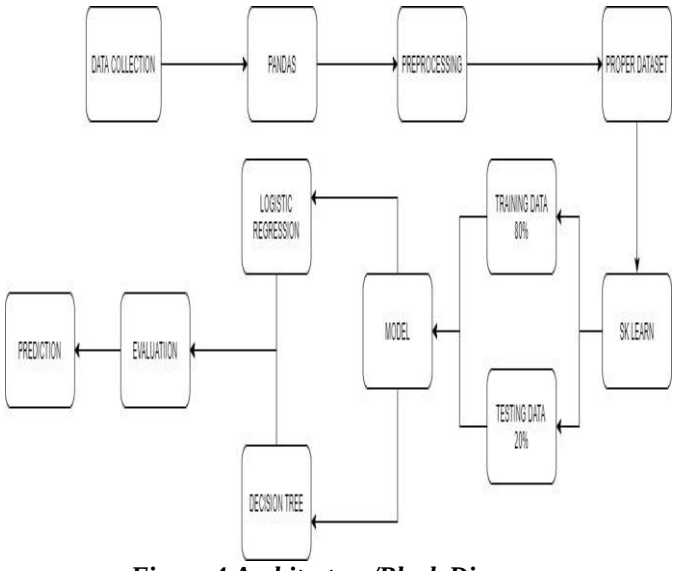

Figure 4 Architecture/Block Diagram

The dataset is classified using Decision tree and Logistic regression algorithm. Logistic regression is a very helpful in finding out the odds if more than one exemplary variable is present in the dataset. This process is quite similar to multiple linear regression only exception being the response variable binomial. Finally, the result obtained by logistic regression is the impact of the instances of variable on the odds ratio of the dataset. Logistic Regression produces a result which is unbiased, has lower false positives and variance. Thus, giving a much efficient output.

Since we were considering many features in the dataset i.e., 35, Decision tree was found to be the most suitable algorithm here. One of the major advantages of using decision tree is that it takes into consideration all the features and traces each instance to a conclusion. So here decision tree algorithm acted as a decisive factor for the model to decide upon which feature should be included in the classifier out of the 35 features. In decision tree each node or branch is labelled with the feature of the input and

each leaf is labelled with the class and then the probability distribution function of that class/classes is calculated and labelled to the tree which creates furthermore branches in the tree structure. In this method, the training data set is broken into simple and less complex subsets and the tree is incremented further till the entire training set is returned.

The primary idea to implement a decision tree is to find out the dense clusters and sparse regions in the dataset.

IV. OUTCOMES OBTAINED

The accuracy achieved using the logistic regression and decision tree classification algorithm is 97.47% against the initial accuracy of 85.82% of the pre-processed data before the feature extraction stage. The accuracy of the algorithm was improved by selecting only relevant features for the classifier instead of all the classifiers which lead to a faster and efficient output delivery and reduced calculation overhead. This also made possible the use of this model with minimal or extremely low system requirements and produce extremely effective results. An extremely large dataset comprising of 420646 URLs was used to train the model out of which around 55000 were phishing instances thus making the split up of training and testing data as 80% and 20% respectively.

Machine Learning Algorithm	**Accuracy of preprocessed dataset**	**Accuracy after feature extraction and classification**
Logistic Regression	85.882%	97.478%
Decision Tree	82.965%	97.526%

Table 4 Comparison of accuracies of algorithms

V. CONCLUSION

Detection and prediction of phishing websites is a very crucial step and the model used for detection needs to be efficient and reliable. Details and drawbacks of various other models that are currently in practice have been discussed in this paper. After conducting an in-depth survey and analysis of those existing models, architecture and functioning of this model was finalized keeping in mind the limitations and challenges faced by existing phishing classifiers. Out of all the algorithms and methods Logistic Regression and Decision tree were chosen because the improved accuracy of the dataset upon implementing this classification algorithm is 97.478%.

Hence, Decision tree and Logistic Regression were chosen as the final algorithm for detecting and classifying websites as phishing or legitimate.

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 4- Issue 2,Paper 2, July 2021

REFERENCES

[1] A Ahmad, Ayesha M, Associative "Data Mining based on Phishing detection and classification" 2019.

[2] B Nair, Wang et. Al "Detection of phishing websites using visual similarity approach "IIMAC 2018

[3] Chen A, Miller T, June 2020 "Detecting phishing websites using aggression techniques and cart IEEE energy conference "2020.

[4] Tomsaic S, Chandrakal V," Image consumption for detection of content based phishing techniques ACWG" 2019
[5] Sathya S, Videsha KL, "Detection of phishing website using based on similarity index of spatial layouts", APMMC, 2020.

[6] Haris Y, T kumaran, "Implementation of NLP and ML in the prevention of phishing attacks", IEEE 14th conference on ICSC 2020.

[7] Karthick A, Summina T, "Content approach to detect phishing websites", IEEE 16th SEEC conference.

[8] Anuthya TS, Salim, "A feature-based framework based on Machine learning and data mining techniques", SAARIC 2020

[9] Smadi S, Alsam Md, "Online e-mail spamming prevention framework based on re-inforcement learning", ICCAS 2020.

[10] Stallings R, Waryman B, "Content based high performance framework on detection of phishing attacks," 2019 IEEE conference, San Diego.

[11] Nalin R, Dr Thama, "Data Analysis of URL in detection of phishing websites using NLP and Machine Learning". IJAST 2019.

[12] T Vinnarasi, M Shankra, "Applying Heuristic method to detect web phishing using machine learning approaches", ICCT 2019.

[13] Singh Priya, Suresh YP, "Supervised learning networks to detect phishing websites", international computing conference 2018.
[14] Ravi kr G, R Nivetha, "Phishing data analysis usimg URL and static feature analysis", 4th International conference on Compting Application, 2020.

[15] Natin Jhad, Samant Sharma, Suri, "A novel data mining framework to detect phishing websites," IJPAM 2019.

[16] Ghosh Mona, A Subashis, Abdul Bari, Personalized "Whitelist Approach to detect and implement anti-phishing framework", IEEE 2020.

[17] https://En.Wikipedia.Org "Implementation of CART", Eng US 2018.

[18] Prakash P, Gupta SH, "Phishnet : blacklisting preditive framework to prevent web phishing and email spamming", 29th Confernce on SPJC , USA 2020

[19] Sheng S, Xhang hon C, "Empirical Analysis of blacklisting and whitelisting methodology to detect phishing websites.", 5th IJSARC 2019.

[20] Varshani Ram, T N Thakurta, " A Novel framework approach to detect phishing attacks", ICJCI SRMIIST 2019.