Phishing website detection using machine learning and deep learning techniques

J. Phys.: Conf. Ser. **1916** (2021) 012407 https://doi.org/10.1088/1742-6596/1916/1/012407

Retraction

Retraction: Phishing website detection using machine learning and deep learning techniques (*J. Phys.: Conf. Ser.* **1916** 012169)

Published 23 February 2022

This article (and all articles in the proceedings volume relating to the same conference) has been retracted by IOP Publishing following an extensive investigation in line with the COPE guidelines. This investigation has uncovered evidence of systematic manipulation of the publication process and considerable citation manipulation.

IOP Publishing respectfully requests that readers consider all work within this volume potentially unreliable, as the volume has not been through a credible peer review process.

IOP Publishing regrets that our usual quality checks did not identify these issues before publication, and have since put additional measures in place to try to prevent these issues from reoccurring. IOP Publishing wishes to credit anonymous whistleblowers and the Problematic Paper Screener [1] for bringing some of the above issues to our attention, prompting us to investigate further.

[1] Cabanac G, Labbé C and Magazinov A 2021 arXiv:2107.06751v1

Retraction published: 23 February 2022

Phishing website detection using machine learning and deep learning techniques

Selvakumari M[1], Sowjanya M[1], Sneha Das[1] and Padmavathi S[1]
[1]Department of Computer Science & Engineering, Sri Krishna College of Technology, Kovaipudur, Coimbatore, Tamil Nadu, India
17tucs211@skct.edu.in, 17tucs228@skct.edu.in, 17tucs225@skct.edu.in, padmavathi.s@skct.edu.in

Abstract. Phishing has become more damaging nowadays because of the rapid growth of internet users. The phishing attack is now a big threat to people's daily life and to the internet environment. In these attacks, the attacker impersonates a trusted entity intending to steal sensitive information or the digital identity of the user, e.g., account credentials, credit card numbers and other user details. A phishing website is a website which is similar in name and appearance to an official website otherwise known as a spoofed website which is created to fool an individual and steal their personal credentials. So, to identify the websites which are fraud, this paper will discuss the machine learning and deep learning algorithms and apply all these algorithms on our dataset and the best algorithm having the best precision and accuracy is selected for the phishing website detection. This work can provide more effective defenses for phishing attacks of the future.

1. Introduction

Phishing is one of the most damaging and dangerous criminal exercises growing in cyberspace [1]. It is deliberate that the users who go online to access the facilities delivered by the internet have been rapidly caught in phishing attacks for the past few years [2]. To obtain the user's credentials such as the username, password, card number and other sensitive and private credentials of the user, the attackers tend to attract these victims to click on the spoofed websites they have made [3]. These attackers in turn are profitable by these phishing attacks. Their usual attacks encounter on platforms like e-banking, e-payment platforms and e-commerce applications [4]. From the survey we have taken, it is known that, in the beginning, the blacklist technology was developed to prevent these phishing attacks [5]. Although that technique has been effective, the attacker can still be able to rat run the blacklist system by modifying the required characters in the URL in order to make the system believe that it is legitimate, and it makes the system not to identify it as a phishing website. Moreover, attackers have found a technique so as to pull the users to their side by asking security questions and pretend that it is a high-level secured website [6]. When the users reply to those questions, they easily get caught into that phishing website attack. Therefore, these attacks can be kept from happening by detecting those fake or spoofed websites and by creating alertness for users. Machine learning algorithms is one of the robust techniques in identifying the phishing websites [7]. In this work, several different models are compared including a deep learning model and the model having the best accuracy is selected for the phishing websites detection.

2. Related Work

There are several related works that apply machine learning techniques in identifying phishing websites. [8] They have described a focused literature study from a security perspective for phishing websites detection. Research workers have implemented a range of methods for phishing website detection. Those strategies deviate from blacklist/whitelist models to machine learning based techniques. The significant models of machine learning techniques used in the detection process are

shown here. Altogether, supervised algorithms alone were used in the majority of the existing systems. Optimistic techniques such as online learning and deep learning remain unexamined.

3. Methodology

The framework in figure 1 represents the module description of the analysis.

Figure 1. Block diagram.

3.1. Dataset

In this model, we have used a phishing dataset from various online websites like Kaggle and some data sets are produced on our own. A 20% of phishing dataset from Kaggle is used to test our model and then the other 80% of the dataset is used for model training. The dataset comprises 95911 rows and 12 columns of phishing and legitimate website data [9].

3.2. Data preprocessing

Data preprocessing consists of cleansing, instance selection, feature extraction, normalization, transformation, etc. The results of data preprocessing is that the absolute training dataset. Data pre-processing may impact how results of the ultimate processing is interpreted. Data cleaning could be a step where filling the missing data, smoothing of noise, recognizing or removing outliers and resolving incompatibilities is done. Data Integration may be a method where the addition of certain databases, or data sets is done. Data transformation is whereby collection and normalization are performed to measure a particular data. By doing data reduction we can achieve an overview of the dataset that is very small in size but, which helps to produce the identical outcome of the analysis [10].

3.3. Exploratory Data Analysis

A technique in data analysis that provides more than one method that is primarily diagrammatic is known as Exploratory Data Analysis (EDA) as shown in Figure 3. It maximizes the perception of a data set, unveil the hidden structure, excerpt essential parameters, locates outliers as well as anomalies and test hidden presumptions.Figure 2 shows the Heatmap.

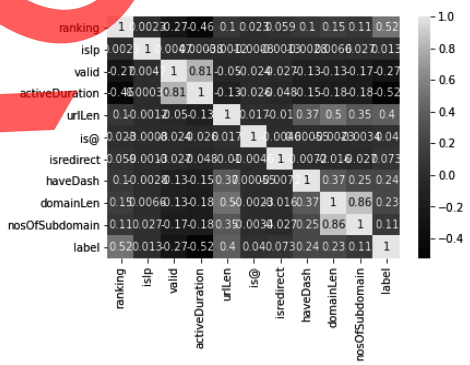

Figure 2. Heatmap.

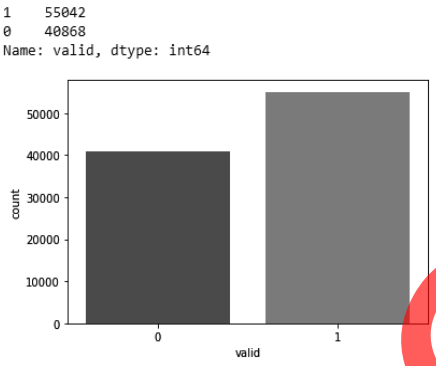

Figure 3. Exploratory Data Analysis

3.4. Train-test split

The dataset is part into two subsets as testing set and training set so that the training dataset can be equipped with the algorithms and then used for detecting the phishing websites on testing dataset. 30% of the data is reviewed for the testing set so that the training model will train and learn the data effectively.

3.5. Machine learning Algorithms

3.5.1. Logistic Regression. Logistic regression is a mathematical approach used to foretell the probability of binary response based on one or more independent variables. It means that, given certain factors, logistic regression is used to foretell a result that has two values such as 0 or 1, pass or fail, yes or no etc. Like the rest of other regression models, the logistic regression is a prognostic analysis. It is accustomed to illustrate data and to point out the association between one dependent binary and one or more nominal variables, ordinal and interval or ratio-level independent variables. It also needs a more complex cost function. This cost function is known as the 'Sigmoid function' or 'logistic function' in place of a linear function. The hypothesis of this algorithm leans to the limit of the cost function between 0 and 1 as shown in Equation (1).

$$0 \leq h_\theta(x) \leq 1 \tag{1}$$

3.5.2. K Nearest Neighbor. The k-nearest neighbors (KNN) algorithm is the easiest algorithm. It can be accustomed to do classification-based and regression-based problems. It is frequently used in image recognition technology, simple advisable systems and decision-making systems. Online platforms like Amazon or Netflix use KNN to suggest a variety of books for the users to buy a product or to watch movies. KNN works based on the well-established mathematical concepts. The first thing to do while implementing KNN is to convert data elements into their accurate values. That's how it works by pointing out the space between the numerical rate of these points. The popular method to figure out the above distance is by using Euclidean distance as explained hereunder as shown in Equation (2).

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}. \tag{2}$$

KNN executes the above given formula to find the intermediate distance between every single data point and the testing data. Then it computes the possibility of points which are identical to the testing data and classifies them based on what all points give the best probability. While using KNN for classification, the outcome is computed from the class which has the best frequency from the K closest resembling instances. Class probabilities are computed using the standardised prevalence of cases that is a component to each of the class in the set of K closest resembling cases for every new data

instance. In a binary classification, a class can be either 0 or 1.
p(class=0)=count(class=0)/(count(class=0)+count(class=1))
Bonds can be separated constantly by increasing the value of K by 1 and consider the group of the further most identical instances in the train set.

3.5.3. Decision Tree. A decision tree is a tree-like structure where an inward node means a parameter, the branch means a decision command, and leaf node is the conclusion. The uppermost node in a decision tree is called the root node. It partitions based on the parameter value. It splits the tree in a recursive manner which is known as recursive partitioning. This diagrammatic figure 4 helps you in making the decision. It's conception like a flowchart diagram equals the human aspect of thinking. This is the reason why these decision trees are easier to comprehend and to explain. It is a white box type of algorithms in machine learning. It describes the inward logic of decision-making. Its training time is quicker when compared to a neural network algorithm. The decision tree's time complexity is based on a function of the number of records and parameters in the taken dataset. The decision tree is a dispense-free or non-parametric method, which does not rely on probability distribution inference. They are capable of handling major dimensional data with best accuracy. It easily identifies non-linear patterns and it also requires only a few data preprocessing, in other words, there is no necessity to normalize columns. It is also used for feature engineering such like finding missing data.

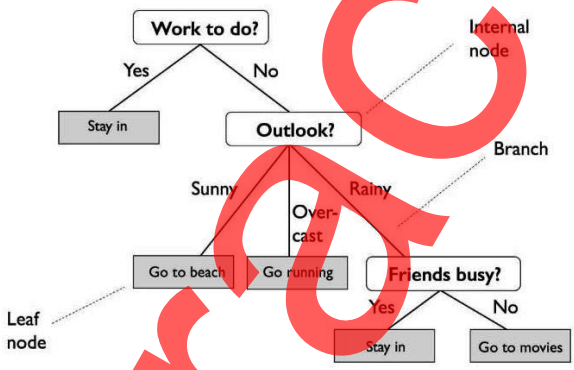

Figure 4. Decision tree example.

3.5.4. Random Forest. Random Forest is known to be a very potential and flexible supervised machine learning technique that merges many decision trees to form a "forest." It is applied for classification problems and regression problems. It supported the idea of ensemble learning, that could be a procedure of mixing many classifiers to unravel a posh difficulty and to enhance the work of the model. Random Forest combines multiple decision trees for a more precise prediction. The idea behind the Random Forest model is that multiple models together perform much superior than one model performing alone. When Random Forest is implemented as classification, each tree gives a vote. The forest picks the classification which has the most number of votes. Whereas while implementing Random Forest for regression, the forest takes the results of all the trees. The distinct decision trees may generate errors but the bulk of the trees are correct, thus accepting the common result within the right guidance .It takes reduced training time when compared to other techniques. It predicts the result with high accuracy. It runs effectively even for big datasets. It also maintains the accuracy when an outsized data is missing. Bootstrap performs row sampling and makes sample datasets for each model. Aggregation minimizes these sample datasets into a brief statistics to observe and combine them. Variance is an oversight coming from liable to small variations in the dataset used for training. High variance trains inapplicable data or noisy data in the dataset rather than the expected results which is known as signal. This difficulty is named overfitting. An overfitted model will perform better in training, but it cannot distinguish the noise from the signal in a testing. Bagging is a technology of the bootstrap technique to a high difference. Overall, Random Forest is faultless, effective and remarkably quick to develop. Figure 5 shows the Random Forest Simplified.

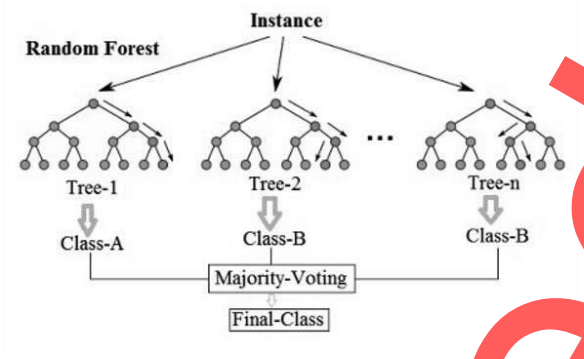

Figure 5. Random Forest Simplified.

3.5.5. XG Boost. XG Boost stands for eXtreme Gradient Boosting. It is an application of gradient boosted decision trees, which is intended for its speed and performance. Boosting is an ensemble learning method where advanced techniques are included in order to rectify the errors made by the already proposed models. Models are included consecutively till we find that no additional enhancement can be carried out. While adding new models it uses a gradient descent technique to minimize the loss. The application of this algorithm is to provide efficient computational time and memory supplies. The aim of this design was to produce the best necessity of the accessible sources to train the model. Execution Speed and Model Performance are the two main reasons to work with XG Boost. This approach can support both classification and regression models.

3.5.6. AdaBoost. AdaBoost is a shortened form for Adaptive Boosting. It is a meta machine learning technique. The significant advantage of the AdaBoost model is it is very fast, simpler and easier to perform. The algorithm has the pliability to be integrated with any machine learning algorithms. It has the ability to transform weak learners or weak predictors to strong predictors so as to achieve the solution of classification.

3.6. Deep learning Algorithms

3.6.1. Artificial Neural Network. Artificial Neural Network (ANN) is an approach that was influenced from the human nerve system that lets it learn by taking instances from data which tells the decision procedure. ANN is able to set up a trial association between independent variables and dependent variables. The indirect information as well as complicated understanding is extracted from the dataset. The association between independent variables as well as dependent variables can be formed without any presumptions about the statistical depiction of the aspect. It contributes positive gains on regression algorithms which includes its competence to act with noisy data. ANN consists of one or more layers of hidden nodes which includes input nodes layer and output nodes layer. Input nodes layer move the details to hidden nodes layer by sending stimulation functions and then the hidden nodes layer ignite or stay inert based on the proof handed over. The hidden layers use the weighted functions to the proof and when the value of a specific node or set of nodes in this layer gets to a limit a value is proceeded to one or more nodes of the output layer. The ANN model has to be trained with a huge number of datasets. ANNs are not suitable for infrequent or utmost events where data is inadequate in order to train it. ANNs do not permit the embodiment of human mastery to be substitutive for perceptible proof.

4. Implementation

Anaconda environment is used to implement the work and, in this work, the dataset is taken from "Kaggle" website. This dataset is first and foremost checked for all the missing, duplicate and noisy values for finer performance of the model. Exploratory data analysis for the dataset is done using modules matplot and seaborn. The parameters of string type are encoded to integer datatype using LabelEncoder. Then, the dataset is divided into 2 sub-datasets; such as the training dataset and testing

dataset in the ratio of 70:30. The classifier models are implemented using the module sklearn in python, the models are tested and trained using the data obtained and the accuracy of respective algorithms is calculated. The ensemble algorithms are implemented using the module ensemble. Likewise, all the models are tested and trained. A deep learning algorithm is also trained and tested using the tensorflow in the backend and using the module keras. The accuracy comparison of all algorithms using data visualization is done at the end using the module chart-studio.

5. Result

The phishing website detection model has been tested and trained using many classifiers and ensemble algorithms to analyze and compare the model's result for best accuracy. Each algorithm will give its evaluated accuracy after all the algorithms return its result. each is compared with other algorithms to see which provides the high accuracy percentage as shown in Table 1. Each algorithm's accuracy will be depicted in the confusion matrix for greater comprehension. The dataset is also trained using a deep learning algorithm. The final accuracy comparison of algorithms is shown in Figures 6 and 7.

Table 1. Comparison Table.

S.No	Algorithm	Training set accuracy	Testing set accuracy	Precision Score
1	Logistic Regression	79.00	79.00	82.30
2	K Nearest Neighbor	96.70	93.10	93.84
3	**Decision Tree**	**100**	**95.50**	**96.13**
4	Random Forest	95.00	94.40	94.81
5	XG Boost	93.80	93.40	93.92
6	Ada Boost	87.00	86.90	84.71

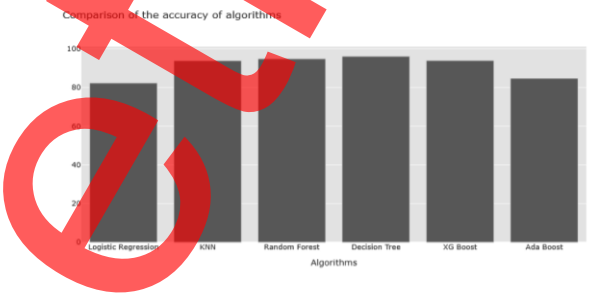

Figure 6. Comparison of all ML algorithms.

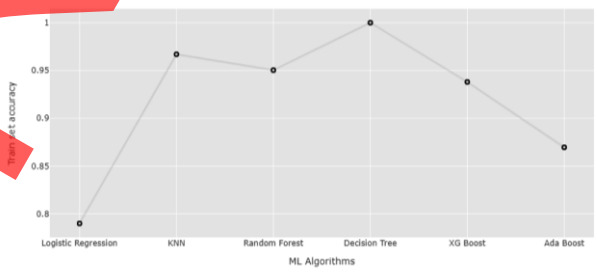

Figure 7. Comparison graph.

6. Conclusion

With the obtained result, it can be concluded that the Decision Tree model provides the best and highest accuracy. Whereas ensemble algorithms on the other hand have also been proved to be convenient as they are fast in speed and performance and are using more than one classifiers for prediction and also gives better performance. Nowadays, website phishing is more damaging. It is becoming a big threat to people's daily life and networking environment. In these attacks, the intruder puts on an act as if it is a trusted organization with an intention to purloin liable and essential information. The methodology we discovered is a powerful technique to detect the phished websites and can provide more effective defenses for phishing attacks of the future.

References

[1] Das, Avisha, et al. 2019 SoK: a comprehensive reexamination of phishing research from the security perspective. IEEE Communications Surveys & Tutorials 22.1

[2] Wu, Jiajing, et al. 2020 *Who are the phishers? phishing scam detection on ethereum via network embedding*. IEEE Transactions on Systems, Man, and Cybernetics: Systems

[3] Chen, Sen, et al. 2019 *GUI-squatting attack: Automated generation of Android phishing apps*. IEEE Transactions on Dependable and Secure Computing

[4] Li, Qi, et al. 2020 *LSTM based Phishing Detection for Big Email Data*. IEEE Transactions on Big Data

[5] Allodi, Luca, et al. 2019 *The need for new anti-phishing measures against spear-phishing attacks*. IEEE Security & Privacy 18.2

[6] Niu, Xiaofei, Guangchi Liu, and Qing Yang 2020 OpinionRank: *Trustworthy Website Detection using Three Valued Subjective Logic*. IEEE Transactions on Big Data

[7] D. Devikanniga, A. Ramu, and A. Haldorai, Efficient Diagnosis of Liver Disease using Support Vector Machine Optimized with Crows Search Algorithm, EAI Endorsed Transactions on Energy Web, p. 164177, Jul. 2018. doi:10.4108/eai.13-7-2018.164177

[8] H. Anandakumar and K. Umamaheswari, Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers, Cluster Computing, vol. **20**, no. 2, pp. 1505–1515, Mar. 2017.

[9] Guillod, Thomas, Panteleimon Papamanolis, and Johann W Kolar 2020 *Artificial neural network (ANN) based fast and accurate inductor modeling and design*. IEEE Open Journal of Power Electronics **1**: 284-299.

[10] Zhang, Shichao, et al. 2017 *Efficient KNN classification with different numbers of nearest neighbors*. IEEE transactions on neural networks and learning systems