International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 4527 - 4544

Optimal Features Based Phishing URL Detection Using Decision Tree - Learning Approach for Web Applications

Mr. M. Sathish Kumar#1 , Dr. B.Indrani*2
#1 Doctoral Research Scholar, Department of Computer Science, Directorate of Distance Education, Madurai Kamaraj University, Madurai, Tamilnadu, India.
E-mail: sathish.friends89@gmail.com
2, Assistant Professor & Head(i/c), Department of Computer Science, Directorate of Distance Education,*
Madurai Kamaraj University, Madurai, Tamilnadu, India.

Abstract

In the field of phishing site identification, numerous methodologies are considered to recognize the phishing URL, for example, the blacklist technique and some machine learning models. Yet, it's having some impediment in the training and testing process. To defeat issues, we will build up the inventive model to distinguish the fake or malicious or phishing URL site. Generally, the databases are acquired is data slicing, and afterward the for the most part entire database partitioned into two sections (i) Training and (ii) Testing. Our proposed, fake detection process extricate the essential features from chosen training database, the feature like using IP address, request URL, page setup, login details, lexical features (like URL length, prefix/suffix so on.), and some different features. From that features, optimal features is selected based on Discrete Bat Algorithm (DBA). These optimal features can be easily detecting the fake URL sites by performing the fitness as similarity measure. When the optimal features are getting from the trained URL site, the classification techniques Decision Tree (DT) is utilized to detect the considered site is phishing, non-phishing or suspicious. Performance measures of the proposed work is analyzed such as confusion matrix based precision, recall and F-measure.

Keywords: *Phishing, URL, Features, optimization, detection malicious and fake profile*

1. Introduction

Phishing is carried out by deceiving the online user into visiting a fake website that impersonates a target legitimate site [1]. Due to the fast developments of the worldwide networking and correspondence technologies, heaps of our day by day life activities, for example, social networks, electronic banking, e-commerce, etc. are transferred to the cyberspace [2]. Social engineering has been formed by challenging dangers to the two individuals for personal data and associations [3] for customer data. Attackers use e-mail or text messages [5] to send false security alerts or prize data to trap users into tapping on the phishing webpage and submitting basic personal data. Phishes use a number of different social engineering and e-mail spoofing plays to endeavor to trap their unfortunate casualties [6]. For detecting the phishing URL, some classifier models are used; phishing URL classification scheme based on investigative the suspicious URL and accelerate the running time of the system. Therefore machine learning based phishing detection system which uses the URL features and analyzed it utilizing naive Bayesian and SVM classifiers [7-8]. All data mining techniques, Decision Tree (DT) is very essential, it's a famous supervised learning process that is used for classification as well as for regression errands [9].

This classifier repetitively divides the preparation dataset into the subparts to identify the separation lines in a treelike structure [10].With DT characterizing phishing URLs and showed that the mix of host-based and lexical features results in the highest classification exactness [11]. Selecting the efficient features, of phishing URL detection, an optimization technique is involved, optimization means a number of ways is there, for fathoming the objectives, some inspired optimizations, for example, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Bat Algorithm (BA, etc [12-13]. By using the ideal features the

proposed classification calculation is to detect the phishing URL. However, the rule works based on human experience rather than an intelligent data mining technique [14]. With the prevalence of machine learning, phishing detection has concentrated on the utilization of machine learning algorithms. To defeat a few issues in the most recent decade specialists have connected new innovative algorithms for malignant URL detection. Utilizing the training data with the suitable feature representation, the subsequent stage in structure the expectation model is the real preparing of the model. There are a lot of grouping calculations can be legitimately utilized over the training data (Naive Bayes, Support Vector Machine [19], Logistic Regression [16], and so forth.). Notwithstanding, there are sure properties of the URL information that may make the training difficult (both as far as versatility and learning the suitable idea). Clearly, these techniques are not constant and may cost a great deal of time and exertion. Subsequently, the traditional time for customary models might be too high to possibly be pragmatic. This strategy was not proficient and conservative since it uses the previous information of the URL, which has a low detection speed and a greatest expense.

The issue with this procedure is its all-out reliance on the outsider. Basically, when the last is stable an attack, the security of the whole framework is undetermined. Impel by the assumption, we minimize the phishing collection issue as the content classification issue. Some classifier and optimization models like SVM, ANN, KNN, and PSO, GA [15, 16, 18, 19, 20, 21] take care of classification and issues just as regression issues. In NN [18], the number of hidden layers, the ideal number of neurons in each hidden layer and distinctive pair of the transfer function for hidden layer and output layer, the training work by utilizing this capacity to identify the given URL is phishing or non-phishing. In any case the spammers, phishers have opposed to these security strategies and advance to the new methods that can't be distinguished by the conventional procedure. A standout amongst the most predominant system that spammer is utilizing in the SNS is a short URL procedure. So, in this study presents the new technique to detecting the phishing URL with new innovative techniques. The main contribution of the paper is demonstrated as below:

- Primarily, the databases are gathered from URL phishing web site database which contains more features.
- Initially, the features are extracted from the initial database to get a more effective detection model.
- The extracted databases are optimized and the optimal features are selected with the help of DBA.
- Then the phishing and non phishing URL are classified with the help of Decision Trees.
- Optimal legitimate and fraudulent web page URLs and build our very own dataset, at long last tree based classifier utilized to detect the givenURL, is malicious or begin.
- The performance of the proposed work is measured by some performance measures such as precision, recall, and F-measure with high detection rate.

1.1 Attackers in URL Detection

Attackers utilize various kinds of techniques not to be detected at all by security mechanisms or system administrators [12]. In reality, the connection to the website is dexterity to complete the attack, making it very hard to mark lacking expert information. Companies regularly embattled by phishing attacks, for example, Citibank, eBay, and US Bank don't use advanced signatures by any stretch of the imagination. Compromised URLs that are utilized for cyber-attacks are expressed as malicious URLs. Truth be told, it was noted that close to 33% of all websites are potentially vindictive in nature [2], demonstrating wild use of pernicious URLs to perpetrate cyber-crime. Noxious SNSs provider or malignant member with the suitable set of concession can be capable of the track the correspondence. It is very rigid to recognize correspondence attack.

2. Recent Literature Analysis

"Machine learning based phishing detection from URLs" by (Sahingoz, O. K et. al.2018) [15], an instantaneous anti-phishing system that utilizes seven various classification calculations and Natural Language Processing (NLP) based features, was proposed. For calculating the presentation of the system, an innovative dataset was created, and the experimental results were experienced on it. As per the experimental and comparative results from the implemented classification calculations, Random Forest calculation with NLP based features. Harmony Search (HS) which was deployed based on nonlinear regression technique and Support Vector Machine (SVM). The nonlinear regression approach was used to order the websites, where the parameters of the proposed regression model were obtained utilizing HS calculation by (Babagoli, M, et. al.2018) [16], A method to order the Uniform Resource Locator (URL) into Phishing URL or Nonphishing URL.

Securing the web interface expects solutions for an arrangement with dangers from mutually specialized vulnerabilities and social components. Phishing assaults are a standout amongst the most generally abused vectors in social building assaults (Mao, J et. al.2019) [17]. The learning-based collection investigation is to prefer page layout similitude which is utilized to recognize phishing pages. The static investigation uses classification algorithms in AI to distinguish certain Benign and malicious site pages. Artificial Neural Network has been trained by ANN-PSO (Gupta, S., and Singhal, A.2017) [18]. A dissimilar proportion of learning and different enactment works on a number of hidden layers, output layer. (Zouina, M., &Outtaj, B. 2017) [19] Features are the URL measure, the number of hyphens, the number of dashes, and thenumber of numeric characters in addition to a discrete variable that compares to the nearness of an IP address in the URL lastly the comparability record.

Ramanathan, V. et al. [25] in 2018 had exhibited vigorous server side procedure to distinguish phishing attacks, called phishGILLNET, which incorporates the intensity of natural language processing and machine learning systems. phishGILLNET was a multi-layered way to deal with recognize phishing attacks. The presentation of phishGILLNET1 was assessed by utilizing PLSA overlay in system and the classification was accomplished utilizing Fisher similarity. Also, phishGILLNET3 required a little rate (10%) of data be clarified in this manner sparing critical time, work, and keeping away from errors brought about in human comment. Zouina, M. et al. in 2017 [26] had recommended that a novel lightweight phishing detection approach totally dependent on the URL (uniform asset locator). The referenced framework creates a fantastic recognition rate which is 95.80%. Jain, A.K et al. [27] in 2016 had explored to ensure against phishing attacks utilizing auto-refreshed white-list, of authentic destinations gotten to by the individual client. The proposed methodology has both quick access time and high detection rate. At the point when clients attempt to open a site which isn't accessible in the white-list, the program cautions clients not to reveal their touchy data. The outcome demonstrated four prominent machine learning classifiers on their exactness and the components influencing their outcomes. In 2016, Iuga, C et al. [28] directed a web-report with 382 members that concentrated explicitly on recognizing factors that help or ruin internet clients in recognizing phishing pages from genuine pages. The creators thought about relationships between statistic qualities of people and their capacity to accurately recognize a phishing attack, just as time-related components. Given that solitary 25 % of the members accomplished an identification score of more than 75 %. In 2019, Li, Y. et al. [29] had exhibited the stacking model to identify phishing site pages utilizing URL and HTML features. Regarding features, the creators structured lightweight URL and HTML includes and presents HTML string installing without utilizing the outsider administrations. The proposed methodology outflanked many machine learning models on different measurements, accomplishing 97.30% on precision, 4.46% on missing alarm rate, and 1.61% on false alarm rate on 50K-PD dataset.

Author	Database	Features	Algorithms	Parameters	Pros	Cons
Sahingoz, O. K. et al. [15] 2018	(Ebbu2017 Phishing Dataset, 2017)	Natural Language Processing (NLP) based features	Seven different classification algorithms	Precision, sensitivity, f-measure, and accuracy	NLP based features gives the best performance with the 97.98% accuracy rate for detection of phishing URLs	Checking these features need some extra time, therefore they may not be preferred for real-time detection.
Babagoli, M et al. [16] 2018	UCI Datasets	30 features	harmony search (HS) SVM decision tree and wrapper	Precision, recall, f-measure accuracy	Nonlinear regression based on HS led to accuracy rates of 94.13 and 92.80% for train and test processes. HS results in better performance compared to SVM.	Less efficiency
Mao, J., et al. [17] 2019	real-world web page samples from phishtank.com	page similarity based features	Support Vector Machine (SVM), Decision Tree, AdaBoost, and Random Forest	Precision, recall, f1 score	accurate and effective in determining similarity from page layout enhance the performance of existing anti-phishing mechanisms	Minimum accuracy

Gupta, S et al. [18] 2017	UCI Repository	31 attributes	ANN_PSO	Accuracy and RMSE	Minimize the root mean squared error good accuracy	ANN is a cost effective technique
Zouina, M. et al. [19] 2017	Synthetic dataset	URL size, the number of hyphens, the number of dots, the number of numeric characters plus a discrete variable	SVM	Recognition rate	very satisfying recognition rate which is 95.80% detection systems improves the overall recognition rat	unsatisfactory recognition rate exceptional impact of the similarity index

3. The Phishing Detection Approach

We built up the optimal feature based examination to distinguish the malicious or phishing website pages (URL). Present days, the phishing has expanded greatly over the recent years and it is a genuine risk to global security and economy, so just we are examined innovative strategy. Our proposed phishing URL detection model comprises two critical stages that are (i) Training and (ii) Testing. In training stage, a machine learning algorithm is utilized to remove the features of the training dataset to create a classifier. At that point, the testing stage, the features of test dataset likewise are extracted, and these features are sustained into the setup classifier. URL search phase, a few features are considered for the malicious recognition process. The implementation strategy formed by three primary modules, for example,

- URL Database Module
- Extraction of Feature Module
- Optimization Module
- Detector Module

From these above modules, the malicious URL is identified once the features are extracted. For upgrading the recognition dimension of phishing site pages, motivated DBA is used to choose the optimal attribute. When the optimal feature is getting from the trained URL site, the classification system Decision Tree (DT) is used to identify whether the considered site is phishing, non-phishing or suspicious. The detail clarification of our technique is discussed in beneath subsection.

4.1 Static URL Database

In the proposed work, two datasets are viewed as which is acquired from the UCI - Machine Learning Repository; contains the Phishing Web Site Dataset. One dataset comprises of 30 features and 1 target feature [20]. At the point, when a site is viewed as suspicious that implies it very well may be either phishy or non-phishing, which means the site held some genuine and phishy features. The syntax for one sample URL is delineated as underneath figure 1.

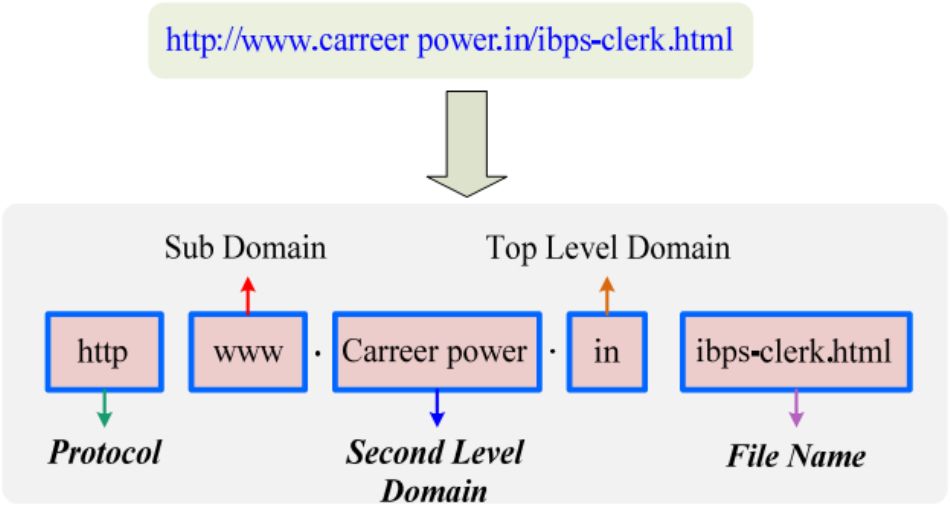


Figure 1: URL Syntax

4.2 Extraction of Features

A malicious website page is bound to be conveyed with a malicious URL. Along these lines, rather than coordinating examples, we approach the URL as an organized string and concentrate a portion of its features. Table 1 clarifies the list of phishing features in six unique areas that are Domain personality and URL, Encryption and security, Java content and source code, Contents and page style, Web address bar and Social human factor. The description of each features are depicted in table 1.

Table 1: List of Sample Feature sets

Domain identity and URL	
Features	**Description**
via IP address	Using IP address instead of domain name in websites can show that somebody wants to steal the information
Require URL	In the phishing websites, the objects are loaded from different domains. If more than 66% of objects are loaded from various domains, the feature is regarded as fraudulent
URL of anchor	Links in the websites are placed in <a> tags. If more than 61% of the anchor tags are irrelevant to the webpage name, the feature is defined as phishing
DNS details	For phishing websites, the DNS record is not recognized in WHOIS datasets
Strange URL	
(ii) Encryption and security	
Features	**Description**
SSL certificate	The certificate of SSL
Certification authority	The certificate authority of URL is detected

Strange cookie	The cookies present in the web page
Distinguished names certificate (DN)	The distinguished names certificates
(iii)Javascript and source code	
Features	**Description**
Redirect pages	Code has redirect pages
Straddling attack	URL has straddling attacks
Pharming attack	URL has Pharming attacks
Using on Mouse Over Server form handler (SFH)	IURL has using on Mouse Over Server form handler
(iv) Contents and page style	
Features	**Description**
Spelling mistake	URL contains spelling mistakes
Replicating a website	URL has replicating a website
‘‘Submit’’ button	The page contains the submit button
Via pop-up windows	Windows has pop-up windows
Disabling right-click	If the URL has disabling right click option
(v) Web address bar	
Features	**Description**
Long URL address	Phishers used to use long URLs to hide fake addresses. This feature is to compute the URL length
Replacing similar characters for URL	Maintaining similar character in URL
Adding prefix or suffix	Whether URL has _-‘
Using the ‘@’ to confuse	Whether URL has _@‘, _//‘
Using hexadecimal character codes	Whether URL has hexadecimal codes
(vi) Social human factor	
Features	**Description**
Much stress on security and response	Whether URL has such stress on security and response
Generic welcome	URL generates generic welcome
Buying time to log on accounts	Phisers used buying time to log on accounts

It is essential to choose a decent subset of beginning features to recognize malicious pages from benign pages. Correlation-based Feature Selection (CFS) is utilized to choose the most agent subset of features since it assesses and positions subsets of features free of the succeeding classification model. The CFS function uses to assess subsets of features is as per the following condition:

$$F_s = \frac{n r_{cf}}{\sqrt{n + n(n-1) r_{ff}}} \qquad \ldots\ldots\ldots (1)$$

The subset of features can be described as the ratio of mean feature class relationship to the average feature-feature class relationship. The expansion of the above equation is F_s the heuristic “merit” of a

feature subset S that contains n features r_{cf} is the mean feature-class relationship (f ∈S), and r_{ff} is the average feature-feature inter-correlation. The denominator of the formula gives a sign of how much redundancy there is among the features. Figure 2 shows the structure of proposed phishing URL system.

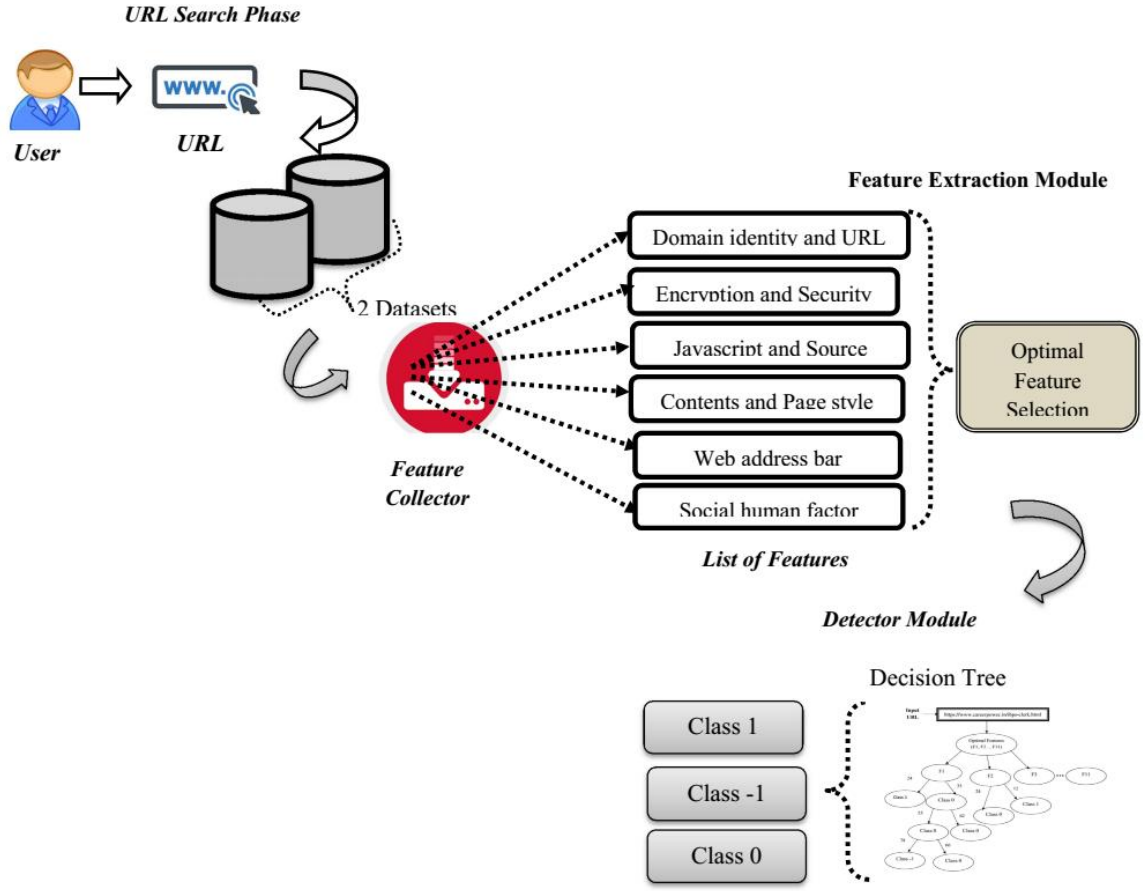


Figure 2: Proposed Phishing URL Detection System

4.3 Optimal Feature selection

After the formation of the feature matrix, we utilized subset based feature selection strategies to identify the most noticeable features. Feature selection techniques are useful to optimize the dataset measurement by expelling excess and insignificant features as for the learning stage in the investigation. In the wake of applying this, we got another feature matrix with a different number of features. For optimal feature selection, we present one meta-heuristic based methodology called DBA. This algorithm accepts the extracted features as a contribution, from that most noticeable feature are chosen to utilize DBA.

BAT Algorithm (BAT): In nature, bats are captivating creatures. Microbats utilize a sort of sonar, called echolocation, to identify prey, stay away from deterrents, and find their roosting crevices in obscurity. By idealizing a portion of the echolocation qualities of microbats, BAT is proposed. Introduce the bat populace with velocity V_j at position P_j emitting a fixed frequency $Bf_{\min}$, varying wavelength λ, and loudness L_j to search for prey (target) [21].

The discrete function of BAT: Instead of nonstop optimization, the factors utilized in the scientific program (or some of them) are confined to expect just discrete qualities, for example, the whole numbers.

i) Initialization: The underlying populace is created randomly for n number of bats. Every person of the populace comprises genuinely valued vectors with d measurements. Introduced the bat populace (features of URL) as

$$B_{0\,j,k} = B_{0lb_k} + rand(0,1)(B_{0ub_k} - B_{0lb_k}) \text{ ..... (2)}$$

Where $j = 1,2,..n$, $k = 1,2,..d$, B_{0ub_k} and B_{0lb_k} are upper and lower boundaries for dimension k respectively.

ii) Similarity index: It is chosen as the fitness function for optimal feature selection process: In order to measure the similarity or regularity between the input URL and other URLs, distance metrics plays a very important role.

$$\text{Distance}_{AB} = \min_k \left| F_{ik} - F_{jk} \right| \quad \text{..... (3)}$$

The importance of similarity measure: It is important to distinguish, in what way the URLs are interrelated, how different information disparate or comparative with one another and what measures are considered for their examination.

Objective Function $OF = max(\,similarity\ among\ URL\,)$ ..... (4)

The updating process is repeated until the objective function (optimal features with minimum similarity) is achieved.

iii) New Solution Updating Process

a) *Movement of virtual bats:* In simulations, virtual bats are used. The new solutions $P_j^{\,t}$ and velocities $V_j^{\,t}$ at time step t are given by

$$Bf_j = Bf_{\min} + (Bf_{\max} - Bf_{\max}) * \alpha_j \text{ ..... (5)}$$

$$V_j = V_j^{\,t-1} + (P_i - s_0) * Bf_j \text{ ..... (6)}$$

$$P_j^{\,t} = P_j^{\,t-1} + V_j^{\,t} \text{ ..... (7)}$$

The term α_j is an irregular vector which is drawn from a uniform appropriation. From the condition s_0 is the present worldwide best solution. When an answer is chosen among the present best solutions, another solution for each bat is created locally utilizing random walk is given by

$$S_{new} = S_{old} + \eta * L^t \text{ ..... (8)}$$

Where $\eta \in [-1,1]$ is a random number and $L^t = < L_j^{\,t} >$ is the average loudness of all the bats

b) Loudness and pulse rate: Moreover, the loudness L_j and the pulse emission rate E_{p_j} have to be updated accordingly as the iterations proceed. As the loudness, usually, decreases once a bat has found

its prey, while the rate of pulse emission increases. Now a bat has just found the prey and the loudness and pulse emission rate updation is given by,

$$L_j = \beta * L_j^{\ t} \text{ ..... (9)}$$

$$E_{p_j}^{\ t+1} = E_{p_j}^{\ 0}[1 - e^{-\gamma * t}] \text{ ..... (10)}$$

β, γ are the constants and the ranges are $0 < \beta > 1$ and $\gamma > 0$

Their loudness and emission rates will be updated only if the new solutions are improved, which means that these bats are moving towards the optimal solution.

Pseudo Code of DBA

Input: URL features
Output: Optimal features

```
Initialize the URL features as input
Define bat population, pulse frequency, rates, and the loudness
Generate the discrete function of initialized solution
        While (t < iteration count)
        {
        Find a new solution based on an objective function
        Execute equation (1)
       Update the best solution (equation (2) and (4))
              if (rand> rates)
              {
               Select a solution among the best solutions
               Generate a local solution around the selected best solution
              }
End if
     Generate a new solution by flying randomly
    if (rand< loudness& f(position) < f(best solution))
     {
       Accept the new solutions
        Increase pulse rate and reduce loudness
     }
endif
Rank the bats and find the current best
     }
End while
```

4.4 Optimal Features

In the optimal feature selection phase, we have defined heuristics to extract 7 features from the URL and are subjected to classification to determine the non-phishing, phished or suspicious URL. The 7 selected features (named as F1, F2 … F7) along with class attribute and its conditions are listed in table 2. In this table, the conditions of the optimal features are clearly shown. Class: Based on those extracted features,

we can determine the non-phishing, phishing or suspicious URL. In the proposed work, with the intention of distinguishing those three categories URL is named as non-phishing (1), phishing (-1) and suspicious (0).

Table 2: Optimal Feature set for phishing detection

Optimal Features	**Condition**	**Optimal Features**	**Condition**
F1	$\begin{cases} non-phishy, & URL\ anchor < 31\% \\ suspicious, & URL\ anchor \geq 31\% \& \leq 67 \\ phishy, & otherwise \end{cases}$	F5	$\begin{cases} phishy, & if\ '-'\ symbol\ in\ doma \\ non-phishy, & otherwise \end{cases}$
F2	$\begin{cases} non-phishy, & request\ URL < 22\% \\ suspicious, & request URL \geq 22\% \& < 61\% \\ phishy, & otherwise \end{cases}$	F6	$\begin{cases} phishy, & if\ IP\ address\ exist \\ non-phishy, & otherwise \end{cases}$
F3	$\begin{cases} phishy, & SFH\ if\ 'about:blank'\ or\ empty \\ suspicious, & SHD\ redirects\ to\ different\ domain \\ non-phishy, & otherwise \end{cases}$	F7	$\begin{cases} phishy, & if\ '\bullet'\ in\ domain < 3 \\ non-phishy, & otherwise \end{cases}$
F4	$\begin{cases} non-phishy, & URL\ length < 54 \\ suspicious, & URL\ length \geq 54 \& \leq 75 \\ phishy, & otherwise \end{cases}$	Class	$\begin{cases} non-phishy, & 1 \\ phishy, & -1 \\ suspicious & 0 \end{cases}$

4.5 Detector Module

In the detection module, Decision Tree (DT) is utilized to decide the info URL as three classifications: phishing non-phishing, and suspicious. The optimal feature set is allocated into the classification technique (DT) where the malicious sites are recognized dependent on the conditions in every feature; the URL with optimal features improves the accuracy of the classification task in comparison of applying the classification task on the original URL.

Decision Tree (DT): The DT classification is one of the well-known regulated learning forms that are utilized for classification as well as for regression errands. This classifier tediously partitions the training dataset into the subparts to recognize the division lines in a treelike structure [22]. At that point, these lines are utilized to recognize the proper class for the objective thing. Every decision node parts the information into at least two classifications as indicated by a single attribute value. Each leaf node is doled out to a class (particularly by computing probability) in the classification algorithm. In each dimension, the distinguishing proof of the root attribute is finished by attribute selection measures:

a) Information Gain

When, we utilize a node in a DT to partition the training cases into littler subsets, at that point entropy changes. Information gain [22] is a proportion of this adjustment in entropy as depicted in equation (11).

$$Gain(F, A) = E(F) - \sum_{v \in val(A)} \frac{F_v}{F} . E(F_v) \quad \ldots\ldots\ldots (11)$$

Suppose F is a set of features, A is an attribute, Fv is the subset of F with A = v, and Values (A) is the set of all possible values of A.

b) Gini

Gini Index [22] is a measurement to gauge how regularly randomly chosen element would be erroneously recognized. It implies a property with a lowerGini list ought to be favored in equation (12).

$$Gini = 1 - \sum_i F_i^2 \quad \ldots\ldots\ldots (12)$$

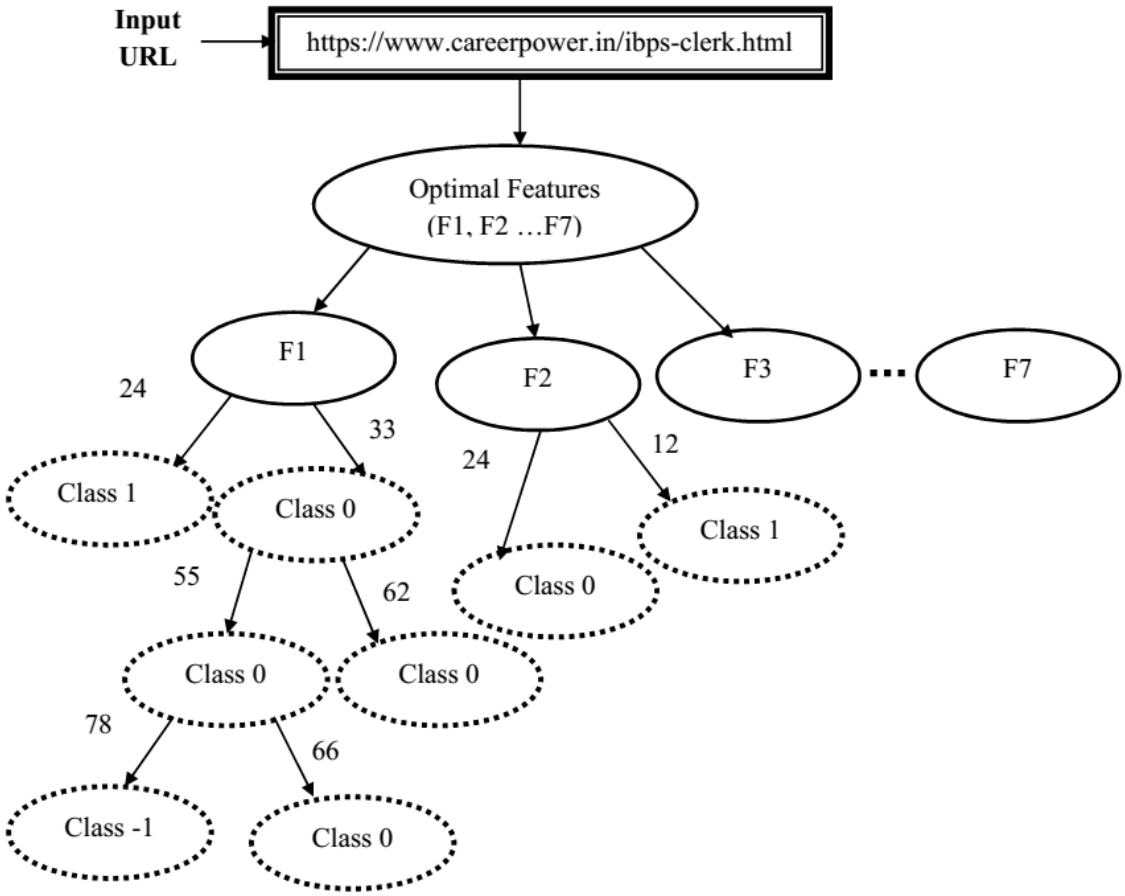


Figure 3: Representation for DT

Figure 3 shows an example for the detection of non-phishing, phishing or suspicious URL using DT. Here, the optimal features F1 (URL anchor) and F2 (request URL) are explained in the tree structure. The URL is distinguished by three classes i.e. phishing, non-phishing and suspicious and the notations are -1, 1 and 0 respectively.

5. Results and Analysis

Our phishing URL detection models are executed in the framework setup of 4GB RAM i3 processor in Windows 10 (64-bit). This feature extraction and selection, detection performed with JAVA and the

classifier model implemented in Net beans 8.0.1, Moreover this proposed model contrasted with other detection and feature selection algorithm with the aid of some performance metrics and database description also examined this area.

5.1 Database Description

In this detection work, two distinctive UCI machine learning repository database, its shows in table 3. A few features of URL which depends on the Address bar. This table depicts the dataset is given underneath, how many numbers of Phishing URLs and how many numbers of non-phishing URLs [23, 24].

Table 3: Database Details

Data Type	Dataset Name	No of Instances	No. of Attributes	Number of Web Hits
Dataset I	Phishing Websites	5456	30	107067
Dataset II	Website Phishing	1353	10	51817

5.2 Performance Evaluation

The efficiency of the proposed method is computed by way of calculate some performance measures.
TP: True positive ***TN:*** True Negative
FP: False positive ***FN:*** False Negative

$$\text{Precision} \quad \frac{TP}{TP+FP} \tag{13}$$

$$\text{Recall} \quad \frac{TP}{TP+FN} \tag{14}$$

$$\text{F-score} \quad \frac{2\times(recall * precision)}{recall + precision} \tag{15}$$

Table 4: Test URL Detection Results of DT- DBA

Measures	Dataset I			Dataset II		
	Class -1	***Class 1***	***Class 0***	***Class -1***	***Class 1***	***Class 0***
TP Rate	0.95	0.943	0.94	0.942	0.92	0.94
FP Rate	0. 069	0.034	0.032	0.047	0.031	0.037
Precision	0.96	0.85	0.94	0.925	0.94	0.89
Recall	0.94	0.92	0.89	0.95	0.914	0.86
F- Measure	0.92	0.94	0.95	0.945	0.924	0.94

Table 4 demonstrates the detection results of the proposed classifier, DT-DBA with greatest classification rate. The features which have been chosen are performed by the DT technique. Each incentive over the bars indicates the accuracy of the DT when the particular feature is picked as the root. Here discussed the two datasets (I&II) detection level, as far as accuracy, recall, and F-Measure by confusion matrix. For dataset, I the precision, recall, F-Measure for class 1 attained 0.85%, 0.89%, and

0.95%. The recall value of the dynamic examination was nearly the value of static investigation. It was exhibited that the two examinations wearable to distinguish malicious web site pages to some degree. Similar, qualities achieved in dataset II the execution measures like 94.5%, 92.4% and 94% for each class F-Measures.

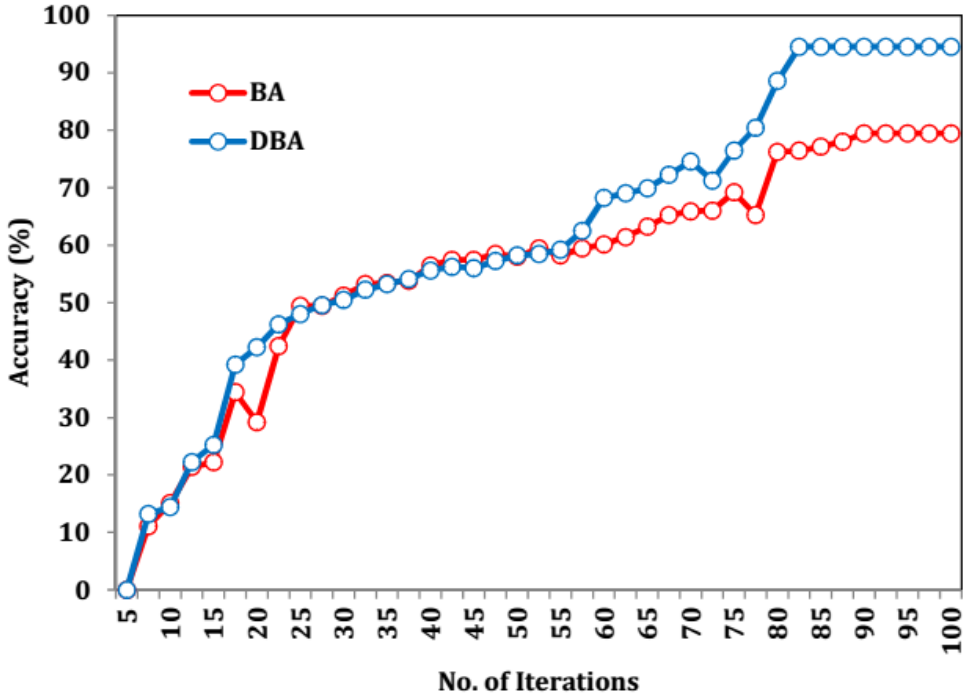


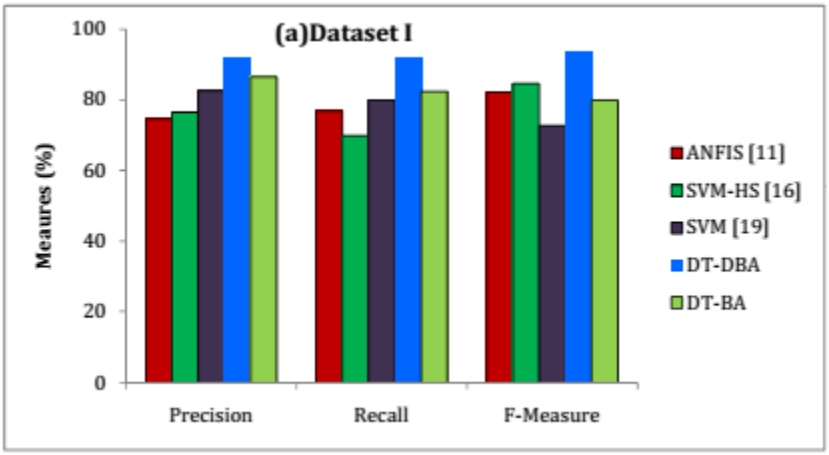
Figure 4: Iteration Vs Accuracy

Figure 4 demonstrates the optimal feature selection model accuracy. Here looked at traditional bat algorithm and our proposed DBA, here dataset I the optimal features are 7 at 76 iterations and 6 features in 80 iterations for dataset II; its shows in table 5. The features utilized in their exploration are assembled from the URL and HTML source of sites. They have utilized 4 grouping strategies for evaluation of the accuracy of the structure. Iteration Vs accuracy in figure 4, the greatest accuracy achieved in 85th emphasis (92.41%). The objective function of optimization is similarity measure, between the phishing site and its objective and different features removed from the URL as input.

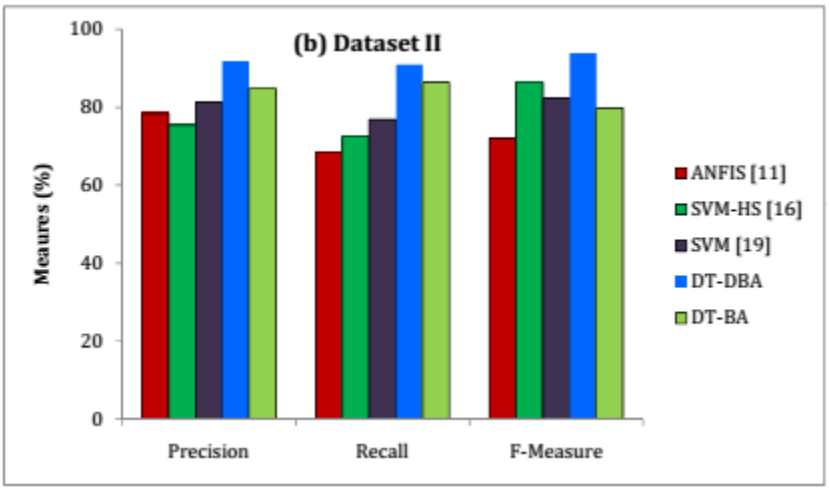
Table 5: Comparative analysis of Optimal Feature Selection

Techniques	Dataset I				Dataset II			
	Number of Features	Number of Iterations	Accuracy	Similarity Index	Number of Features	Number of Iterations	Accuracy	Similarity Index
HS	22	48	89.41	82.08	4	50	72.74	87.26
PSO	18	54	75.55	74.41	8	61	65.74	81.11
BA	20	100	79.45	83.22	7	82	81.48	73.48
DBA (proposed)	7	76	92.12	94.45	6	80	94.22	93.14

In addition feature selection process contrasted with other optimization procedures like PSO, HS, and BA methods. For instance, the most extreme similarity is 94.45 in DBA, it's contrasted with BA, PSO and HS is the thing that matters is beside less 5%. Features of the URL ought to be picked cautiously to show signs of improvement results. That is there as on the first analysis stages valuable even with its static features and classification. In any case, it has turned out to be progressively hard to distinguish malicious pages utilizing just static analysis.



(a)



(b)

Figure 5: Comparative analysis for URL Detection techniques (a) dataset I (b) dataset II

Table 6: Confusion metrics results for training & Testing (DT-DBA)

Train/Test	Dataset I			Dataset II		
	Precision	Recall	F - Measure	Precision	Recall	F - Measure
90%-10%	94.52	88.85	89.74	94.41	92.21	94.25

80%-20%	92.22	92.11	86.1	90.111	86.11	92.77
70%-30%	94.77	94.14	84.42	84.11	89.57	93.5

Comparative investigation of this URL phishing detection with SVM, ANFIS along with the optimizations in figure 5 two database figure 5 (a) for dataset I and (b) for dataset II. All methodologies, by including or evacuating at least one items from the feature list, the accuracy is diminished. The precision of ANFIS is 82.42%, it contrasts with our proposed URL detection, the difference is 4.52% in dataset II, similar in other detection algorithms for URL detection model. At that point table 6 demonstrates the training and testing results of DT-DBA if the training 90% and testing 10% the execution measure is 94.45, 88.85, 89.74 in dataset I, it's contrasted with 80%-20% and 70%-30% the difference is minimal high, similarly dataset II also analyzed. Phishing URL detection utilizing ANFIS and SVM classifiers created the results, it is discovered that when the measure of the training set increases, SVM performs superior to Naïve Bayes classifier to recognize phishing URL.

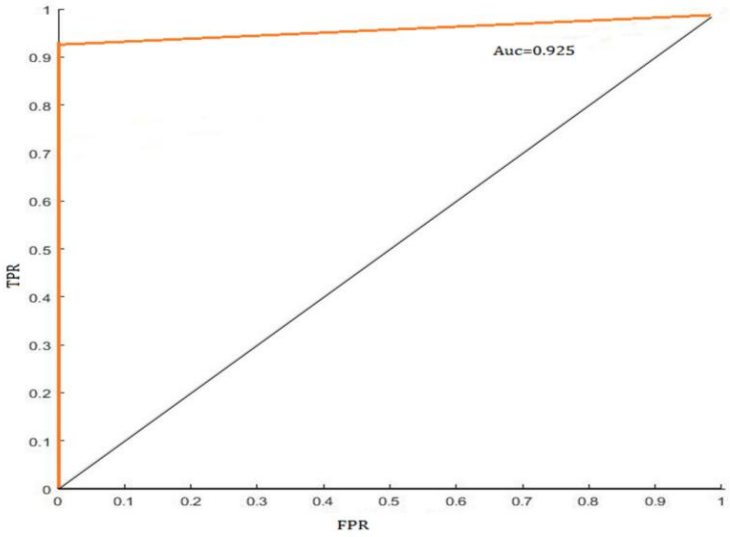


Figure 6: ROC of URL Phishing Detection

ROC curve is connected to look at the execution of URL-Optimal features for DT-DBA approach. Every single malicious sample as positive class and every benign sample as a negative class; moreover, the ROC curve is drawn by the help of TPR and FPR. In a ROC curve, the horizontal axis symbolizes the confusion matrix. The phishing website pages, the optimal lexical features like URL length, estimate in an exertion to the detection model. So figure 6 demonstrates the execution of the proposed detection approach ROC curve (AUC) implies that the chosen subset of features.

6. Conclusion

With the on-going applications of online website security, phishing has become a severe problem. Here, we have proposed a phishing Web page detection method by means of feature extraction and optimal feature selection. For efficient fake URL detection, some features like IP address, request URL, page setup, login details, and lexical features are extracted from the collected two datasets. And then optimal features were chosen by the proposed meta-heuristic approach (DBA) which works based on the features similarity measure. In view of the attained optimal feature sets, we can detect the URL as three categories: Non- phishing, phishing and suspicious by using DT classifier. The proposed fake URL

detection model (DBA-DT) gives maximum accuracy compared to existing work. From the implementation results in our proposed URL Detection the precision, recall, and F-Measure is 91.68%, 91.85% and 93.74%, its maximum performance, compared to other detection techniques. For future investigations, more optimal features are extracted from the datasets, so we can easily detect fake URLs. The optimal features are attained by using hybrid algorithm (combination of two nature-inspired algorithms).

Reference

1. Li, Y., Yang, Z., Chen, X., Yuan, H. and Liu, W., 2019. A stacking model using URL and HTML features for phishing webpage detection. Future Generation Computer Systems, 94, pp.27-39.
2. Babagoli, M., Aghababa, M.P. and Solouk, V., 2018. Heuristic nonlinear regression strategy for detecting phishing websites. Soft Computing, pp.1-13.
3. Karnase, D.K., Mishra, M.G., Dighole, S.H., Shelke, S.R. and Dhanwani, M.D., 2018. A Review on Malicious URL Detection using Machine Learning Systems. International Journal on Recent and Innovation Trends in Computing and Communication, 6(4), pp.214-219.
4. RouthuSrinivasaRao, AlwynRoshanPais,(2019),Jail-Phish: An improved search engine based phishing detection system, Journal of computer security, pp.1-24.
5. Sahoo, D., Liu, C. and Hoi, S.C., 2017. Malicious URL detection using machine learning: a survey. arXiv preprint arXiv:1701.07179.
6. RadhaDamodaram, M.C.A. and Valarmathi, M.L., Phishing website detection and optimization using Modified bat algorithm, Journal of Engineering Research and Applications,Vol.2(1),pp.870-876 (2012).
7. Yi, P., Guan, Y., Zou, F., Yao, Y., Wang, W. and Zhu, T., 2018. Web Phishing Detection Using a Deep Learning Framework. Wireless Communications and Mobile Computing, 2018.
8. James, J., Sandhya, L. and Thomas, C., 2013, December. Detection of phishing URLs using machine learning techniques. In 2013 International Conference on Control Communication and Computing (ICCC) (pp. 304-309). IEEE.
9. Dhanalakshmi, Ranganayakulu, Chellappan,"Detecting Malicious URLs in E-mail – An Implementation",AASRI Procedia,Vol.4,pp.125-131,2013.
10. Sami Smadi, NaumanAslam, Li Zhang (2017),"Detection of Online Phishing Email using Dynamic Evolving Neural Network Based on Reinforcement Learning",Decision Support Systems, pp.1-42.
11. Barraclough, P., Sexton, G., Hossain, A. and Aslam, N., 2014. Parameter optimization for intelligent phishing detection using Adaptive Neuro-Fuzzy. International Journal of Advanced Research in Artificial Intelligence, 3(10), pp.16-25.
12. Sami Smadi, NaumanAslam, Li Zhang(2017), Detection of Online Phishing Email using Dynamic Evolving Neural NetworkBased on Reinforcement Learning,Decision Support Systems,pp.1-42.
13. Chiew, K.L., Yong, K.S.C. and Tan, C.L., 2018. A survey of phishing attacks: their types, vectors and technical approaches. Expert Systems with Applications, 106, pp.1-20.
14. Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A. and Liang, Z., 2018. Detecting Phishing Websites via Aggregation Analysis of Page Layouts. Procedia Computer Science, 129, pp.224-230.
15. HimaniThakur ,Dr.Supreetkaur,"Sahingoz, O.K., Buber, E., Demir, O. and Diri, B., 2019. Machine learning based phishing detection from URLs". Expert Systems with Applications, 117, pp.345-357.
16. Babagoli, M., Aghababa, M.P. and Solouk, V., 2018. Heuristic nonlinear regression strategy for detecting phishing websites. Soft Computing, pp.1-13.
17. Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A. and Liang, Z., 2019. Phishing page detection via learning classifiers from page layout feature. EURASIP Journal on Wireless Communications and Networking, 2019(1), p.43.
18. Gupta, S. and Singhal, A., 2017, August. Phishing URL detection by using artificial neural network with PSO. In 2017 2nd International Conference on Telecommunication and Networks (TEL-NET) (pp. 1-6). IEEE.

19. Zouina, M. and Outtaj, B., 2017. A novel lightweight URL phishing detection system using SVM and similarity index. Human-centric Computing and Information Sciences, 7(1), p.17.
20. Jin-Lee Lee, Dong-Hyun Kim, Chang-Hoon, Lee,"Heuristic-based Approach for Phishing Site Detection Using URL, Features", ntl. Conf. on Advances in Computing, Electronics and Electrical Technology, pp.1-5.
21. ZhihuaCui,Feixiang Li, Wensheng Zhang (2018), "Bat algorithm with principal component analysis", Journal of Machine Learning and Cybernetics,pp.1-20.
22. Bhaskar N. Patel, Satish G. Prajapati and Kamaljit I. Lakhtaria, "Efficient Classification of Data Using Decision Tree ",Journal of Data Mining, Vol. 2, No. 1, pp.6-14,2012.
23. https://archive.ics.uci.edu/ml/datasets/phishing+websites
24. https://archive.ics.uci.edu/ml/datasets/Website+Phishing
25. Ramanathan, V. and Wechsler, H., 2012. phishGILLNET—phishing detection methodology using probabilistic latent semantic analysis, AdaBoost, and co-training. EURASIP Journal on Information Security, 2012(1), p.1.
26. Zouina, M. and Outtaj, B., 2017. A novel lightweight URL phishing detection system using SVM and similarity index. Human-centric Computing and Information Sciences, 7(1), p.17.
27. Jain, A.K. and Gupta, B.B., 2016. A novel approach to protect against phishing attacks at client side using auto-updated white-list. EURASIP Journal on Information Security, 2016(1), p.9.
28. Iuga, C., Nurse, J.R. and Erola, A., 2016. Baiting the hook: factors impacting susceptibility to phishing attacks. Human-centric Computing and Information Sciences, 6(1), p.8.
29. Li, Y., Yang, Z., Chen, X., Yuan, H. and Liu, W., 2019. A stacking model using URL and HTML features for phishing webpage detection. Future Generation Computer Systems, 94, pp.27-39.