JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-19, No.-8, August (2024) pp 94-119 ISSN (Print) 0973-8975

SECURING CYBERSPACE: AN EFFICIENT MACHINE LEARNING BASED APPROACH TO PHISHING ATTACK DETECTION

Attiq Ur Rehman[1], Hamayun Khan[2], Arshad Ali[3], Yazed ALsaawy[4], Irfan Ud din[5], Saif ur Rehman[6], Rao Muhammad Asif[7], Mohammad Husain[8]

[1,2,5]Department of Computer Science, Faculty of Computer Science & IT Superior University Lahore, 54000, Pakistan.

[3,4,8]Faculty of Computer and Information Systems, Islamic University of Madinah, Al Madinah Al Munawarah, 42351, Saudi Arabia.

[6,7]Department of Electrical Engineering, Faculty of Electrical Engineering and Technology, Superior University Lahore, 54000, Pakistan.

Email: [1]ateeq9710@gmail.com, [2]hamayun.khan@superior.edu.pk, [3]a.ali@iu.edu.sa, [4]yalsaawy@iu.edu.sa, [5]irfan@superior.edu.pk, [6]saifurrehman@superior.edu.pk, [7]rao.m.asif@superior.edu.pk, [8]dr.husain@iu.edu.sa

Corresponding Author: **Attiq Ur Rehman**

https://doi.org/10.26782/jmcms.2024.08.00008

(Received: June 08, 2024; Revised: July 28, 2024; Accepted: August 11, 2024)

Abstract

We explore machine learning strategies and evaluate their viability in distinguishing characteristics that separate secure websites from phishing ones. Given the essential need to defend delicate information and maintain network integrity, we aim to determine the most proficient strategy for identifying phishing websites. Our research focuses on the Random Forest Classifier, illustrating its predominance over other strategies. We have achieved significant improvements in detection rates, with the Random Forest Classifier accomplishing an F1 score of 0.99, precision of 0.99, recall of 0.99, and an AUC of 1.00, outperforming other classifiers. By specifying each strategy and utilizing various assessment methods for visual performance representation, we provide a robust model for phishing detection.

Keywords: Phishing assault recognition, AI, Random Forest, phishing site location.

I. Introduction

Phishing attacks are a common and unpretentious strategy utilized by hackers to invade frameworks in today's computerized environment. These attacks utilize various communication channels to deceive individuals into revealing confidential information in counterfeit settings. Contingent upon the assailant's goals and the sort

of information exposed, the consequences of falling victim to such scams can range from personal harm to significant organizational impact.

To avoid phishing attacks, raise mindfulness and advance careful behaviors. Be cautious while browsing, confirm interface genuineness, and utilize browser expansions to alert users to fake websites. These devices avoid unauthorized get to to critical data.

Furthermore, implementing measures that restrict access to only whitelisted websites can enhance network security. While these steps improve security, they might compromise user transparency. By combining knowledge, cautious online behavior, and the strategic deployment of security protocols, individuals and organizations can strengthen their defenses against the pervasive threat of phishing attacks [I].

Research has identified five primary factors contributing to individuals falling victim to phishing techniques:

I. Insufficient experience with URLs.

II. Lack of knowledge regarding reliable web sources.

III. Inability to differentiate between legitimate and fraudulent websites.

IV. Limited ability to view entire URLs due to redirects or hidden URLs.

V. Insufficient time to thoroughly examine URLs and inadvertent entry into certain websites.

A notable example of such a cyberattack is the Bangladeshi Bank Cyberspace. Hackers, identified as privacy intruders, sent 35 fraudulent directives across the system known as SWIFT, attempting to unlawfully move nearly a billion dollars out of the Bangladesh Bank's Federal Reserve Bank of New York account. Of these directives, five effectively moved $101 million, with $20 million followed to Sri Lanka and $81 million to the Philippines. Luckily, the Central Bank of New York interceded, obstructing the leftover 30 exchanges and forestalling a likely deficiency of $850 million. Identification was made conceivable by seeing an incorrectly spelled guidance, which raised doubts among specialists. About $18 million of the $81 million transferred to the Philippines was recovered, with the majority of the money diverted into four personal accounts. In contrast, the funds sent to Sri Lanka were fully recovered [II].

The suspected method behind this attack is believed to involve the Dridex virus, which uses macros hidden in Word or Excel documents to steal bank credentials. Windows users who opened email attachments containing these macros were susceptible to these assaults. Activating the macros upon opening these documents initiated the download of Dridex, subsequently infecting computers and setting the stage for a bank heist. In such scenarios, a vigilant and knowledgeable staff member or a program designed to identify similar attacks could have a pivotal impact [III].

In the field of machine learning, algorithms are extensively employed to identify concealed patterns in datasets. Commonly used algorithms include support vector

machines, arbitrary woods, choice trees, and k-closest neighbors. Additionally, belief rule-based expert systems can extract rules from datasets [IV].

This study focuses on making machine learning models to distinguish phishing web pages from genuine ones. It analyzes each show, highlighting information pre-pressing's basic part in ideal usefulness, and recognizes other researchers' commitments to collective information enhancement.

The subsequent sections of the paper are structured as follows: Section II provides an overview of the literature, Section III introduces the proposed methodology, Section IV details the empirical results derived from the proposed approach, and Section V encompasses the conclusion and exploration of potential avenues for further research.

II. Literature Review

Machine learning has become a cornerstone in detecting phishing attacks due to its ability to analyze vast amounts of data and identify patterns that may not be immediately apparent to human analysts. Various approaches have been proposed, each leveraging different algorithms and techniques to enhance detection accuracy. For instance, the use of Random Forest classifiers has shown significant promise, with studies indicating high F1 scores and improved detection rates. Other methods, such as support vector machines, logistic regression, and neural networks, also contribute valuable insights into the strengths and weaknesses of machine learning models in cybersecurity. However, the effectiveness of these techniques can vary based on the dataset, feature selection, and specific implementation details. This review synthesizes recent advancements in machine learning-based phishing detection, comparing their methodologies, performance metrics, and practical applicability in real-world scenarios.

Table 1: Comparison of Algorithms

Algorithm	Approach	Advantages	Disadvantages	Ref
Random Forest	Ensemble method using multiple decision trees.	High accuracy, handles large datasets well, resistant to overfitting.	Can be computationally intensive, less interpretable.	[V]
Support Vector Machine (SVM)	Finds optimal hyperplane for classification.	Effective in high-dimensional spaces, robust with a clear margin of separation.	Memory-intensive, less effective with noisy data.	[VI]
K-Nearest Neighbors	Classifies based on the majority class among the	Simple and intuitive, effective with small	Computationally expensive, sensitive to	[VII]

(KNN)	k-nearest neighbors.	datasets.	irrelevant features.	
Logistic Regression	Statistical model for binary classification.	Interpretable coefficients, works well with linearly separable data.	Limited to linear decision boundaries, less effective with non-linear data.	[VIII]
Decision Tree	Model based on recursive binary splitting.	Easy to understand and interpret, handles both numerical and categorical data.	Prone to overfitting,unstable with small changes in data.	[IX]
Extra Trees	Similar to Random Forest but uses random splits.	Reduces variance, computationally efficient.	Less interpretable, can be sensitive to noisy data.	[X]
Stochastic Gradient Descent (SGD)	An iterative method for optimizing differentiable functions.	Efficient with large datasets, versatile with different loss functions.	Requires tuning of hyperparameters, sensitive to feature scaling.	[XI]

This table presents a comparison of different machine learning calculations commonly utilized for phishing assault locations. Each calculation is portrayed briefly highlighting its approach, points of interest, and drawbacks. For occasion, Random Forest is famous for its tall accuracy and resistance to overfitting, making it appropriate for expansive datasets, even though it can be computationally serious and less interpretable. SVM is viable in high-dimensional spaces but is memory-intensive. KNN is basic and successful for little datasets but can be computationally costly. Logistic Regression offers interpretability but is restricted to linear decision boundaries. Decision Trees are simple to get but inclined to overfitting. Extra Trees decrease change and are productive, whereas SGD is flexible but requires cautious tuning. This comparison makes a difference in understanding the qualities and shortcomings of each calculation, supporting in selecting the suitable strategy for particular utilized cases in phishing discovery.

II.i. Types of Phishing Attacks

Attackers employ several algorithms to extract sensitive data from a website's database, various types are discussed below.

II.i.a. Phishing based on algorithms:

Attackers employ several algorithms to extract sensitive data from a website's database. A method for anti-phishing detection utilizes a rule-based system grounded on a genetic algorithm (GA) to identify phishing URLs. A URL is considered phishing if it matches the ruleset of the GA and is subsequently stored in a database [VII].

II.i.ib. Forged phishing

This procedure involves sending malicious links in emails, which direct individuals to harmful websites where there is a high risk of disclosing delicate data. They provide a comprehensive analysis of phishing attacks disguised as legitimate ones, along with various countermeasures. Researchers in [VIII] evaluate the pros and cons of different interventions and the strategies used by phishers.

II.i.c. URL phishing

Cybercriminals have the capability to insert concealed links that lead to malicious pages directly into a URL, catching users off guard. Mohammed Nazim Feroz and Susan Mengel suggest an approach for detecting URL phishing through URL ranking. Their method involves classifying URLs based on their lexical and host-related attributes, subsequently categorizing and assigning rankings to these URLs with the assistance of Numerous URL reputation services available online [IX].

II.i.d. Hosts report poisoning

Substituting names of hosts within-host data has the potential to disrupt the usual DNS server process, where the servers attempt to fetch authentic IP addresses from external networks. This method involves contaminating the records, allowing legitimate Sites which are supposed to direct users to safe websites to direct users to dangerous ones instead. This occurs because of hacking on the server, and IP associations. Researchers present an interesting DNS-harming method that can move beyond security toolbars and phishing channels. By employing fake DNS cache records, they generate deceptive outcomes, effectively evading detection while targeting four well-known protection plugins and three different phishing filters in widely used browsers [X].

II.i.e. Content injection phishing

In this method, data collection is accomplished by integrating malicious segments within an authentic website. Researcher in [XI] outlines various ways in which phishing techniques can deceive individuals, providing a compilation of strategies for detecting phishing. The paper recommends that organizations implement robust protocols to ensure the continuous updating of their security features.

II.i.f. Clone phishing

Successfully targeting unsuspecting users can be achieved by replicating previously sent emails and incorporating a malicious link. Ahmad Alamgir Khan introduces an innovative approach wherein websites employ combat phishing attempts, a user-machine identification system, and a one-time password. In this method [XII], web

servers dispatch a user's one-time password by email or SMS. Upon the user entering the password, the system generates an encrypted token for the device.

II.ii. Phishing Website Detection Techniques:

To prevent unwanted sites recorded on blacklists from reaching a client's machine, various security measures such as DNS servers, firewalls, and email servers can employ these filters various techniques are discussed below.

II.ii.a. Blacklist filter

Blacklist filters maintain a catalogue comprising components such as domains and IP addresses, and IP netblocks are often used by cybercriminals. A scalable methodology is utilized to gauge the adequacy of program boycott channels. Their findings indicate that for a majority of mobile browsers, blacklist filters are ineffective against phishing attacks, rendering them more susceptible [XIII]. Researchers propose an original strategy including a boycott generator that actively monitors phishing website blacklists. In [XIV], Their methods show that they can identify phishing websites with 100% accuracy and genuine sites with 91% accuracy.

II.ii.b. Whitelist filter

In contrast to blacklists, whitelist channels empower recorded site URLs, plans, or spaces to arrive at the client machine, while at the same time obstructing any remaining unrecorded locales. Unlike blacklists, whitelists consist of a comprehensive list of all legitimate websites. The use of a support vector classifier in conjunction with machine learning to further filter URLs that are not prohibited by the whitelist [XV]. Researchers tested and compared the efficacy of anti-phishing toolbars with blacklists and whitelists. Their research revealed no significant performance difference between the two toolbars but underscored the importance of toolbars offering instructive guidance to assist users in identifying phishing websites [XVI].

II.ii.c. Pattern matching filter

Examining if individual tokens or data sequences are present within a provided dataset is done through a method for matching patterns. In [XVII] they suggest a methodology employing pattern correspondence for the identification of phishing websites. This method involves utilizing a database that consists of both blacklisted and whitelisted patterns, encompassing malicious and original URL patterns. The user-requested URL is then matched against this database.

II.iii. Methods Based on Machine Learning:

Various methods based on Machine Learning techniques are discussed below.

II.iii.a. Malicious website identification

Models for machine learning are undergoing training to enhance their ability to detect phishing websites, a prevalent form of cyberattack. A strong feature selection process is presented by researchers aimed at improving the performance of malicious domain detection models. The dataset comprises information from 1350 malicious URLs and 5000 trustworthy URLs. The resulting prototypes demonstrate resilience to various

malevolent anomalies, emphasizing the efficacy of models developed using particular characteristics [XVIII].

Expressing concerns about the multitude of training features and dataset types, as well as emphasizing the efficacy of domain names. Their learning model achieves a high accuracy of 99.7% in detecting unknown live phishing URLs, advocating for the superiority of domain names in phishing website detection [XIX]. An analysis that looks at the similarities between various fraudulent domain name constructions. The fundamental goal of malicious activity detection is to find commonalities between collections of hostnames, URL names, and domain names [XX].

II.iii.b. Spam screening for emails

Emails undergo screening using a variety of scoring techniques that leverage thousands of rules to predict the likelihood of being a legitimate or spam email. If the calculated probability exceeds an acceptable threshold, the electronic mail is intercepted and restricted by the filter for spam. Scammers often exploit unsolicited emails as a means to redirect recipients to malicious web pages for data theft. Investigates how well a phishing email classifier can be built using a random forest classifier. Their research involves removing relevant characteristics from a collection containing 2000 emails with phishing and ham. 99.7% classification accuracy is shown by the suggested machine learning models, with very few false positives or negatives [XXI].

In a similar vein, researchers focus on the appropriate extraction characteristics from the content of spam emails and features based on behavior crucial for detecting spam emails. The machine learning approach they recommended depends on certain attributes, and consistently achieves 99% accuracy in detecting unsolicited emails [XXII].

Table One provides insights into the benefits and drawbacks of the current phishing detection system research. Notably, many of the referenced studies focus on a limited quantity of datasets and characteristics. In this study, efforts are made could get around these restrictions by adding more features and expanding the volume of the dataset.

Table 2: A comparison of machine learning-based phishing detection techniques

Overview	Advantages	Disadvantages	Ref
Identifies phishing attempts through the utilization of a whitelist filter.	If a page makes it past the whitelist filter, Support Vector Machines perform extra filtering.	The dataset is restricted to 850 pages. Exhibits a high rate of false positives.	[XXIII]
Create a browser plugin with a comment spam detection to	Applies WEKA filters to collect the most appropriate characteristics in	Performs inadequately until a trained resample filter is used, and a random	[XXIV]

eliminate spam comments.	order to achieve dataset balance.	dataset is applied.	
Suggests a machine learning approach for identifying potential phishing attacks on web pages.	The suggested approach relies on a readily obtainable feature vector, eliminating the need for additional computations.	Utilizes only 10 features for detection. Constrained by a dataset comprising only 1353 instances.	[XXV]
Proposes a method based on machine learning to identify if a web page displays signs of phishing attacks.	Improved accuracy is achieved through the implementation of feature selection. Computational time is reduced by utilizing feature selection.	Might face challenges when dealing with datasets having an equal number of authentic and fraudulent websites.	[XXVI]
Develops a machine learning system capable of categorizing websites based on their URLs.	Enables the construction of a rule-based system with associative rules for URL classification.	Operates with a restricted dataset of 1353 URLs.	[XXVII]
Introduces a learning-based aggregation analysis mechanism to determine the similarity in page layout, utilized for detecting phishing pages.	Naturally prepares classifiers to survey page closeness in view of CSS format features, eliminating the need for human expertise.	The method is lightweight, focusing on a single class of features, specifically CSS structure.	[XXIII], [XXVIII]
This study utilizes a clever quality named the "space top page closeness" viability.	Enhances the f-measure in addition to reducing the rate of errors.	The effectiveness of the model relies heavily on the precision of the characteristics	[XXIX]
The article in question suggests a continuous enemy of phishing framework with seven normal language handling	Operates independently of language and outside services. Utilizes an extensive collection containing data that is both authentic and	Systems based on machine learning struggle to effectively utilize such an extensive dataset. However, it is constrained by the	[XXX] ,

(NLP) based categorization methods and characteristics.	fraudulent.	dataset's size and the dispersion of samples	[XXXI]
Conducts a thorough analysis of squatting phishing, a scenario in which Phishing websites imitate target brands both in terms of content and domain names.	An openly available tool. Leverages phishing sites' evasive tactics for building classifiers.	Unable to identify phishing pages employing cloaking techniques. Concentrates solely on widely recognized brands.	[XXXII], [XXXIII]

III. Methodology

In this specific area, we outline our proposed framework for distinguishing phishing sites through information-driven techniques. The dataset is obtained from the Mendeley online storehouse. We utilize strategies, for example, Head part examination, Pearson and Shapiro positioning, and equal directions are utilized to remove highlights. The ID of phishing sites is completed utilizing KNN, choice trees, Random Forest, SVM, and logistic regression.

III.i. Proposed Solution

The proposed solution involves a machine learning framework for detecting phishing websites using a dataset of website features sourced from the Mendeley online repository. The dataset includes 48 features across 1000 websites, evenly split between phishing and genuine sites.

III.ii. Data Utilization and Feature Extraction

We employed Principal Component Analysis (PCA), Pearson and Shapiro ranking, and parallel coordinates for feature extraction. The extracted features include lexical features, host-based features, and correlation features. Principal component analysis (PCA), Pearson and Shapiro ranking, and parallel coordinates were used for feature extraction. Parallel coordinates were utilized for visualizing and analyzing the dataset, while PCA was applied to reduce its dimensionality. The features are detailed in Table I

III.iii. Proposed Model

Our proposed model focuses on the Random Forest algorithm for detecting phishing websites. This model is selected due to its high accuracy, robustness, and ability to handle large datasets efficiently. The following sections outline the key components of our proposed solution, including data utilization, feature extraction, and the Random Forest classifier.

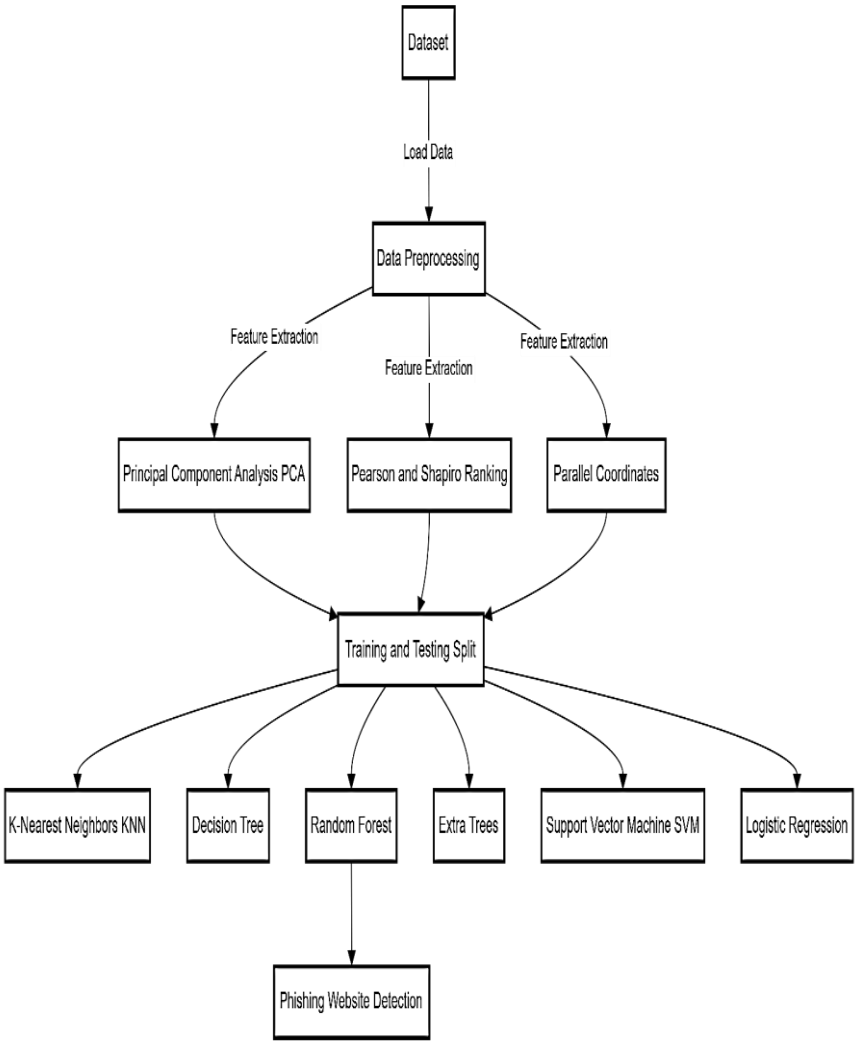

Fig.1. *Proposed Model*

Figure 1 delineates the technique for phishing site location utilizing machine learning. The method starts with stacking the dataset, taken after data preprocessing. Include extraction is conducted using three strategies:

III.iv. Datasets

The dataset on phishing webpages comprises 48 features sourced from the Mendeley online repository. It encompasses a total of 1000 websites, evenly split between 5,000 of them are phishing, and 5,000 are real. Class label 0 designates a fraudulent website, whereas label 1 designates a trustworthy website.

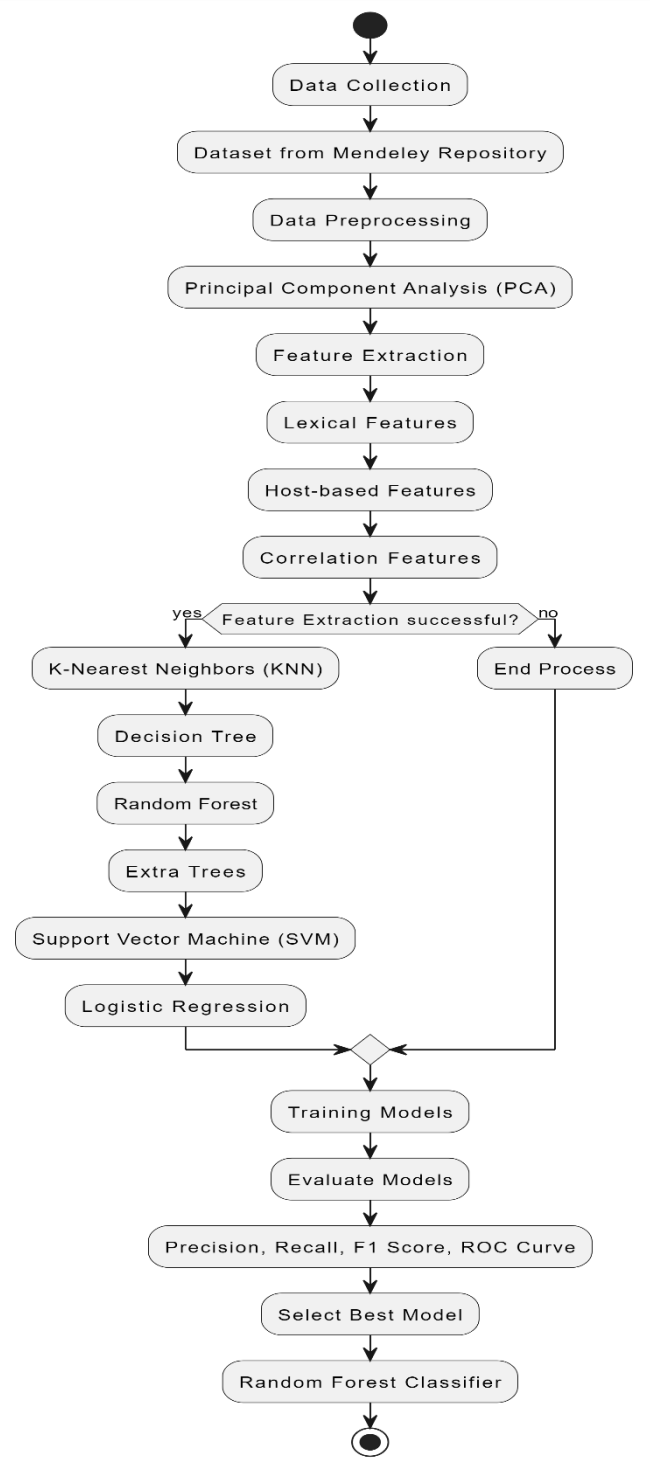

Fig. 2. *Flow Chart*

III. v. Classifier

In our system, we utilize various machine learning algorithms, including K-Nearest Neighbors (KNN), decision trees, random forest, extra trees, Support Vector Machine (SVM), and logistic regression.

III. v.a. KNN (K-Nearest Neighbors)

Equation (1) defines the Euclidean approach that we used to determine the distance.

$$d(x, x') = \sqrt{(x_1 - x_1')^2 + \cdots + (x_n - x_n')^2} \quad (1)$$

Our KNN model based on equation (2)

$$P(y = j \mid X = x) = \frac{1}{K}\sum_{i \in A} I\left(y^{(i)} = j\right) \quad (2)$$

48 characteristics make up our dataset. A Class label with a value of 0 designates a phishing website and a value of 1 designates a reputable website. When presented with an unknown sample, KNN will initially use the Euclidean distance to calculate how far away the unknown sample is from its neighbors. The worth of K that might be picked by indicating the worth of "n_neighbors" will be the number of neighbors that it will look at. By considering the characteristics of the samples included in the dataset, the distances will be calculated. Next, the nearest neighbors' majority class will be allocated to the unidentified sample.

Table 3: Lexical feature group's rundown of url highlights

Feature	Data Type	Description
NumDots	Numeric	are contained in the web address?
SubdomainLevel	Numeric	Establishes the quantity levels of subcategories.
PathLevel	Numeric	Identifying the path's level in the URL.
UrlLength0	Numeric.	Each URL's length is utilized in the collection. The length indicates how many characters or symbols were used to form the URL
NumDash	Numeric	Sum of the dashes in a URL.
NumDashInHostname	Numeric	How many dashes are in a hostname?
AtSymbol	Boolean	Total '@' symbols found in the URL.
TildeSymbol	Boolean	The URL's absolute tilde ('~') images.
NumUnderscore	Numeric	The amount of "_" highlights utilized in the URL.
NumPercent	Numeric	The all-out rate image is tracked down in the URL.
NumQueryComponents	Numeric	The whole amount of parts in the query.
NumAmpersand	Numeric	Sum of all '&' characters.

III.v.b. Random Forest

We employed Gini importance to assess the significance of each node in a decision tree, assuming that there is binary data in the tree, implying every node possesses a maximum of a pair of kids. To trim the tree's branches, We made use of equation (3).

$$ni_j = w_j C_j - w_{\text{left }(j)} C_{\text{left }(j)} - w_{\text{right }(j)} C_{\text{right }(j)} \tag{3}$$

To determine the significance we used formula (4) on each feature in a decision tree.

$$fi_i = \frac{\sum_{j:node j \text{ splits on feature } i} ni_j}{\sum_{k \epsilon \text{ all nodes}} ni_k} \tag{4}$$

Subsequently, these values can be normalized to fall within the range of 0 to 1 using the formula (5).

$$\text{normfi } i_i = \frac{fi_i}{\sum_{j \epsilon \text{ all features}} fi_j} \tag{5}$$

The cumulative importance value of each feature across all trees is computed using formula (6) and then divided by the total quantity of trees.

$$RFfi_i = \frac{\sum_{j \epsilon \text{ all trees}} \text{norm } fi_{ij}}{T} \tag{6}$$

A random forest classifier is comprised of numerous decision trees that operate collectively. Initially, it generates a bootstrap dataset of size "N" by randomly sampling data points with replacements from the original dataset. These bootstrap samples are then used to construct individual trees within the random forest. Unlike a single decision tree, a random forest introduces the concept of Feature Randomness during tree construction. This means that when selecting a tree's root node in the random forest, just a portion of its attributes is considered. The impurity Gini is calculated within these feature groups, as well as the feature subset with the lowest impurity score is chosen as the root node. The process is iterated for subsequent nodes. After constructing the trees, the random forest is prepared for making predictions. When presented with an unknown sample from the test dataset, it is evaluated across all trees. Each tree provides a class prediction, and the final prediction is determined by the majority vote among all the trees. The success of the random forest classifier with large datasets is attributed to its ability to maintain model diversity through bootstrap aggregation, these feature subsets, and feature randomness.

III.v.c. Support Vector Machine

Formula (7) was employed to compute the misfortune capability for our help vector machine (SVM).

$$\min_w \lambda \parallel w \parallel^2 + \sum_{i=1}^{n} (1 - y_i \langle x_i, w \rangle)_+ \tag{7}$$

To compute gradients, we applied formula (8).

$$\frac{\partial}{\partial w_k}\lambda \parallel w \parallel^2 = 2\lambda w_k$$
$$\frac{\partial}{\partial w_k}(1 - y_i\langle x_i, w\rangle)_+ = \begin{cases} 0, & \text{if } y_i\langle x_i, w\rangle \geq 1 \\ -y_i x_{ik}, & \text{else} \end{cases} \quad (8)$$

Through the utilization of SVM, we address every data of interest as an instance in an n-layered space, where 'n' connects with the number of components (48 within our collection). Each element's worth fills in as a direction for the separate point. SVM then, at that point, looks to distinguish a hyperplane or choice limit in this space that effectively separates the different instructions. When two data points—known as support vectors—maintain an equal and maximum distance apart, that hyperplane is considered ideal.

While SVM is straightforward for linearly separable datasets, such cases are uncommon in real-world scenarios. The kernel trick in SVM becomes instrumental in handling more complex datasets. What sets SVM apart is its ability to operate in infinite dimensions. The kernel, without explicitly generating infinite dimensions, effectively simulates a higher-dimensional space. This proves advantageous as it transforms non-separable problems into separable ones by introducing additional dimensions. The number of dimensions added is contingent upon the features in each sample.

III.v.d. Logistic Regression

Plotting a line on axes to represent a data set is the basis of linear regression, which is expanded upon by logistic regression given the dataset. In logistic regression, a conditional probability function is employed to yield a binary output for the variable Y based on the variable X. To estimate any unknown parameters within this function, the maximum likelihood method is utilized. The calculation of the conditional probability is determined by employing equation (9).

$$\Pr(Y = 1 \mid X = x) = \log\frac{p(x)}{1-p(x)} = \beta_0 + x \cdot \beta \quad (9)$$

Equation (10) was also utilized for the sigmoid function,

$$S(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1} \quad (10)$$

Equation (11) is the cost function,

$$-\frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)}\log h_0\left(x^{(i)}\right) + \left(1 - y^{(i)}\right)\log\left(1 - h_0\left(x^{(i)}\right)\right)\right] \quad (11)$$

We compute the angle by utilizing the conditions (12), (13), (14), and (15).

$$J = -\frac{1}{m}[\sum_{i=1}^{m} y_i\log h_i + (1 - y_i)\log 1 - h_i] \quad (12)$$
$$\frac{\partial J}{\partial \theta_n} = -\frac{1}{m}\cdot\left[\sum_{i=1}^{m}\frac{y_i}{h_i}\cdot h_i^2\cdot x_n\cdot\frac{1-h_i}{h_i} + \frac{1-y_i}{1-h_i}\cdot -h_i^2\cdot x_n\cdot\frac{1-h_i}{h_i}\right] \quad (13)$$
$$\frac{\partial J}{\partial \theta_n} = -\frac{1}{m}\cdot[\sum_{i=1}^{m} x_n\cdot(1-h_i)\cdot y_i + x_n\cdot h_i\cdot(1-y_i)] \quad (14)$$
$$\frac{\partial j}{\partial \theta_n} = \frac{1}{m}\cdot x_i\cdot[\sum_{i=1}^{m} h_i - y_i] \quad (15)$$

IV. Result Analysis

The below section elaborates various results Outstandingly, this setup accomplishes the most noteworthy AUC worth of 0.98, showing unrivaled execution. The steepness of the bend is outstandingly nearer to the upper left place on the diagram.

. IV.i. ROC curve

In Figure 1 shows the backing vector machine's ROC curve displayed, when the genuine positive rate is represented by the Y-axis and the false positive rate is shown by the X-pivot. The comparing AUC is 0.97. Continuing Fig. 2, outlines The non-uniform help vector machine's ROC bend, creating an AUC worth of 0.96 with the X-hub addressing the misleading favorable percentage and the Y-pivot addressing the genuine positive rate. The straight help vector machine's ROC bend is displayed in Figure 3, with the genuine positive rate addressed by the Y-pivot and the bogus positive rate by the X-hub.

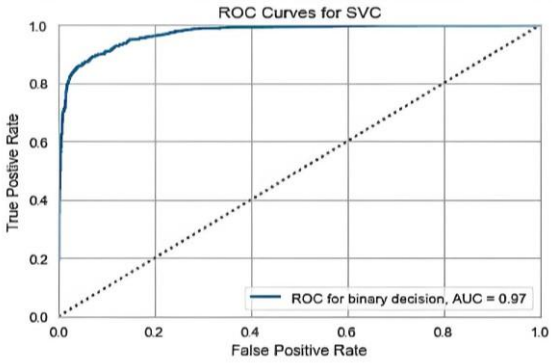

Fig. 3. SVC curves

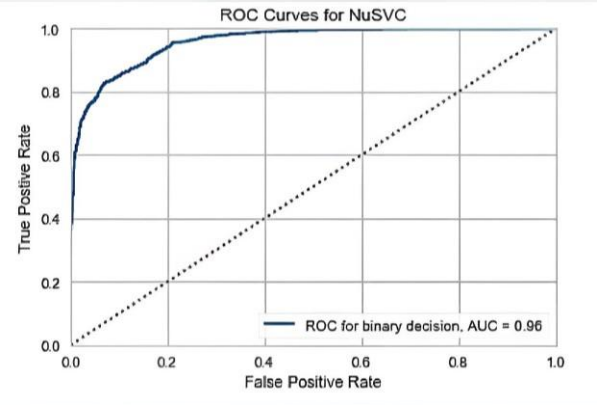

Fig. 4. NuSVC ROC Curves.

The ROC bend for KNN is presented in Figure 4, where the Y-hub represents the true positive rate and the X-pivot represents the deceptive positive rate. Specifically, the class 0 AUC (phishing site) is 0.94, while for class 1 (genuine site), it is also 0.94.

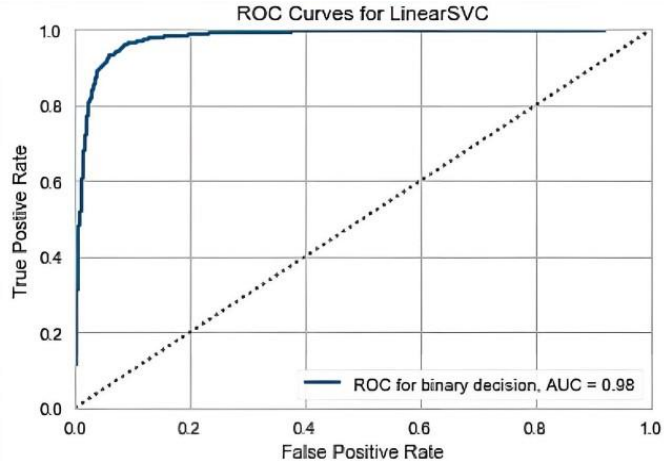

Fig. 5. ROC Diagrams in Linear SVC**.**

Additionally, The ROC curve's macro and micro average AUC values are found to be 0.94. The ROC curve for Logistic Regression is shown in Fig. 5, where the true positive rate is shown by the Y-axis and the false positive rate is represented by the X-axis. Particularly, the AUC is 0.96 for both class 0 (phishing websites) and class 1 (genuine websites). Additionally, the ROC curve's macro and micro average AUC values are both consistently reported at 0.96.

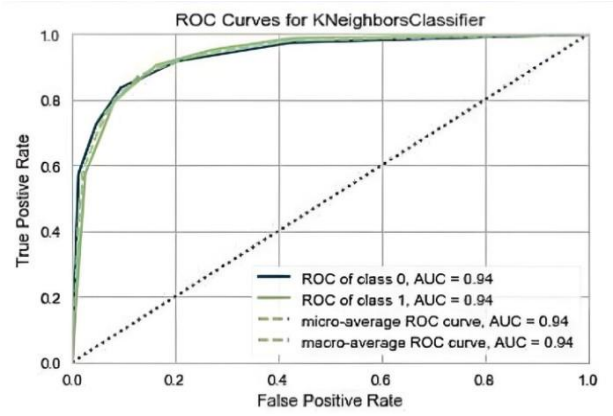

Fig. 6. KNeighborsClassifier ROC Curves**.**

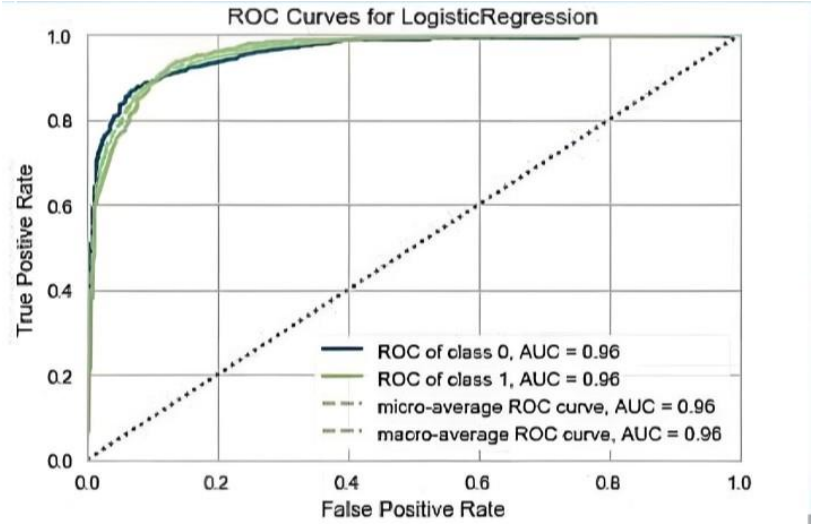

Fig. 7. ROC Curves for Regression Analysis**.**

In Fig. 6, the ROC twist for stochastic slant plunge (SGD) is presented, where the X-center tends to the deceptive positive rate, and the Y-center point infers the veritable positive rate. Very, the AUC an impetus for this game plan is recorded at 0.97. Figures 7 and 8 display the logistic regressionCV ROC curve is depicted, with the false positive rate indicated by the X-axis, and the Y-axis indicating the actual rate of positive. Notably, the AUC for the phishing website in class 0 achieves a value of 0.98, and for class 1 (real website), it similarly stands at 0.98.

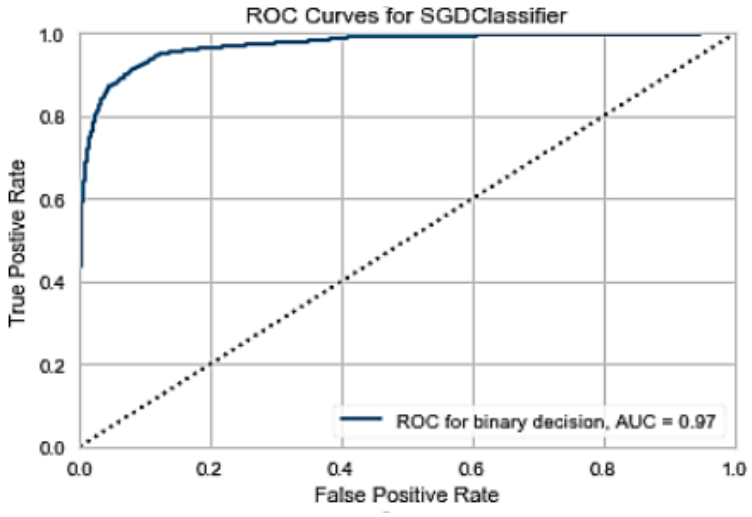

Fig. 8. ROC Curves for SGD Detector.

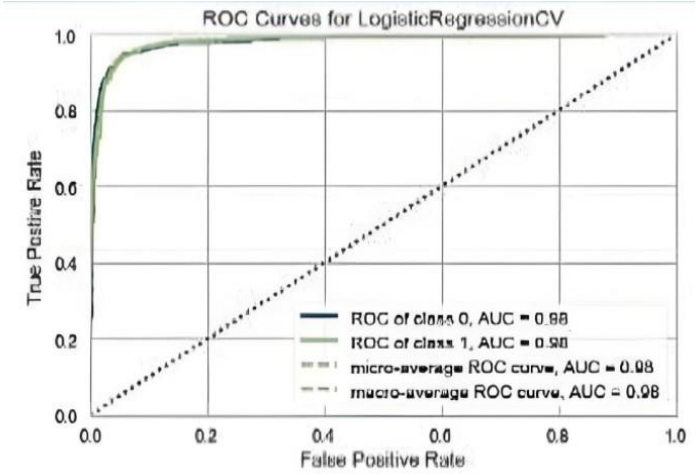

Fig. 9. Logistic Regression CV ROC Curves.

The ROC bend for the extra trees classifier is displayed in Fig. 9, where the genuine positive rate is displayed on the Y-pivot and the bogus positive rate is demonstrated on the X-hub. Essential is the remarkable AUC execution, coming to 1.00 for the two sites in classes 0 (phishing) and 1 (true sites). Moreover, both The ROC bend's large scale and miniature normal AUC values both come to a perfect 1.00. This is the best ROC bend that can be noticed hitherto, clear in The angle of the curve at the upper-left corner is quite severe.

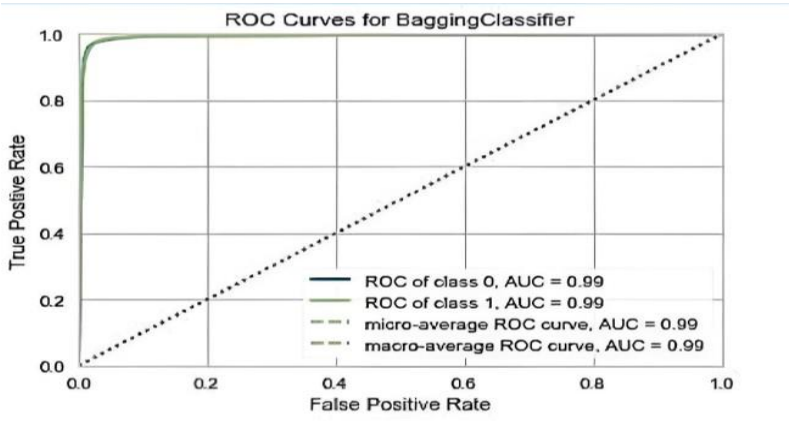

Fig. 10. ROC Curves applied Classification Bagging.

The arbitrary backwoods classifier's ROC bend is displayed in Fig. 10, with the genuine positive rate addressed by the Y-hub and the misleading positive rate by the X-hub. Eminently, the AUC for both class 0 (phishing site) and class 1 (genuine site) accomplishes an ideal 1.00. Essentially, both the large-scale and miniature normal AUC values for the ROC bend likewise accomplish an impeccable 1.00. Strikingly, this mirrors the exhibition saw in the Additional Trees classifier. The bend's steepness

situated at the upper-furthest left corner shows that both case, the Irregular Backwoods and Additional Trees classifiers had the best ROC bends.

IV.ii. Discrimination Threshold

IV.ii.a. Examining our models' discriminatory threshold:

The edge plot for the help vector machine is shown in Fig. 11. The score is shown by the Y-axis, while the discriminating threshold is represented by the X-axis. Notably, the discrimination threshold for this model is observed to be 0.03. At this threshold, recall, accuracy, and F1 score are roughly recorded at 0.89.

Exploring our models' discriminatory threshold, Fig. 12 displays the non-uniform support vector machine's threshold plot. The score is indicated by the Y-axis, while the discriminating threshold is represented by the X-axis. Notably, the discrimination threshold for this model is observed to be 0.00. At this threshold, the approximate values for accuracy, recall, and F1 score are 0.86.

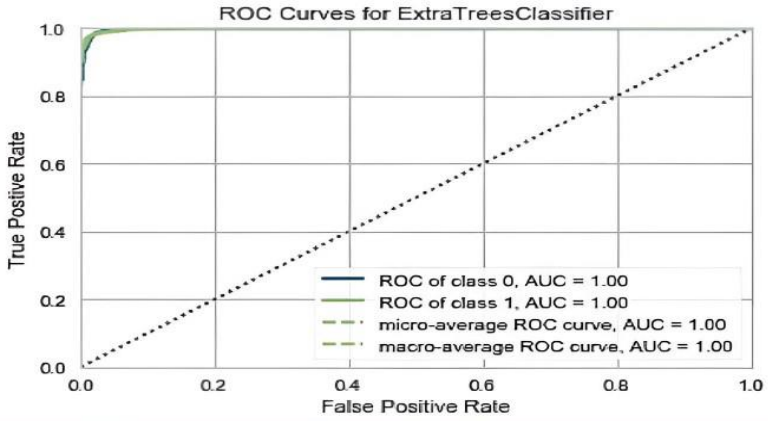

Fig. 11. ROC Curves for the Classifier ExtraTrees.

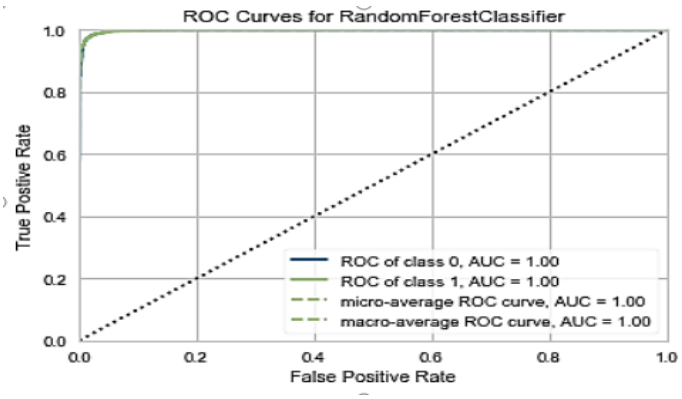

Fig. 12. Random Forest Classifier's ROC Curves.

Examining the discrimination the linear support vector machine's threshold, Fig. 13 presents the threshold plot. The X-axis corresponds to the discriminating threshold, with the score shown on the Y-axis. Notably, the discrimination threshold for this model is found to be 0.05. At this threshold, precision, recall, and F1 scores are approximately recorded at 0.9.

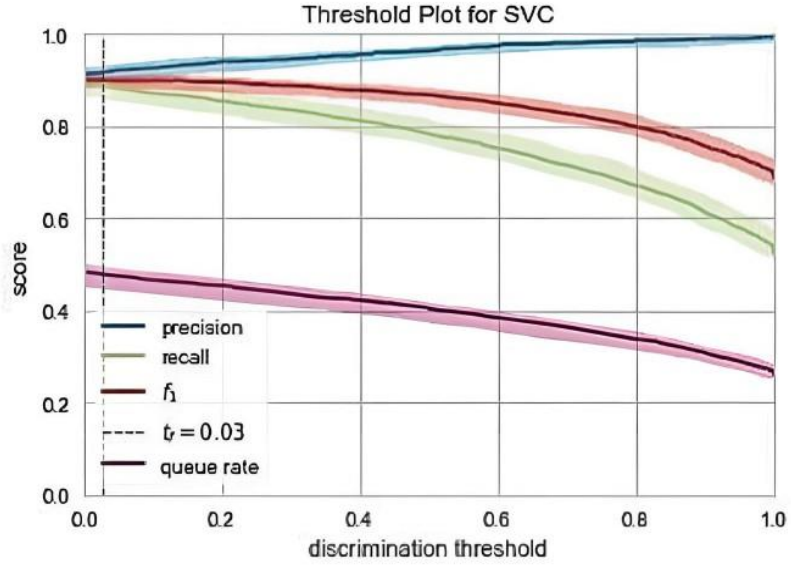

Fig. 13. SVC Threshold Plot**.**

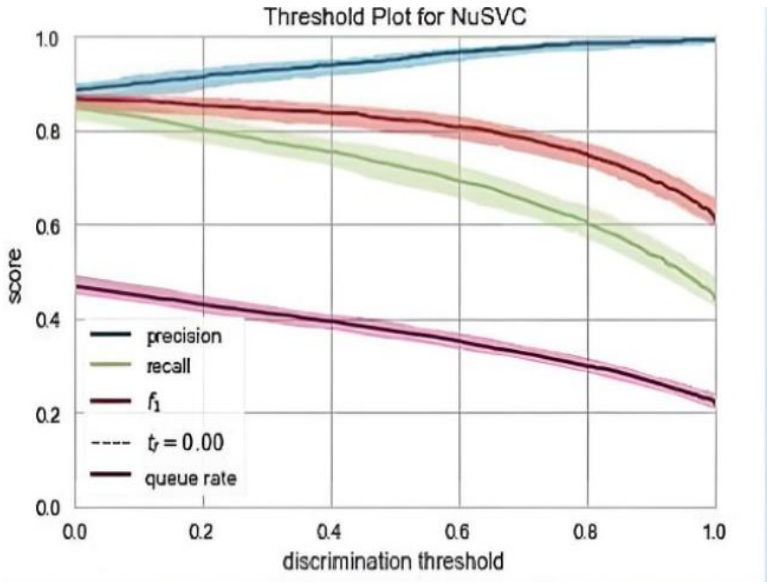

Fig. 14. NuSVC Threshold Plot**.**

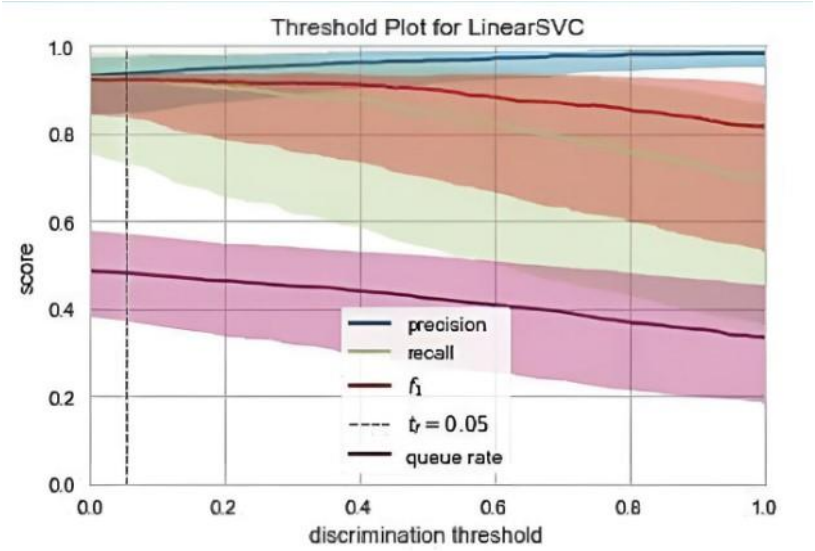

Fig. 15. Plotting Threshold for Linear SVC.

Exploring the discrimination threshold, Fig. 14 shows the KNN threshold plot. The score is indicated by the Y-axis, while the discriminating threshold is represented by the X-axis. In this case, the discrimination threshold is observed to be 0.50.

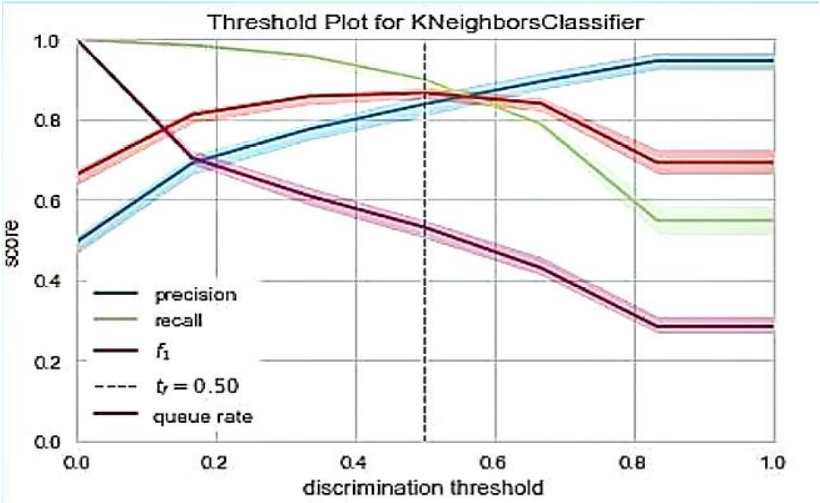

Fig. 16. Limit Diagram for K-Nearby Classifier.

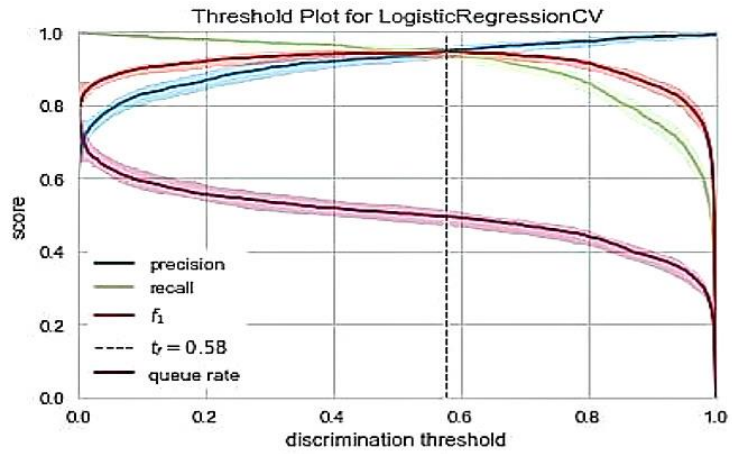

Fig. 17. CV of the Logistic Regression Threshold Plot**.**

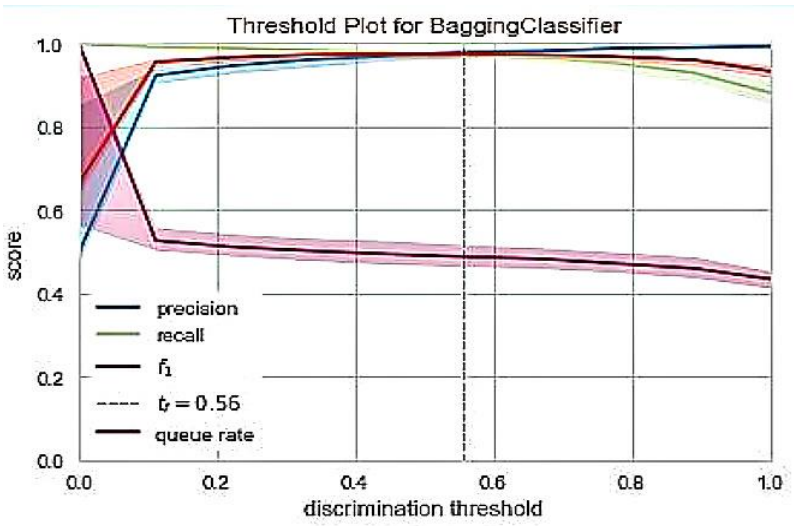

Fig. 18. Benchmark Diagram for Bagging Classification**.**

V. Conclusion & Future Work

Our research investigates several machine learning strategies to recognize phishing websites, centering on a dataset of site qualities. This paper serves as a specialized asset by advertising a thorough examination of different strategies. Eminently, our results highlight the Random Forest Classifier's extraordinary execution, accomplishing an F1 score of 0.99, which shows a critical advancement over other classifiers.

By specifying each technique and utilizing different evaluation methods for visual execution representation, our study gives a strong model for phishing detection. We utilized Principal Component Analysis (PCA), Pearson and Shapiro positioning, and

parallel facilitates highlight extraction, upgrading the dataset's viability. The dataset, sourced from the Mendeley online store, incorporates 48 highlights over 1000 websites, equitably part between phishing and veritable sites.

Our contributions are apparent within the fastidious design and usage of machine learning algorithms, counting K-Nearest Neighbors (KNN), decision trees, random forests, extra trees, Support Vector Machine (SVM), and logistic regression. Through rigorous assessment, the Random Forest Classifier has risen as the prevalent procedure, illustrating a perfect adjustment of accuracy, review, and F1 score.

This research aims to move forward with the dataset assistance and investigate progressed machine learning strategies, such as multilayer perceptrons and artificial neural systems, to handle the energetic cluster of cybersecurity issues within the advanced world.

Conflict of Intrest

There is no conflict of interest regarding this article.

References

I. A. Ogata, N. Aikawa; M. Sato, "A design method of low delay FIR bandpass filters", IEEE International Symposium On Circuits And Systems Emerging Technologies For The 21st Century, Vol. 1, pp. 92 - 95, 2000

II. A. Belabed, E. Aïmeur and A. Chikh, "A Personalized Whitelist Approach For Phishing Webpage Detection", 2012 Seventh International Conference On Availability, Reliability And Security, Prague, Pp. 249-254, 2012

III. A. Naz, H. Khan, I. U. Din, A. Ali, M. Husain, "An Efficient Optimization System for Early Breast Cancer Diagnosis based on Internet of Medical Things and Deep Learning", Engineering, Technology & Applied Science Research, Vol.14, No.4, pp. 15957-15962, 2024

IV. Hassan, H. Khan, I. Uddin, A. Sajid, "Optimal Emerging trends of Deep Learning Technique for Detection based on Convolutional Neural Network", Bulletin of Business and Economics (BBE), Vol.12, No.4, pp. 264-273, 2023

V. H. Khan, A. Ali, S. Alshmrany, "Energy-Efficient Scheduling Based on Task Migration Policy Using DPM for Homogeneous MPSoCs", Computers, Materials & Continua, Vol.74, No.1, pp. 965-981, 2023.

VI. H. Sarwar, H. Khan, I. Uddin, R. Waleed, S. Tariq, "An Efficient E-Commerce Web Platform Based on Deep Integration of MEAN Stack Technologies", Bulletin of Business and Economics (BBE), Vol. 12, No.4, pp. 447-453, 2023

VII. Hammad, E., Zhao, , "Mitigating link insecurities in smart grids via QoS multi-constraint routing", In 2016 IEEE International Conference on Communications Workshops (ICC)", pp. 380-386. 2016

VIII. H. Khan, I. Uddin, A. Ali, M. Husain, "An Optimal DPM Based Energy-Aware Task Scheduling for Performance Enhancement in Embedded MPSoC" Computers, Materials & Continua, Vol.74, No.1, pp. 2097-2113, 2023

IX. Hammad, A. A., Ahmed, "Deep Reinforcement Learning for Adaptive Cyber Defense in Network Security", In Proceedings of the Cognitive Models and Artificial Intelligence Conference, pp. 292-297, 2016

X. H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, "Offline Earliest Deadline first Scheduling based Technique for Optimization of Energy using STORM in Homogeneous Multi-core Systems," IJCSNS Int. J. Comput. Sci. Netw. Secur, Vol.18, No.12, pp 125-130, 2018

XI. Hossein Shirazi, Bruhadeshwar. B,"Kn0w Thy Doma1n Name": Unbiased Phishing Detection Using Domain Name Based Features. In Proceedings Of The 23nd Acm On Symposium On Access Control Models And Technologies (Sacmat '18). Association For Computing Machinery, New York, Ny, Usa, pp. 69-75, 2018

XII. H. Khan, S. Ahmad, N. Saleem, M. U. Hashmi, Q. Bashir, "Scheduling Based Dynamic Power Management Technique for offline Optimization of Energy in Multi Core Processors" Int. J. Sci. Eng. Res, Vol.9, No.12, pp 6-10, 2018

XIII. H. Khan, K. Janjua, A. Sikandar, M. W. Qazi, Z. Hameed, "An Efficient Scheduling based cloud computing technique using virtual Machine Resource Allocation for efficient resource utilization of Servers" In 2020 International Conference on Engineering and Emerging Technologies (ICEET), IEEE, pp 1-7, 2020

XIV. Hammad, M., Jillani, R. M., Ullah, S., Namoun, A., Tufail, A., Kim, K. H., & Shah, H, "Security framework for network-based manufacturing systems with personalized customization", An industry 4.0 approach, Sensors, vol. 23. No. 17-55, 2022

XV. H. Khan, Q. Bashir, M. U. Hashmi, "Scheduling based energy optimization technique in multiprocessor embedded systems" In 2018 International Conference on Engineering and Emerging Technologies (ICEET), IEEE, pp 1-8, 2018

XVI. H. Khan, A. Yasmeen, S. Jan, U. Hashmi, "Enhanced Resource Leveling Indynamic Power Management Techniqueof Improvement In Performance For Multi-Core Processors", Journal Of Mechanics Of Continua And Mathematical Sciences, Vol.6, No.14, pp. 956-972, 2019

XVII. H. Khan, K. Janjua, A. Sikandar, M. W. Qazi, Z. Hameed, "An Efficient Scheduling based cloud computing technique using virtual Machine Resource Allocation for efficient resource utilization of Servers" In 2020 International Conference on Engineering and Emerging Technologies (ICEET), IEEE, pp 1-7, 2020

XVIII. H. Huang, J. Tan And L. Liu, "Countermeasure Techniques For Deceptive Phishing Attack", International Conference On New Trends In Information And Service Science, Beijing, pp. 636-641, 2009

XIX. H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, "Offline Earliest Deadline first Scheduling based Technique for Optimization of Energy using STORM in Homogeneous Multi-core Systems" IJCSNS Int. J. Comput. Sci. Netw. Secur, Vol.18, No.12, pp 125-130, 2018

XX. H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, A. Saleem, "Performance Evaluation for Secure DES-Algorithm Based Authentication & Counter Measures for Internet Mobile Host Protocol" IJCSNS Int. J. Comput. Sci. Netw. Secur, Vol.18, No.12, pp 181-185, 2018

XXI. J. Chen; J. Tan, C. Chang, F. Feng, "A New Cost-Aware Sensitivity-Driven Algorithm for the Design of FIR Filters", IEEE Transactions on Circuits and Systems I, Vol. 64, No. 6 pp: 1588 - 1598, 2017

XXII. M. Y. A. Khan, F. Khan, H. Khan, S. Ahmed, M. Ahmad, "Design and Analysis of Maximum Power Point Tracking (MPPT) Controller for PV System" Journal of Mechanics of Continua and Mathematical Sciences, Vol.14, No.1, pp 276-288, 2019

XXIII. M. Y. A. Khan, "A GSM based Resource Allocation technique to control Autonomous Robotic Glove for Spinal Cord Implant paralysed Patients using Flex Sensors", Sukkur IBA Journal of Emerging Technologies, Vol.3, No.2, pp 13-23, 2020

XXIV. M. Y. A. Khan, "A high state of modular transistor on a 105 kW HVPS for X-rays tomography Applications", Sukkur IBA Journal of Emerging Technologies, Vol.2, No.2, pp 1-6, 2019

XXV. M. Shah, S. Ahmed, K. Saeed, M. Junaid, H. Khan, "Penetration testing active reconnaissance phase–optimized port scanning with nmap tool" In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, pp 1-6, 2019

XXVI. M. Y. A. Khan, M. Ibrahim, M. Ali, H. Khan, E. Mustafa, "Cost Benefit Based Analytical Study of Automatic Meter Reading (AMR) and Blind Meter Reading (BMR) used by PESCO (WAPDA)," In 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, pp 1-7, 2020

XXVII. M. Y. A. Khan, "Enhancing Energy Efficiency in Temperature Controlled Dynamic Scheduling Technique for Multi Processing System on Chip" Sukkur IBA Journal of Emerging Technologies, Vol.2, No.2, pp 46-53,2019

XXVIII. M. U. Hashmi, S. A. ZeeshanNajam, "Thermal-Aware Real-Time Task Schedulabilty test for Energy and Power System Optimization using Homogeneous Cache Hierarchy of Multi-core Systems" Journal of Mechanics of Continua and Mathematical Sciences, Vol.14, No.4, pp 442-452, 2023

XXIX. M. Y. A. Khan, U. Khalil, H. Khan, A. Uddin, S. Ahmed, "Power flow control by unified power flow controller" Engineering, Technology & Applied Science Research, Vol.9, No.2, pp 3900-3904, 2019

XXX. R. Waleed, A. Ali, S. Tariq, G. Mustafa, H. Sarwar, S. Saif, I. Uddin, "An Efficient Artificial Intelligence (AI) and Internet of Things (IoT's) Based MEAN Stack Technology Applications" Bulletin of Business and Economics (BBE), Vol.13, No.2, pp 200-206, 2024

XXXI. S. Khan, I. Ullah, M. U. Rahman, H. Khan, A. B. Shah, R. H. Althomali, M. M. Rahman, "Inorganic-polymer composite electrolytes: basics, fabrications, challenges and future perspectives" Reviews in Inorganic Chemistry, Vol.44, No.3, pp 1-29, 2024.

XXXII. S. Khan, I. Ullah, H. Khan, F. U. Rahman, M. U. Rahman, M. A. Saleem, A. Ullah, "Green synthesis of AgNPs from leaves extract of Salvia Sclarea their characterization, antibacterial activity and catalytic reduction ability" Zeitschrift für Physikalische Chemie, Vol.238, No.5, pp 931-947, 2024

XXXIII. T. M. Gondal, Z. Hameed, M. U. Shah, H. Khan, "Cavitation phenomenon and its effects in Francis turbines and amassed adeptness of hydel power plant" In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, pp 1-9, 2019