URLGuard: A Holistic Hybrid Machine Learning Approach for Phishing Detection

First Author[1], Second Author[2], Third Author[3] Font Size 12

Dr.Pradip Paithane
Mr.Siddheshwar Patil
Mr.Abhishek Kubde
Mr.Shubham Gaikwad
Mr.Rohit Mhetre

---***---

Abstract - The fast growth of Internet technology has significantly changed online users' experiences, while security concerns are becoming increasingly overpowering. Among these concerns, phishing stands out as a prominent criminal activity that uses social engineering and technology to steal a victim's identification data and account information. According to the Anti-Phishing Working Group (APWG), the number of phishing detections increased by 46 in the first quarter of 2018 compared to the fourth quarter of 2017. So to overcome these situations below paper introduces a phishing detection system using a hybrid machine learning approach based on URL attributes. It addresses the growing threat of phishing attacks that exploit email manipulation and fake websites to deceive users and steal sensitive data. The study employs a phishing URL dataset with over 11,000 websites, extracted from a reputable repository. After pre-processing, a hybrid machine learning model, which includes Decision Tree, Random Forest, and XGB is employed to safeguard against phishing URLs. The proposed approach undergoes evaluation with key metrics such as precision, accuracy, recall, F1-score, and specificity. Results demonstrate that the proposed method surpasses other models, achieving superior accuracy and efficiency in detecting phishing attacks.

Key Words: Anti-Phishing Working Group (APWG), Decision Tree, and Random Forest, and XGB, hybrid machine learning.

1. INTRODUCTION *(Size 11, Times New roman)*

In the present interconnected digital environment, the internet functions not only as a conduit facilitating the vast exchange of information but also as a platform utilized for engaging in malicious activities. One of the most widespread forms of these malicious activities is phishing, which entails the use of deceptive strategies by cybercriminals with the aim of misleading individuals into revealing confidential information such as their login credentials, financial particulars, or personal data.

Phishing attacks commonly manifest in the guise of seemingly authentic emails, messages, or websites that are meticulously crafted to dupe users into believing they are interacting with a reputable and trustworthy source. Despite the fact that the concept of phishing is not novel, its level of sophistication and prevalence have undergone significant transformations in recent times, thereby presenting formidable obstacles to individuals, enterprises, and institutions on a global scale. As reported by the Anti-Phishing Working Group (APWG), a staggering number exceeding 200,000 distinct phishing websites were identified solely in the initial quarter of the year 2023, thereby underscoring the magnitude and gravity of this perilous threat.

Furthermore, it is imperative to highlight the significant financial ramifications associated with phishing attacks, which are truly astounding, leading to the annual loss of billions of dollars as a result of deceitful activities carried out through these cunning methods. The primary objective of this academic inquiry is to thoroughly investigate the complex realm of phishing websites, delving into their intricate operations, distinguishing features, and profound effects on the realm of cybersecurity. Through a comprehensive analysis of the strategies utilized by malicious actors to fabricate and disseminate phishing websites, along with the approaches for identifying and containing these risks, the primary aim of this scholarly endeavor is to enrich the comprehension of the dynamic landscape of online fraudulence.

Through empirical analysis, case studies, and scholarly insights, this paper will provide valuable insights into the following key areas:

1. The anatomy of phishing websites: Understanding the design, structure, and functionality of phishing websites, including common tactics used to mimic legitimate entities and exploit human psychology.

2. Detection and classification techniques: Exploring the methodologies and technologies utilized to identify and categorize phishing websites, from heuristic analysis to machine learning algorithms.

3. Impacts and consequences: Assessing the multifaceted repercussions of phishing attacks on individuals, businesses, and society at large, including financial losses, reputational damage, and erosion of trust in online platforms.

4. Countermeasures and best practices: Examining proactive measures and defensive strategies aimed at combating phishing threats, ranging from user education and awareness campaigns to technical solutions such as email filtering and website authentication protocols.

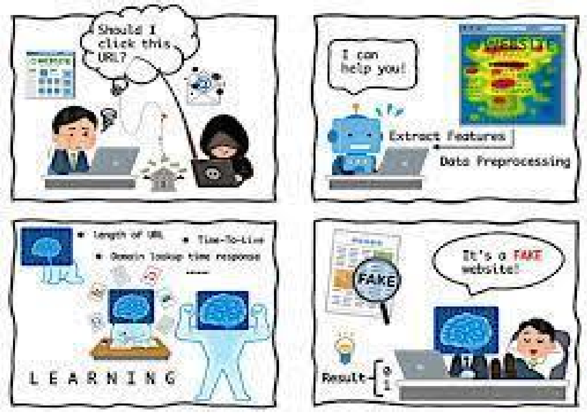


Figure 1 Phishing detection steps by applying AI solutions.

Objectives:

1. To create an ongoing detection system to spot phishing URLs.
2. To perform data preprocessing on phishing dataset (rows=11054, columns=33).
3. To introduce a hybrid model (DT+RF+XGB) for enhanced phishing detection.
4. Discuss and compare evaluation parameters to demonstrate the superiority of the proposed approach.

By shedding light on the intricate dynamics of phishing websites, this research endeavors to empower individuals and organizations with the knowledge and tools necessary to navigate the digital landscape safely and securely. Ultimately, the fight against phishing requires a concerted effort from all stakeholders, including cybersecurity professionals, policymakers, and end-users, to mitigate the risks posed by these insidious online threats.

2. Body of Paper

The body of the paper consists of numbered sections that present the main findings. These sections should be organized to best present the material.

It is often important to refer back (or forward) to specific sections. Such references are made by indicating the section number, for example, "In Sec. 2 we showed…" or "Section 2.1 contained a description…." If the word Section, Reference, Equation, or Figure starts a sentence, it is spelled out. When occurring in the middle of a sentence, these words are abbreviated Sec., Ref., Eq., and Fig.

At the first occurrence of an acronym, spell it out followed by the acronym in parentheses, e.g., charge-coupled diode (CCD).

Table -1: Sample Table format

IJSREM sample template format ,Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

3. Literature Survey

TABLE1………………………………..

3.1. Phishing detection using hybrid machine learning model

S. Raman Kumar Jog, ABDUL KARIM, MOBEEN SHAHROZ, KHABIB MUSTOFA, AND SAMIR BRAHIM BELHAOUARI[1]:MachineLearningModels:Logistic Regression (LR), Support Vector Classifier (SVC), and Decision Trees (DT) combined into a hybrid model. In summary, the hybrid model (LSD) that has been suggested seeks to improve phishing detection efficiency and accuracy.

It makes use of grid search, hyper parameter optimization, and canopy feature selection. On the other hand, false positive or negative rates are not discussed, and the study does not provide any information on scalability for huge datasets. Phishing Detection Using Machine Learning Techniques Machine Learning Models[8] : Logistic Regression, Ada Booster, Random Forest[9], K-Nearest Neighbour's (KNN) [10], Neural

Networks, Support Vector Machines (SVM) [11], Gradient Boosting, Naive Bayes [12], and XGBoost. The study focuses on phishing prevention techniques, including email filtering, user education, and real-time machine learning detection. It also gives us a detailed explanation of the workings of different machine learning models. However, it does not explore potential vulnerabilities or weaknesses in the machine learning models, and dataset details are omitted, impacting generalizability.

3.2. Phishing Detection Using Machine Learning Techniques

Machine Learning Models [2]: Logistic Regression,Ada Booster, Random Forest[16] [17] [20], K-Nearest Neighbors (KNN) [13] [18], Neural Networks, Support Vector Machines (SVM) [14], Gradient Boosting, Naïve Bayes [15], and XGBoost. The study focuses on phishing Prevention techniques, including email filtering, user education,and real-time machine learning detection.It also gives us a detailed explanation of the workings of different machine learning models. However, it does not explore potential vulnerabilities or weaknesses in the machine learning models, and dataset details are omitted,impacting generalizability.

In

3.3. Phishing detection using random forest with NLP-features

kOzgur Koray Sahingoz, Ebubekir Buber, Onder Demir, Banu Diri[5] Machine Learning Models: Random Forest with NLP-Based Features. The study presents a high accuracy real-time anti-phishing system that uses a variety of machine learning (ML) algorithms and features for URL-based phishing detection. Nevertheless, issues pertaining to NLP-based features are not examined, and the performances of the seven classification algorithms are not thoroughly analyzed[7]. The Na¨ıve Bayes classification is a probabilistic machine learning method, which is not only straightforward but also powerful. Due to its simplicity, efficiency and good performance, it is preferred in lots of application areas such as classification of texts, detection of spam emails/intrusions, etc. It is based on the Bayes theorem, which describes the relationship of conditional probabilities of statistical quantities.

3.4. Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism

YONG FANG, CHENG ZHANG, CHENG HUANG,LIANG LIU, AND YUE YANG[4]Machine Learning Models[12]: Deep learning model named THEMIS(advanced version of RCNN). THEMIS enhances embedding for improved performance detection. This approach allows for modeling emails at multiple levels, including the email header, email body, character level, and word level simultaneously, enhancing the model's ability to capture relevant features[13]. However, the study notes that some phishing emails without an email header may reduce efficiency and accuracy. It

is more accurate than previous methods, and it is also more efficient and robust.

1. The Bidirectional Long Short-Term Memory (Bi-LSTM) enhances the RCNNmodel. Next, an enhanced RCNN model is used to model the email at several levels.When as little noise as possible is introduced, the email's context can be better understood.
2. The email header and body are subjected to the attention mechanism, with varying weights allocated to each section. This allows the model to concentrate on more distinct and valuable data found in the email header and body.
3. On an unbalanced dataset, the THEMIS model presented in this work performs admirably. The accuracy is 99.848, and THEMIS's evaluation metrics are better than current detecting technology.

3.5. Hybrid ensemble feature selection framework for enhancing machine learning-based phishing detection systems

Kang Leng Chiew, Choon Lin Tan, Kok Sheik Wong, Kelvin S.C. Yong, Wei King Tiong[10] Machine Learning Models: Hybrid Ensemble Feature Selection (HEFS),Random Forest Classifier[15]. The study uses a real browser for feature extraction, improving robustness. However, detailed accuracy results for baseline features with classifiers other than Random Forest are provided. The phishing dataset used for benchmarking HEFS is not specified. According to experimental findings, HEFS works best when combined with the Random Forest classifier, identifying phishing and authentic websites with 94.6% accuracy while utilizing just 20.8 % of the original characteristics. It consists of two phases: the first phase uses the CDF-g algorithm to generate primary feature subsets, which are then used in a data perturbation ensemble to produce secondary feature subsets. The second phase derives a set of baseline features from the secondary feature subsets using a function perturbation ensemble.

4.MATERIAL AND METHODS

4.1.DATASET

The dataset was gathered and saved as a CSV file from the well-known Kaggle dataset repository, which offers benchmark datasets for academic use. The collection included 33 attributes and 11054 items that were taken from over 11,000 websites. Some characteristics that help distinguish between phishing and legitimate website URLs are UsingIP, LongURL, ShortURL, Symbol@, Redirecting//, PrefixSuffix-, Sub-Domains, HTTPS, DomainRegLen, Favicon, NonStdPort, HTTPSDomainURL, Reques-tURL, AnchorURL, LinksIn-ScriptTags, and ServerFormHandler. As seen in Figure 4, the dataset was divided into two classes: phishing and legitimate. The dataset needs to be improved because it was in vector form. To prepare the dataset for preprocessing, the null values were eliminated.The entire dataset was preprocessed and then combined into a single corpus and put to use in further processing. The entire corpus was split into two parts: 70% for training and 30% for testing. The machine learning model was trained *using 70% of the* training data, while 30% of the data was kept for predictions. The performance of the suggested method was also assessed.

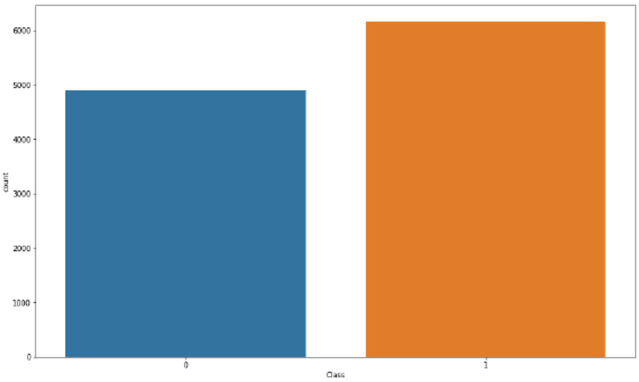


Figure 2

Dataset presentation according to number of classes phishing and legitimate, where (1) presents phishing and (0) legitimate URLs.

4.2. Canopy-based feature selection

Centroid selection and canopy clustering are the two main processes in canopy centroid selection. Two distance thresholds, an inner threshold (T2) and an outer threshold (T1), where T1 is greater than T2, create a canopy, which is a representation of a cluster. Data points are allocated to canopies in the clustering phase according on how close they are to the canopy center. Points that are located inside the inner threshold (T2) are specifically allocated to the same canopy. Each canopy is represented by a centroid once the canopies have formed. The mean or median of the feature values at each position inside the canopy is used to calculate this centroid. By using this technique, the centroids are guaranteed to correctly depict the data points' central tendency inside each canopy.

4.3. Cross fold validation

Cross-fold validation is a reliable technique for assessing model effectiveness. The dataset is partitioned into k fold sections of approximately similar size, usually k = 5 or 10. The model is trained and evaluated k times as the procedure iterates across these folds. Every iteration uses a different fold as the test set and uses the rest of the folds for training. The model's dependability is increased by this method, which guarantees that every data point is used for both training and testing. A more reliable assessment of the model's overall performance is obtained by averaging or otherwise aggregating the performance measures from each fold after all iterations. By reducing the bias and variation brought about by a single train-test split, this technique provides a thorough analysis of the model.

4.4. grid search

Grid search is an organized technique for fine-tuning hyperparameters by iteratively going through a predetermined list of possible combinations. This method assesses the performance of the model by doing cross-validation for every combination of hyperparameters. Next, the hyperparameter set with the best performance is chosen. Grid search was used for this project with the following hyperparameter configurations: 'xgb__max_depth' with values [3, 5, 7], 'dt__max_depth' with values [None, 5, 10], and 'rf__n_estimators' with values [50, 100, 200]. The most ideal hyperparameters were found by analyzing these combinations, which enhanced and strengthened the model's performance. This approach makes sure that the hyperparameter space is well explored, which improves the efficacy of the model.

4.5. APPLIED MACHINE LEARNING ALGORITHMS

Algorithms for machine learning are computational tools that let computers learn from data and come to conclusions or predictions. They fall into three categories: reinforcement learning, where agents learn to interact with surroundings to accomplish goals by trial and error; supervised learning, where models learn from labeled data; and unsupervised learning, where patterns are found from unlabeled data. A subset called deep learning uses sophisticated neural networks to interpret natural language and recognize images. Several models are combined in ensemble methods to improve performance. Numerous applications, including financial forecasts, driverless cars, medical diagnostics, and recommendation systems, are powered by these algorithms.

In the above paper to find phishing websites we have used hybrid machine learning model containing decision tree, random forest and XGboost, which provided different accuracy for different feature engineering techniques.

4.5.1. Decision Tree

A decision tree [40] - a basic representation that classifies instances. A decision tree Constitutes of the following:

- Nodes: specic attributes' estimation is tested by nodes.
- Branches: they are the interface with following nodes or the leaf nodes and relatesto the result.
- Leaf nodes: Nodes that are terminal and anticipate the result.

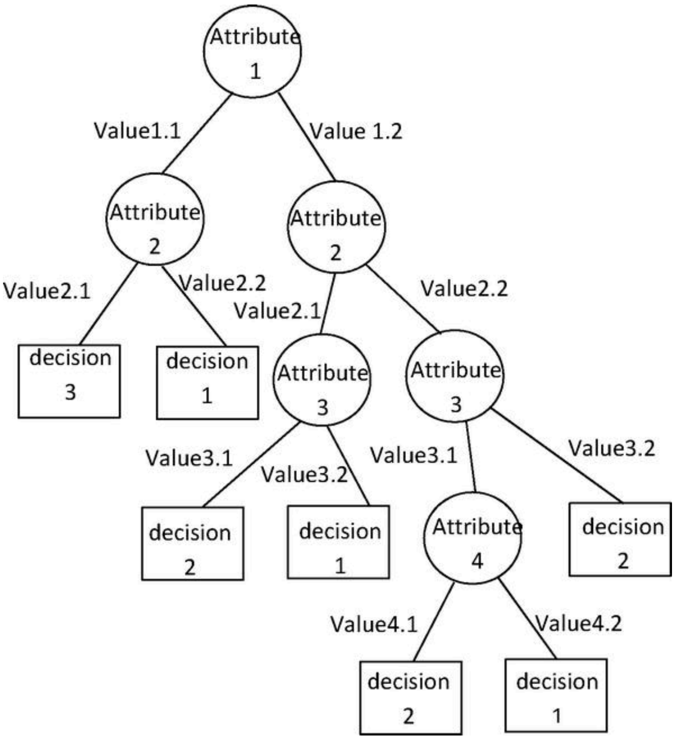


Figure 3 General Representation of Decision Tree

In the figure, the decision tree starts with an initial question at the top node, labeled “Attribute 1”. Depending on the answer (Value 1.1 or Value 1.2), the data is split into two branches. This process is then repeated for each branch, using a different attribute as the test question at each node. The decision tree makes its classifications at the terminal nodes, which are labeled with the final decision.

For instance, consider the leftmost branch of the tree in the figure. If the answer to the question at the top node is “Value 1.1”, then the next question is “Attribute 2”. If the answer to that question is “Value 2.1”, the data is split again, following the question labeled “Attribute 3”. If the answer to that question is “Value 3.1”, then the decision tree reaches a terminal node labeled “decision 2”. This means that the instance belongs to class “decision 2”.

1.Entropy (E(S)) : Entropy measures the uncertainty or randomness in a dataset.

$$E(S) \ = - \sum_{i=1}^{c} Pilog2(pi)$$

2.Conditional Entropy(E(T,X)): Conditional entropy measures the amount of uncertainty in T (target variable) given X (predictor variable).

$$E(T, X) \ = - \sum_{c \in X} p(c). E(c)$$

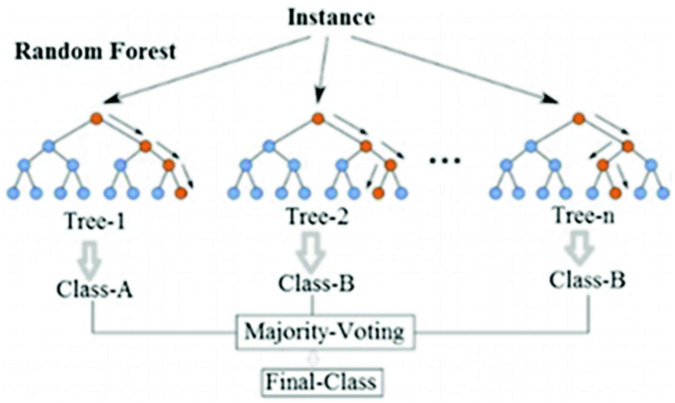
3.Information Gain(IG(T,X)): Information gain measures the reduction in entropy when a dataset is split by the values of a predictor variable.

$$IG(T,X) = E(T) - E(T,X)$$

Where, S is a set,
c = the class or values in set,
Pi = probability of occurrence of elements i in set S,
T is a set, and X is an attribute or feature,
p(c) is the probability of occurrence of value c in set X.

4.5.2.Random Forest

Random Forest is an ensemble learning method widely used for classification and regression tasks. It comprises decision trees trained on random subsets of the data, with randomness introduced both in bootstrapped sampling and feature selection. During prediction, individual tree outputs are either averaged (for regression) or combined through majority voting (for classification). This randomness helps prevent overfitting and improves robustness. Random Forest handles high-dimensional data well and provides feature importance scores. However, it can be computationally expensive and may not perform optimally on highly imbalanced datasets. Its simplicity and effectiveness make it a popular choice in various domains, including fishing detection. By leveraging its ensemble nature and randomization techniques, Random Forest can contribute to hybrid machine learning models aimed at enhancing fishing detection system accuracy and robustness.



The diagram illustrates a random forest classifier, which is a machine learning technique used for classification. It works by creating a collection of decision trees, each of which uses a random subset of features. New data points are then passed through each tree, and the most frequent class prediction is chosen as the overall prediction.

The left side of the diagram shows a random forest with multiple decision trees. Each tree is created using a random subset of features from the available data. The right side shows how a new instance is classified. The instance is passed through each tree in the forest, and each tree makes a prediction about the class of the instance. The final classification is determined by a majority vote of the trees.

In essence, random forests improve the accuracy and stability of classification models by reducing the variance of decision trees.

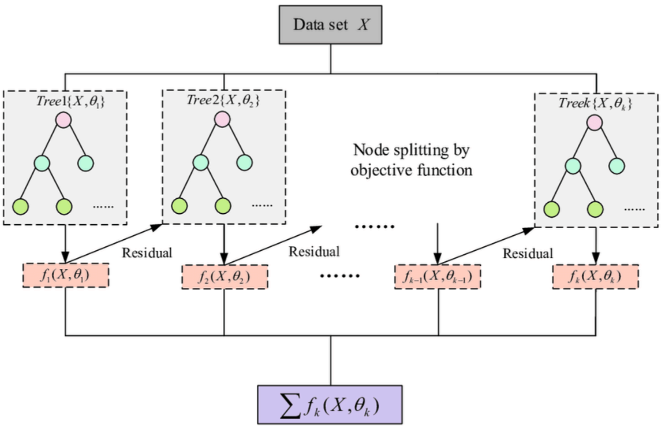
1.Ensemble Method: The ensemble model aggregates the predictions of multiple base models to make a final prediction.

$$F(Xt) = \frac{1}{B}\sum_{i=0}^{B} Fi(Xt)$$

Where,
F(Xt) is the output of the random forest for the input Xt,
B is the number of trees in the random forest,
Fi(Xt) is the output of the ith tree in the forest for the input Xt,

4.5.3.XGBoost

Extreme Gradient Boosting, or XGBoost for short, is a sophisticated version of gradient boosting machines. It is well known for its effectiveness in supervised learning tasks as well as its scalability and efficiency. XGBoost sequentially constructs an ensemble of weak prediction models, usually decision trees. By optimizing a predetermined objective function—typically a mix of the loss function and regularization term—each new model fixes the flaws of its predecessors. XGBoost, in particular, presents regularization methods including shrinkage (learning rate) and feature column subsampling to avoid overfitting. In addition, it makes effective use of hardware resources by parallelizing training inside a distributed computing architecture. Because of its versatility in solving classification, regression, and ranking problems, XGBoost is a well-liked option for both real-world and data science applications where speed and accuracy are crucial.



A dataset with the label "Data Set" is at the top of the tree. Then, depending on the values of the characteristics in the data, it divides into many branches. The example shows that an unknown feature (X) with a value of 3 provides the basis for the first split.Different results are represented by each branch according to the split decision. "Tree1{X,03}" is reached via the left branch, and "Tree2{X,02}" by the right branch. This process continues until the data reaches a leaf node, which is shown by the rectangles with the labels "f(X,0)" and "f(X,02)" at the bottom. These leaf nodes hold the forecast of the model for that specific data route.The total of the predictions for each leaf is indicated by the phrase "Σ(X,0)" in the bottom right.

1.Objective Function:

$$\text{Objective} = \sum_{i=1}^{n} \text{Loss}(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(\text{fk})$$

Where,

n is the number of samples.
ŷi is the predicted value for the ith sample.
yi is the true label for the ith sample.
K is a parameter representing some set of values.

2.Gradient Boosting:

$$\hat{y}(\text{t}) = \sum_{k=1}^{t} f_k(\text{X})$$

Where,

t is the iteration or boosting round.
fk(X) is the prediction of the Kth model at input X.

3.Regularization:

$$\Omega(f_k) = \text{৭}\mathbf{T} + \frac{1}{2}\lambda \sum_{j=1}^{T} W_j^2 + \propto \sum_{j=1}^{T} \left|W_j\right|$$

Where,

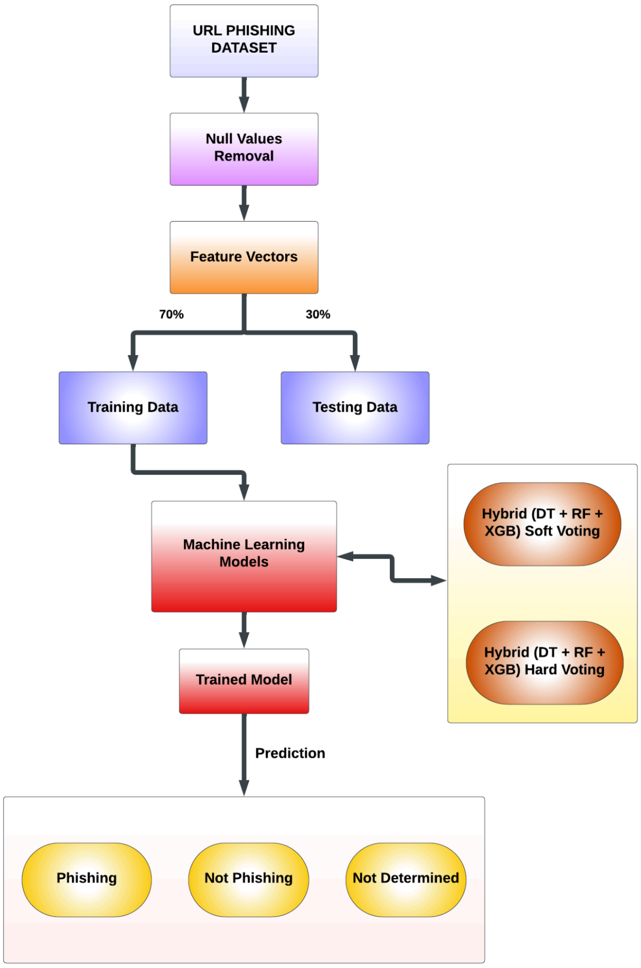
T is the number of leaves in the tree,
W_j is the weight assigned to the jth leaf,
৭ , λ and ∝ are regularization parameters.

4.4.4.Hybrid Model(RF+DT+XGB)

Utilizing the advantages of each approach, a hybrid model that combines XGBoost (XGB), Random Forest (RF), and Decision Trees (DT) enhances prediction performance in a variety of areas. Known for its gradient boosting architecture, XGBoost minimizes overfitting by using regularization techniques like shrinkage and feature subsampling. It is an excellent tool for managing complicated relationships and huge datasets. By combining predictions from several decision trees trained on various data subsets, Random Forest, an ensemble of decision trees, offers resilience against noise and outliers. Decision trees are helpful in comprehending feature importance and linkages within the data because of their interpretability and simplicity.XGBoost functions as the main learner in the hybrid model, identifying intricate patterns in the data. By adding more variety to predictions, Random Forest enhances XGBoost and lowers variance while enhancing generalization. Decision trees serve as interpretable elements that facilitate feature analysis and model comprehension.

Depending on the objective, the hybrid model uses voting or averaging to mix the Decision Tree, Random Forest, and XGBoost outputs during prediction. Through the use of an ensemble technique, which balances the strengths and weaknesses of each member, a predictive model that is reliable and accurate is produced that can be used to a variety of machine learning tasks, such as regression, classification, and anomaly detection.

5. Architecture



The method of developing a machine learning model to identify phishing URLs is shown in the diagram.

The "URL Phishing" dataset serves as the process's starting point. It's possible that some of the URLs in this dataset have been flagged as phishing or authentic. The process of "Null Values Removal" is applied to the data, implying that any missing values in the dataset are being eliminated prior to model training. "Feature Vectors" is the next stage, when the data is converted into a format that can be interpreted by the machine learning model. This probably entails taking characteristics from the URLs, such the length of the domain name or the existence of specific keywords."Training Data" (70%) and "Testing Data" (30%) are the two sets of data that are created after that. The machine learning model is taught using the training data, and its performance is assessed using the testing data.It is clear from the statement "Machine Learning Models" that the training data is being used to train several models. Although the exact models aren't stated, the language that follows implies that they are ensemble models that integrate the predictions of three distinct machine learning models: XGBoost, Random Forest, and Decision Tree. The models' predictions may be combined in two ways: "Soft Voting" and "Hard Voting."The testing data is used to assess the models once they have been trained. The graphic does not display the model's performance on the testing data.Lastly,a new URL is put through the trained model to predict whether it’s a phishing site or not. The model outputs a classification: “Phishing”, “Not Phishing," or “Not Determined.”.

6. Evaluation Parameter

A number of evaluation criteria must be used to assess machine learning performance. Results from the machine-learning system are given as forecasts. The number of accurate and inaccurate predictions the model makes in both legitimate and phishing classes is measured by the evaluation parameters. There were several parameters used, including the F1-score, recall, specificity, accuracy, and precision.

Model performance is measured by accuracy, which is expressed as the number of correct predictions the model produces, as indicated by Equation

$$\text{Accuracy} = \frac{((TP + TN))}{(TP + TN + FP + FN)}$$

The evaluation parameter known as precision is utilized to analyze the models; it indicates the frequency at which a classifier stays accurate when we wish to forecast the positive class. Precision expresses the degree to which the model classifies the phishing URLs and gauges the positive rate, or the degree to which the model predicts the positive values. With regard to precision, each classifier fared well.

$$\text{Precision} = \frac{TP}{(TP + FP)}$$

A metric for analyzing classification models that indicates how many times the model correctly identified out of all the possible positive labels. The classifier accurately identifies the classifier for both sorts of URLs in order to predict both phishing and authentic URLs.

$$\text{Recall} = \frac{Tp}{(TP + FN)}$$

The F1score is the harmonic mean of precision and recall, where the F1 score reaches its best value (perfect precision and recall). The general formula is as follows:

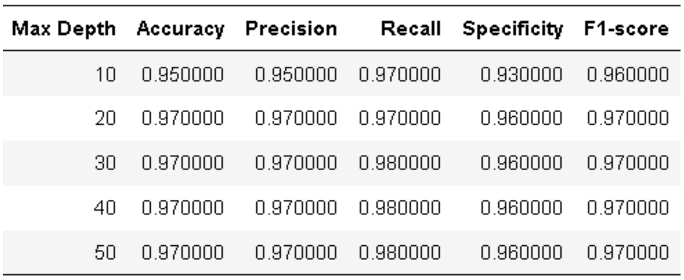
$$\text{F1Score} = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

Therefore, an F1 score is required when asking about the balance between precision and recall.

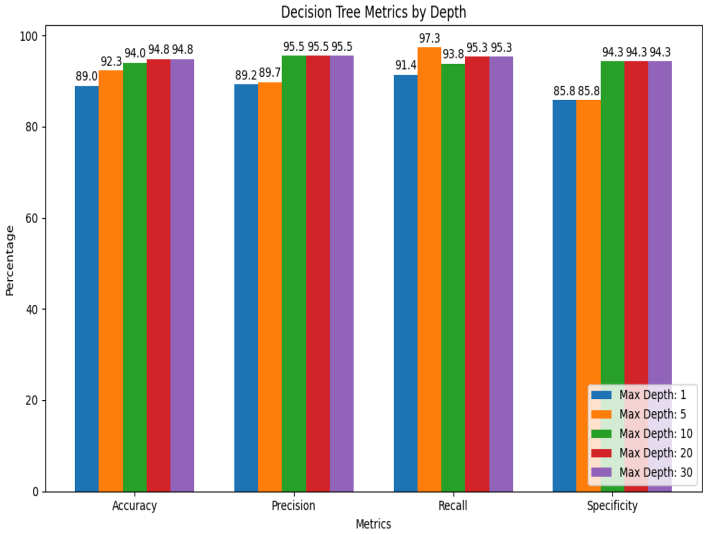
7. Results and Discussion

The internet is a vast network-based industry full of hackers, attackers, or cybercriminals. Civilians, businessmen, industries, and every market that consists of the Internet and networks need security to prevent phishing and provide protection to their customers, as well as to their own system safety. The methodology proposed in this study was successfully implemented as a prototype using a dataset comprising phishing and legitimate URLs. These experiments are carried out using many machine learning algorithms that are discussed separately in each heading to evaluate and illustrate the effects of the machine learning algorithms that are given below.

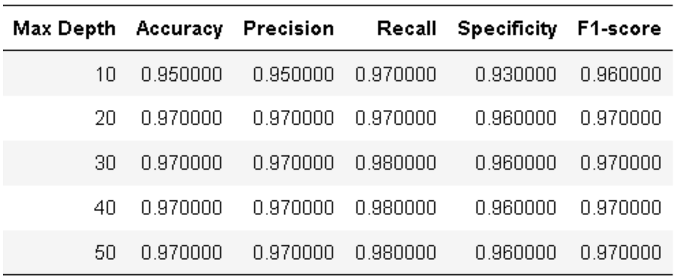
7.1. Experimental Results of Decision Tree



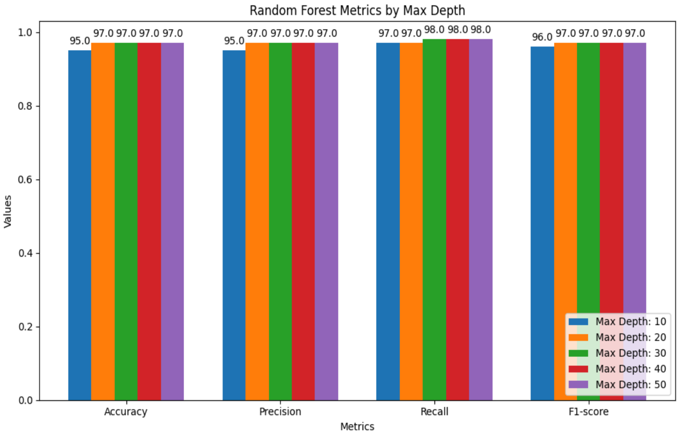
Max Depth	Accuracy	Precision	Recall	Specificity	F1-score
10	0.950000	0.950000	0.970000	0.930000	0.960000
20	0.970000	0.970000	0.970000	0.960000	0.970000
30	0.970000	0.970000	0.980000	0.960000	0.970000
40	0.970000	0.970000	0.980000	0.960000	0.970000
50	0.970000	0.970000	0.980000	0.960000	0.970000



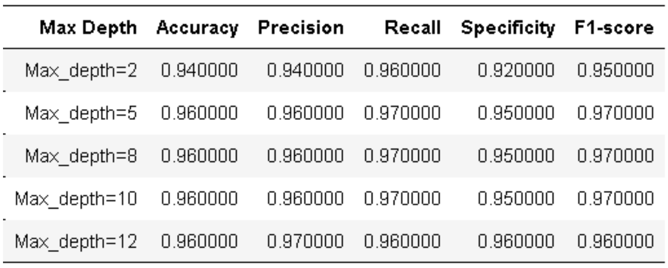
7.2. Experimental Result of Random Forest



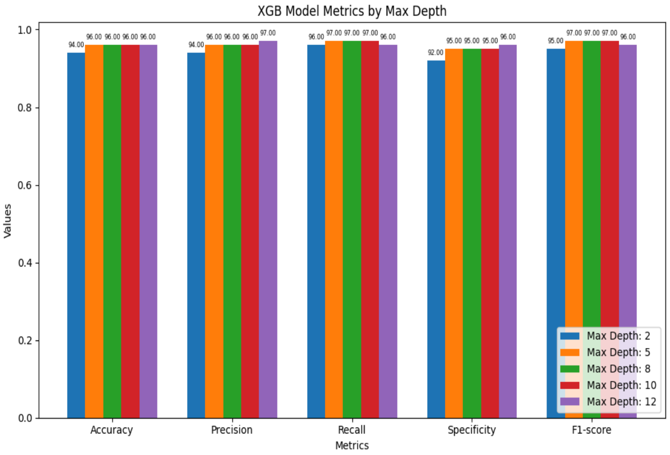
Max Depth	Accuracy	Precision	Recall	Specificity	F1-score
10	0.950000	0.950000	0.970000	0.930000	0.960000
20	0.970000	0.970000	0.970000	0.960000	0.970000
30	0.970000	0.970000	0.980000	0.960000	0.970000
40	0.970000	0.970000	0.980000	0.960000	0.970000
50	0.970000	0.970000	0.980000	0.960000	0.970000



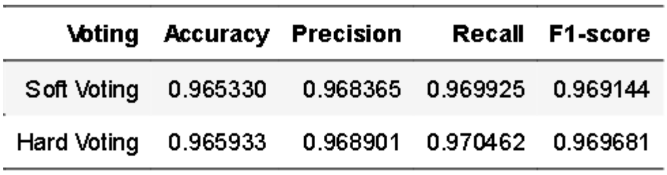
7.3. Experimental Result of XGBoost



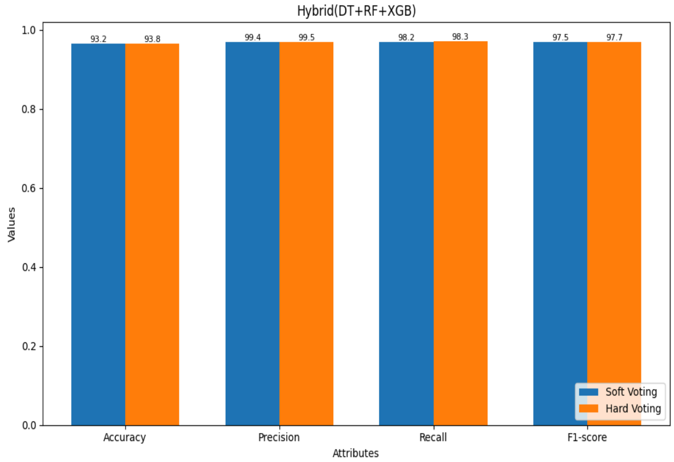
Max Depth	Accuracy	Precision	Recall	Specificity	F1-score
Max_depth=2	0.940000	0.940000	0.960000	0.920000	0.950000
Max_depth=5	0.960000	0.960000	0.970000	0.950000	0.970000
Max_depth=8	0.960000	0.960000	0.970000	0.950000	0.970000
Max_depth=10	0.960000	0.960000	0.970000	0.950000	0.970000
Max_depth=12	0.960000	0.970000	0.960000	0.960000	0.960000



7.4. Experimental Result of Hybrid Model



Voting	Accuracy	Precision	Recall	F1-score
Soft Voting	0.965330	0.968365	0.969925	0.969144
Hard Voting	0.965933	0.968901	0.970462	0.969681



8. Future Scope

- User Interface Design: Create an intuitive and user-friendly interface for systemadministrators, facilitating easy management, monitoring, and response to potentialphishing threats.
- Security Collaboration: Foster collaboration with the wider cybersecurity community to stay informed about the latest threats and best practices, ensuring the system's relevance and effectiveness.
- Continuous Improvement: Establish procedures for continuous monitoring, evaluation, and updating of the system to remain effective against evolving phishing tactics over time.

9.Conclusion

We have successfully developed and implemented a comprehensive URL-based phishing detection system. This system leverages a variety of machine learning algorithms to effectively identify and thwart phishing attempts, particularly those involving deceptive URLs. The project's objectives, including enhancing accuracy, real-time monitoring,and user-friendly interface design, have been achieved.Through continuous monitoring and collaboration with the cybersecurity community, the system remains adaptive and resilient against evolving phishing tactics.

REFERENCES

1. Zouina, M., Outtaj, B.: A novel lightweight url phishing detection system using svm and similarity index. Human-centric Computing and Information Sciences 7(1), 1–13 (2017).
2. Wang, S., Khan, S., Xu, C., Nazir, S., Hafeez, A.: Deep learning-based efficient model development for phishing detection using random forest and blstm classifiers. Complexity 2020, 1–7 (2020)
3. Abdelhamid, N., Ayesh, A., Thabtah, F.: Phishing detection based associative classification data mining. Expert Systems with Applications 41(13), 5948–5959 (2014)
4. Karim, A., Shahroz, M., Mustofa, K., Belhaouari, S.B., Joga, S.R.K.: Phishing detection system through hybrid machine learning based on url. IEEE Access 11, 36805–36822 (2023).
5. Sahingoz, O.K., Buber, E., Demir, O., Diri, B.: Machine learning based phishing detection from urls. Expert Systems with Applications 117, 345–357 (2019)

6. Shahrivari, V., Darabi, M.M., Izadi, M.: Phishing detection using machine learning techniques. arXiv preprint arXiv:2009.11116 (2020)

7. Buber, E., Diri, B., Sahingoz, O.K.: Nlp based phishing attack detection from urls. In: Intelligent Systems Design and Applications: 17th International Conference on Intelligent Systems Design and Applications (ISDA 2017) Held in Delhi, India, December 14-16, 2017, pp. 608–618 (2018). Springer

8. Paithane, P.M.: Random forest algorithm use for crop recommendation. ITEGAM-JETIA 9(43), 34–41 (2023)

9. Paithane, P.M.: Yoga posture detection using machine learning. Artificial Intelligence in Information and Communication Technologies, Healthcare and Education: A Roadmap Ahead 27 (2022).

10. Chiew, K.L., Tan, C.L., Wong, K., Yong, K.S., Tiong, W.K.: A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. Information Sciences 484, 153–166 (2019)

11. Paithane, P., Kakarwal, S.: Lmns-net: Lightweight multiscale novel semantic-net deep learning approach used for automatic pancreas image segmentation in ct scan images. Expert Systems with Applications 234, 121064 (2023)

12. Fang, Y., Zhang, C., Huang, C., Liu, L., Yang, Y.: Phishing email detection using improved rcnn model with multilevel vectors and attention mechanism. IEEE Access 7, 56329–56340 (2019).

13. Rao, R.S., Pais, A.R.: Detection of phishing websites using an efficient featurebased machine learning framework. Neural Computing and applications 31, 3851– 3873 (2019).

14. Sonowal, G., Kuppusamy, K.: Phidma–a phishing detection model with multifilter approach. Journal of King Saud University-Computer and Information Sciences 32(1), 99–112 (2020).

15. Alam, M.S., Vuong, S.T.: Random forest classification for detecting android malware. In: 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, pp. 663–669 (2013). IEEE.

16. Smadi, S., Aslam, N., Zhang, L.: Detection of online phishing email using dynamic evolving neural network based on reinforcement learning. Decision Support Systems 107, 88–102 (2018).

17. Paithane, P.M., Kakarwal, S.: Automatic pancreas segmentation using a novel modified semantic deep learning bottom-up approach. International Journal of Intelligent Systems and Applications in Engineering 10(1), 98–104 (2022).

18. Buber, E., Dırı, B., Sahingoz, O.K.: Detecting phishing attacks from url by using nlp techniques. In: 2017 International Conference on Computer Science and Engineering (UBMK), pp. 337–342 (2017). IEEE.

19. Feng, F., Zhou, Q., Shen, Z., Yang, X., Han, L., Wang, J.: The application of a novel neural network in the detection of phishing websites. Journal of Ambient Intelligence and Humanized Computing, 1–15 (2018).

20. Wagh, S.J., Paithane, P.M., Patil, S.: Applications of fuzzy logic in assessment of groundwater quality index from jafrabad taluka of marathawada region of maharashtra state: A gis based approach. In: International Conference on Hybrid Intelligent Systems, pp. 354–364 (2021). Springer.

21. Shirazi, H., Hayne, K.: Towards performance of nlp transformers on url-based phishing detection for mobile devices. International journal of ubiquitous systems and pervasive networks (2022)

1'st
Author
Photo