URLGuard: A Holistic Hybrid Machine Learning Approach for Phishing Detection

Mr. Siddheshwar Patil

Mr. Rohit Mehetre

Mr. Abhishek Kubde

Mr. Shubham Gaikwad

under supervision of

Dr. Pradip Paithane

Department of Artificial Intelligence and Data Science ,
Vidya Pratishthans Kamalnayan Bajaj Institute of Engineering and Technology,
Vidyanagari, Baramati-413133

Academic Year: 2023-24

Contents

- Introduction
- Problem Statement
- Objectives
- Architecture
- Mathematical Modeling
- Algorithm Development
- Results
- Performance Evaluation Parameters
- GUI
- Conclusion
- References

Introduction

- The phishing attacks poses a significant threat to individuals and organizations alike. To combat the escalating threat of phishing attacks on the internet effectively, the comprehensive URL-based phishing detection system utilizing a variety of machine learning algorithms.
- As we explore the intricacies of such a system, we uncover the pivotal role it plays in ensuring a secure digital environment for users, ultimately contributing to the ongoing battle against cyber threats.
- It introduces the concept of hybrid machine learning, emphasizing its potential to enhance detection accuracy by combining the strengths of multiple algorithms.

Introduction

- Phishing detection using hybrid machine learning models outlines the growing threat of phishing attacks in the digital era and highlights the need for advanced, accurate detection methods.
- 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(0.96), accuracy(0.96), recall(0.97),and F1-score(0.96). Results demonstrate that the proposed method surpasses the state of art.

Problem statement

To combat the escalating threat of phishing attacks on the internet effectively, the comprehensive URL-based phishing detection system utilizing a variety of machine learning algorithms.

Objectives

- To create a ongoing detection system to spot phishing URLs.
- To introduce a hybrid model (DT+RF+XGB) for enhanced phishing detection.
- Discuss and compare evaluation parameters to demonstrate the superiority of the proposed approach.

Architecture

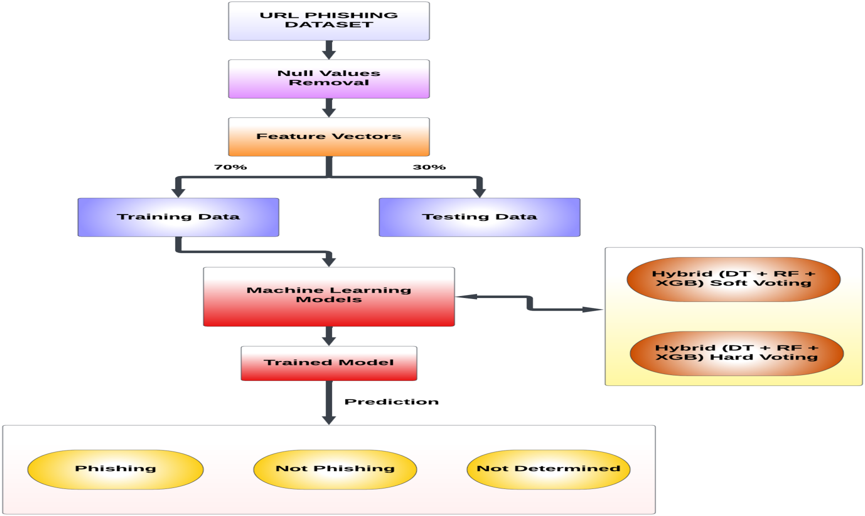

Figure: Architecture

Mathematical Modeling

- **Random Forest:**Random Forest is an ensemble method using decision trees on random data subsets, with randomness in sampling and feature selection, preventing overfitting and improving robustness. It handles high-dimensional data and offers feature importance scores.
    - Ensemble Method

$$(F(x_t) = \frac{1}{B}\sum_{i=0}^{B} F_i(x_t) \qquad (1)$$

- $F(x_t)$ is the output of the random forest for the input x_t,
- B is the number of trees in the random forest,
- $F_i(x_t)$ is the output of the i-th tree in the forest for the input x_t,
- $\sum$ denotes the summation over the specified range.

Mathematical Modeling

XGB: Extreme Gradient Boosting (XGBoost) enhances gradient boosting for supervised learning, known for scalability and efficiency. It builds an ensemble of weak models, typically decision trees, optimizing an objective function to correct previous models errors.

- Objective Function

$$\text{Objective} = \sum_{i=1}^{n} \text{Loss}(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2}$$

- n is the number of samples.
- $\hat{y}_i$ is the predicted value for the i-th sample.
- y_i is the true label for the i-th sample.
- K is a parameter representing some set of values.

Mathematical Modeling

- Gradient Boosting

$$\hat{y}(t) = \sum_{k=1}^{t} f_k(x) \tag{3}$$

  - t is the iteration or boosting round.
  - $f_k(x)$ is the prediction of the k-th model at input x.

- Regularization

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 + \alpha \sum_{j=1}^{T} |w_j| \tag{4}$$

  - T is the number of leaves in the tree.

- w_j is the weight assigned to the j-th leaf.
- γ, λ, and α are regularization parameters.

Decision tree:A decision tree is a tree-like model in data analysis and machine learning. It splits data based on key attributes, creating an interpretable flowchart. It handles categorical and numerical data, effective for classification and regression tasks.

- **Entropy (E(S)):** $\mathrm{E(S)} = -\sum_{i=1}^{c} p_i \log_2(p_i)$
- **Conditional Entropy (E(T, X)):**

$$E(T, X) = -\sum_{c \in X} p(c) \cdot E(c) \tag{5}$$

Mathematical Modeling

- **Information Gain (IG(T, X)):**

$$IG(T, X) = E(T) - E(T, X) \tag{6}$$

  - S is a set,
  - c represents the classes or values in the set,
  - p_i is the probability of occurrence of element 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,
  - $E(c)$ represents the entropy of the subset of T associated with value c, and
  - $\sum$ denotes the summation over the specified range.

Algorithm Development

Algorithm 1: Hybrid Machine Learning Model (RF + DT + XGB)

Input: Training data X_{train}, y_{train}, Test data X_{test}, y_{test}

Output: Predicted class label

```
train_and_test_hybrid_model(X_train, y_train, X_test, y_test, num_of_epochs,
 param_grid):
    initialize rf_model ← RandomForestClassifier()
    initialize xgb_model ← XGBClassifier()
    initialize dt_model ← DecisionTreeClassifier()
    initialize estimators ← [('rf', rf_model), ('xgb', xgb_model),
 ('dt', dt_model)]
    initialize best_model ← None
    for epoch ← 1 to num_of_epochs do
        initialize grid_search ←
 GridSearchCV(estimator=VotingClassifier(estimators),
 param_grid=param_grid, cv=5)
        grid_search.fit(X_train, y_train)
        best_model ← grid_search.best_estimator_
        best_model.fit(X_train, y_train)
        // Evaluate on the test set and print results
        validation_results ← best_model.evaluate(X_test, y_test)
        print(Accuracy: validation_results)
    end
return best_model.predict(X_test)
```

Figure: Proposed Algorithm

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.
- Datasetlink: https://www.kaggle.com/code/eswarchandt/website phishing/input?select=phishing.csv

Results

Algorithm	Accuracy	Precision	Recall	F1 score
Decision Tree (DT)	0.94	0.94	0.94	0.94
Random Forest (RF)	0.96	0.92	0.93	0.92
Extreme Gradient Boosting (XGB)	0.96	0.96	0.96	0.96
DT + RF + XGB (Soft Voting)	0.96	0.96	0.96	0.96
DT + RF + XGB (Hard Voting)	0.96	0.97	096	0.96

Figure: Result Comparison

Results

Sr.no	Author	Algorithm	Evaluation Parameters
1	Proposed algorithm	Hybrid Machine Learning Model(DT+RF+XGB)	• Accuracy= 0.96 • precision=0.96 • Recall= 0.97 • F1-Score=0.96
2	ABDULKARIM1, MOBEEN SHAHROZ 2, KHABIB MUSTOFA 1, SAMIR BRAHIM BELHAOUARI 3, ANDS. RAMANA KUMAR JOGA(2023)	Hybrid Machine Learning Model(SVC + DT +LR)r	• Accuracy= 0.95 • Precision=0.95 • Recall=0.96 • F1-Score=0.95
3	Sonowal, G., Kuppusamy, K(2020)	five-layer phishing detection model called PhiDMA	• Accuracy= 0.92 • Precision=0.91 • Recall=0.90 • F1-Score=0.90

Results

4	KangLengChiew a, ChoonLinTan a,, KokSheik Wong b, Kelvin S.C.Yongc,Wei KingTionga(2019)	Hybrid Ensemble Feature Selection(HEFS),Random ForestClassifier	• Accuracy= 0.94
5	YONG FANG, CHENG ZHANG, CHENG HUANG, LIANG LIU,ANDYUE YANG (2019)	Deep learning model named THEMIS(advanced version of RCNN)	• Accuracy= 0.99 • Precision=0.99 • Recall=0.99 • F1-Score=0.99
6	Ozgur Koray Sahingoz, Ebubekir Buber b,OnderDemirb, BanuDiri(2017)	Using Random Forest with NLP-based features	• Accuracy= 0.97 • Precision=0.97 • Sensitivity=0.99 • F1-Score=0.98

Figure: Result Comparison

Results

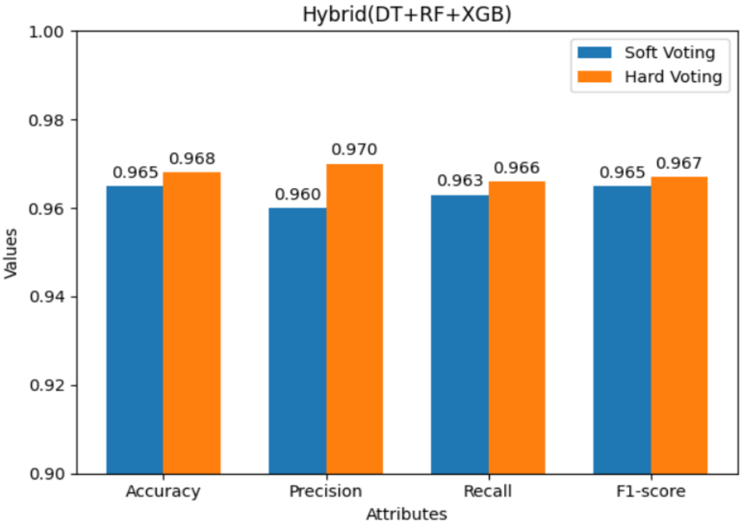

Figure: Results

GUI

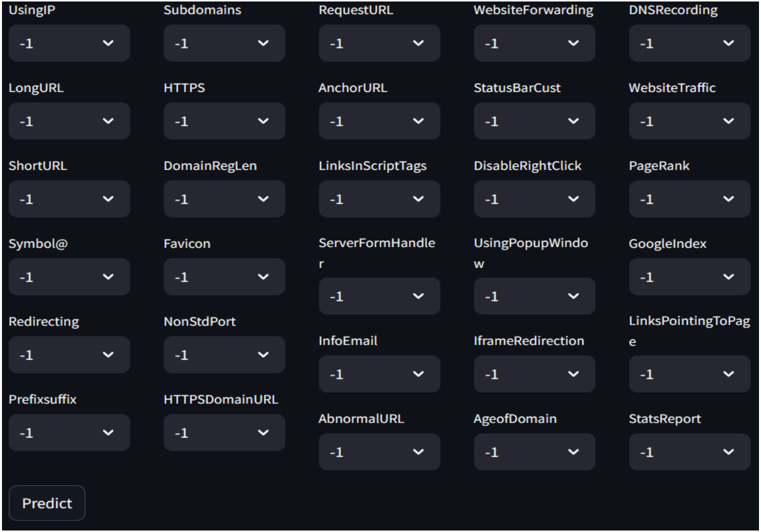

Figure: GUI

Conclusion

- `https://github.com/Rohit9860/Phishing-detection`
- The hybrid machine learning models (DT+RF+XGB) boost phishing detection accuracy, paving the way for future cybersecurity research.

References

- A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari and S. R. K. Joga, "Phishing Detection System Through Hybrid Machine Learning Based on URL," in IEEE Access, vol. 11, pp. 36805-36822, 2023, doi: 10.1109/ACCESS.2023.3252366.
- V. Shahrivari, M. M. Darabi, and M. Izadi, "Phishing detection using machine learning techniques," 2020, arXiv:2009.11116.
- O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," Expert Syst. Appl., vol. 117, pp. 345–357, Mar. 2019

References

- Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism," IEEE Access, vol. 7, pp. 56329–56340, 2019.
- K. L. Chiew, C. L. Tan, K.Wong, K. S. C. Yong, andW. K. Tiong, "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system," Inf. Sci., vol. 484, pp. 153–166, May 2019.


THANK YOU