Fraudulent Website Detection with Nonparametric Based Modelling [⋆]

Xinyi (Vicky) Xiang

University of Washington

Abstract. Phishing websites are a popular tool that attempts to present false situations and scam users to disclose their private information. These scams often disguise as legitimate companies or institutions such as banks and email providers. Based on the complaints submitted to the Internet Crime Complaint Center (IC3), in 2020 solely, 791,790 total complaints were received by the Center, and an estimated loss exceeding $4.2 billion was reported. In real-life classification scenarios, we find models that requires no assumptions about the precise distribution of the samples, such as nonparametric statistical modelling, more suitable and scalable for prediction and detection purposes. In this report, we present promising results shown when using the Random Forest model and the Extreme Gradient Boosting (XGBoost) model comparing to supervised model such as the support-vector machines (SVM) in malicious website detection, when reliability is estimated directly from data.

Keywords: Fraud Intelligence · Loss Prevention · Anomaly Detection · Nonparametric Statistics.

1 Dataset Information

The dataset used in this project is obtained from the Phishing Websites Data Set collected mainly from PhishTank archive, MillerSmiles archive, Google's searching operators in UCI Machine Learning Repository. This dataset covers important features that have proved to be sound and effective in detecting phishing websites, with novel features proposed by the researchers in addition.

2 Model Architecture Setup

2.1 XGBoost

XGBoost is a type of ensemble tree method that apply the principle of boosting weak learners, generally Classification And Regression Trees (CARTs), using the gradient descent architecture. However, it is with observable improvements upon the base Gradient Boosting Machines framework that XGBoost advances its system optimization and algorithmic enhancements.

[⋆] Partipating Project of Winter 2022 University of Washington Department of Statistics Directed Reading Program (DRP)

Cover The Cover metric is the contribution of each feature to the number of observations summed up from each tree expressed in percentage, it is calculated as the second order gradient of training data classified to the leaf.

Gain Gain is a parameter that corresponds to the importance of the node in the model, for split nodes, the gain is the information gain metric of a split node.

In mathematical sense, the total Gain in XGB is expressed as

$$\text{Gain} = \frac{1}{2}[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}] - \gamma \qquad (1)$$

Note that from equation 1, G_L stands for Gini Impurity score from the left branch of the DT, while G_R quantifies the incorrect classification at a split for the total number of classes. , As a corollary, H takes into account of the entropy from the left and right branch.

Hence to express the variable importance in XGB more vividly, we generated the following plot with **R** package *h2o*. Each feature is converted to an one-hot encoder for each instance of the website we are analyzing, the normalized value for each feature across all samples are plotted in a summarized fashion based on their distribution on the normalized value scale, with blue indicating values closer to 0.00 and red indicating a value that is closer to 1.00.

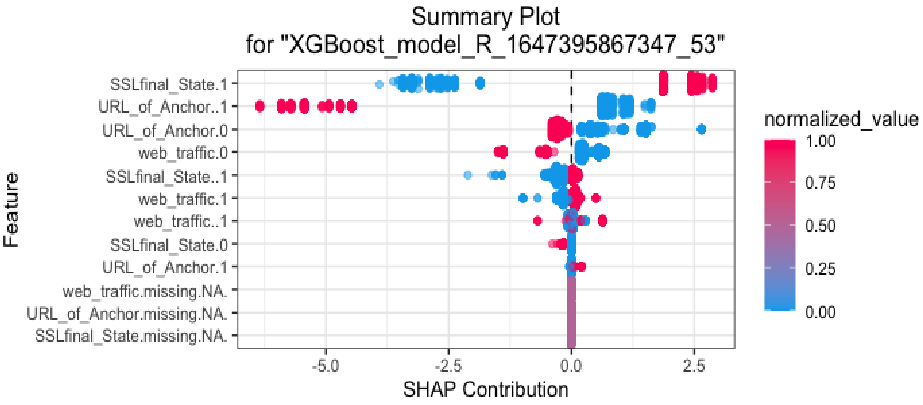

Fig. 1.

2.2 Random Forest

Mean Decrease Gini We start by investigating the model metrics from Gini Impurity, a metric used in Decision Trees to determine how to split data into smaller sizes of groups, specifically, using which variable, and at what threshold.

$$\begin{aligned}\text{Gini impurity} &= 1 - \sum_{i=1}^{K} p_i^2 \\ &= 1 - \text{Gini Index}\end{aligned} \tag{2}$$

where K is the number of class labels, p_i is the proportion of the i^{th} class label

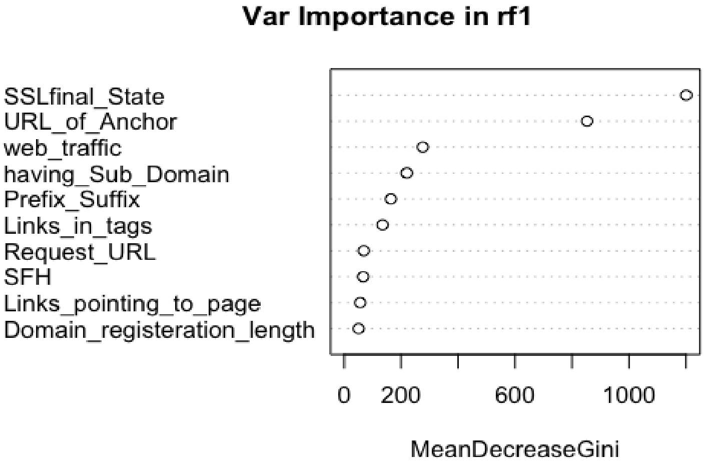

Fig. 2.

Gini Impurity from equation 2 measures how often a randomly chosen record from the data set used to train the model will be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset (e.g., if half of the records in a group are ”1” and the other half of the records are ”0”, a record randomly labeled based on the composition of that group has a 50% chance of being labeled incorrectly). Gini Impurity reaches zero when all records in a group fall into a single category, this makes intuitive sense as when there exists only one label, the probability of classification falls into the category ends up being 100%.

To recap, the Gini Impurity measures in the project is essentially the probability of a website being incorrectly classified with a Decision Tree based on the training data. What shows in Fig 2 is the corresponding Mean Decrease Gini, which standardizes and normalizes based on the sample mean and standard deviation in contrast to a decrease Gini score. In terms of its association, a higher value of Mean Decrease Gini score implies the amount such variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. As we can straightforwardly observe from the plot, SSLfinal_state, URL_of_Anchor and web_traffic are the three most important predictors in determining if a website should be deemed as fraudulent or not. This is incoherence of the variable importance we have received from using a XGBoost model from the prior section. Such result reflect the system design of the models, which in part adopts the central idea and structure composed by [Xiang et al.]

2.3 Support Vector Machine

As for the supervised, linear model used in this project, the idea and mechanism behind a SVM is simple - given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other. We have continously implemented with the *h2o* package for this modelling's purpose of classifying websites with binary outcomes.

Similar to the purpose of any classifier, SVM performs well in tackling classifying non-linear examples, in the context of high-dimensional space (i.e. with large-scale predictors), in addition, it does not suffer from multi-collinearity problem. It is worth bearing in mind, however, that paralleling support vector machine costs considerable amount of time to train, the yielded result does not directly translate to probability estimations. If you are willing to accept the disadvantages mentioned above, SVM would be a recommended model for picking linear kernel that has similar logics as the logistic regressor.

3 Random Forest Error Analysis and Conclusion

After performing cross-modality comparison, we found RF generating the most promising results for phishing website classification. Based on the test statistics listed below, we can clearly observe that random forest model in *Caret* yielded an accuracy around 98% on the large model for test data, for the smaller model containing only the five most important features contributing in the model, this metric maintained an accuracy score around 93%. Shall more features be included in the future, as noticed and proposed by [1] , restrictions encountered when using the random forest model, such as the number of available features, would have alleviated effects for model performance. The decision of features selected followed similar strategies as Subasi et al. [2]

Table 1. Confusion Matrix on Random Forest Test Data using RF$_1$

Reference Prediction	-1	1
-1	3302	1
1	83	4238

Table 2. Other Statistics in RF_1 Test Data

Accuracy	0.9811
95% CI	(0.9778, 0.9841)
No Information Rate	0.5595
P-Value [Acc > NIR]	$< 2e - 16$
Kappa	0.9617
Mcnemar's Test P-Value	0.09673
Sensitivity	0.9755
Specificity	0.9856
Pos Pred Value	0.9816
Neg Pred Value	0.9808
Prevalence	0.4405
Detection Rate	0.4297
Detection Prevalence	0.4377
Balanced Accuracy	0.9805
'Positive' Class	-1

Table 3. Confusion Matrix on Random Forest Test Data using RF$_2$

Reference Prediction	-1	1
-1	1413	154
1	100	1703

Table 4. Other Statistics in RF_2 Test Data

Accuracy	0.9246
95% CI	(0.9152, 0.9333)
No Information Rate	0.551
P-Value [Acc > NIR]	$< 2.2e - 16$
Kappa	0.8482
Mcnemar’s Test P-Value	0.0008826
Sensitivity	0.9339
Specificity	0.9171
Pos Pred Value	0.9017
Neg Pred Value	0.9445
Prevalence	0.4490
Detection Rate	0.4193
Detection Prevalence	0.4650
Balanced Accuracy	0.9255
’Positive’ Class	-1

Bibliography

[1] Hutchinson, S., Zhang, Z., and Liu, Q. (2018). *Detecting Phishing Websites with Random Forest: Third International Conference, MLICOM 2018, Hangzhou, China, July 6-8, 2018, Proceedings*, pages 470–479.

[2] Subasi, A., Molah, E., Almkallawi, F., and Chaudhery, T. J. (2017). Intelligent phishing website detection using random forest classifier. pages 1–5.

[Xiang et al.] Xiang, G., Hong, J., Rose, C. P., and Cranor, L. Cantina+: A feature-rich machine learning framework for detecting phishing web sites. *ACM Transactions on Information and System Security (TISSEC)*.