Machine Learning Applications for Identifying Phishing and Scam Websites

Jonathon Meney
School of Mathematical & Computational Sciences
University of Prince Edward Island
Charlottetown, Canada
jonmeney@gmail.com

***Abstract*—Phishing and other scam websites are becoming increasingly common on the Internet. The research aims to determine whether certain features of a webpage are sufficient to accurately classify a website as either phishing or legitimate. This involves analyzing the dataset to identify patterns and correlations between the features and the target labels, as well as evaluating the performance of machine learning models trained on this dataset. The K-Nearest Neighbors (KNN) classifier achieved an accuracy of 95% and the Decision Tree classifier 93%. Both models exhibited high accuracy, precision, and recall, indicating their effectiveness in distinguishing between phishing and legitimate websites. The high accuracy of both models suggests that the selected features are indeed effective for distinguishing scam sites and legitimate sites. This finding is significant for the development of automated phished detection systems, which can be integrated into web browsers or email clients to provide real-time protection for users.**

***Index Terms*—Machine learning, classification, phishing websites, scam websites**

I. Introduction

A. Motivation for Choosing the Dataset

According to PhishTank.org, over 11 million phishing sites have been submitted to there repository. It has become increasingly common to find phishing and other scam websites on the Internet. In addition they have been becoming more and more difficult to detect and avoid. Phishing involves the creation of fraudulent websites designed to deceive users into divulging sensitive information, such as login credentials, financial details, or personal data. These attacks not only compromise individual privacy, but also pose significant risks to organizations, leading to financial losses, damaged reputation, and/or data breaches. As phishing techniques become increasingly sophisticated, the need for robust, automated methods to detect and mitigate such threats has become critical.

The data set chosen for this research consists of 50,000 websites, each broken down into 43 features. These features capture various aspects of the structure and content of a website, as well as various features of the site URL. Using this dataset, the research aims to contribute to the development of effective tools for combating phishing attacks, ultimately enhancing online security for both individuals and organizations.

B. Goals of the Research

The primary goal of this research is to determine whether the provided features of a webpage are sufficient to accurately classify a website as either phishing or legitimate. This involves analyzing the dataset to identify patterns and correlations between the features and the target labels, as well as evaluating the performance of machine learning models trained on this data. Specifically, the research seeks to answer two key questions: (1) Can the given features reliably distinguish between phishing and legitimate websites? and (2) Is the dataset comprehensive enough to enable confident decision-making in real-world scenarios?

To achieve these goals, the research will explore a variety of machine learning algorithms, ranging from traditional classifiers like logistic regression and decision trees to more advanced techniques. The performance of these models will be assessed using metrics such as accuracy and precision, ensuring a thorough evaluation of their ability to detect phishing websites while minimizing false positives. Additionally, feature importance analysis will be conducted to identify which attributes contribute most significantly to the classification task, providing insights into the underlying characteristics of phishing websites. By addressing these objectives, the research aims to advance the field of URL classification and contribute to the development of more effective phishing detection systems.

II. Background

A. Description of the Dataset

The dataset used consists of 50,000 websites, and is evenly split between phishing and legitimate categories, having 25,000 of each. Each website is represented by 43 features extracted through web scraping, using a Python library called Beautiful Soup. These features capture a wide range of structural and content-related attributes of a webpage, such as the presence of titles, input fields, buttons, images, links, and other HTML elements. Additionally, the dataset includes quantitative measures, such as the number of inputs, buttons, images, and paragraphs, as well as the length of the title and text. The features are designed to reflect the characteristics commonly associated with phishing websites, such as the excessive use of input fields to harvest user data or the absence of typical webpage elements like headers and footers.

The dataset is labeled, with phishing websites assigned a label of 1 and legitimate websites assigned a label of 0, making it suitable for supervised learning tasks. This comprehensive

and balanced dataset provides a robust foundation for training and evaluating machine learning models to classify URLs as either phishing or legitimate.

B. Machine Learning Techniques Used

To analyze the dataset and achieve the research goals, three primary machine learning techniques were employed: feature importance analysis, K-Nearest Neighbors (KNN) classifier, and Decision Tree classifier.

1) Feature Importance Analysis: Before training the models, feature importance analysis was conducted to identify which of the 43 features contribute most significantly to the classification task. As a result several of the features were found to have little to no significance and were removed when training different models.

2) K-Nearest Neighbors (KNN) Classifier: KNN is a simple yet effective algorithm for classification tasks. This model was chosen to begin initial research into the problem. Parameter tuning was performed when using KNN as well to find the most optimal K. The performance was also analyzed by examining the learning curve of the trained model. Additionally, the confusion matrix was examined to determine the accuracy precision and recall of the model.

3) Decision Tree Classifier: The next technique that was used was using a Decision Tree Classifier. Given the number of features in the dataset it seemed most useful to utilize the Decision Tree as information gain is its driving decision property. Similarly to KNN as before, parameter tuning was performed using a grid search cross validation to find the best parameters for the Decision Tree. The Decision Tree also performed slightly worse than the KNN classifier, which was surprising. The same specifics were examined as KNN (learning curve, and confusion matrix).

By combining these techniques, the research aims to build a robust framework for URL classification, using the strengths of each method to achieve accurate results. The insights gained from feature importance analysis, coupled with the predictive power of KNN and decision trees, provide a comprehensive approach to addressing the challenge of phishing detection.

III. Results & Analysis

A. Feature Analysis

After performing a correlation analysis it was determined that several of the features did not have much effect on the target prediction. The following features were removed from the training data as a result: has_title, has_link, length_of_title, has_h2, has_h3, has_footer, and has_nav.

B. K-Nearest Neighbors (KNN) Classifier

1) Parameter Tuning: When constructing the K-Nearest Neighbors model to be used on this dataset, the best value of K was determined via defining a wide K range. The range chosen to determine the most optimal K was 1 to 10. Given the size of the dataset and the robustness of the KNN model this value also took a significant amount of time to generate. After finding the accuracy scores for K from 1 to 10 the following graph was generated:

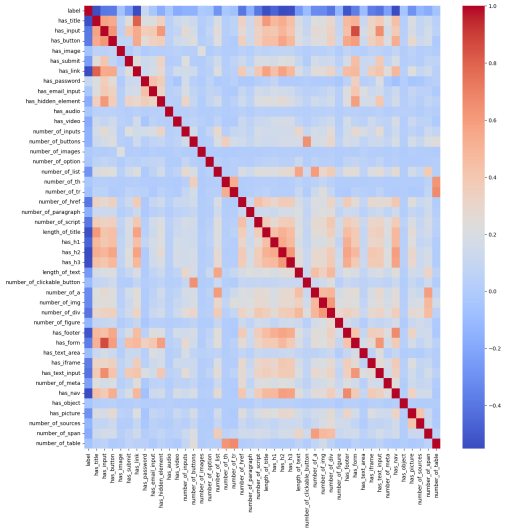

Fig. 1. Heatmap Correlation.

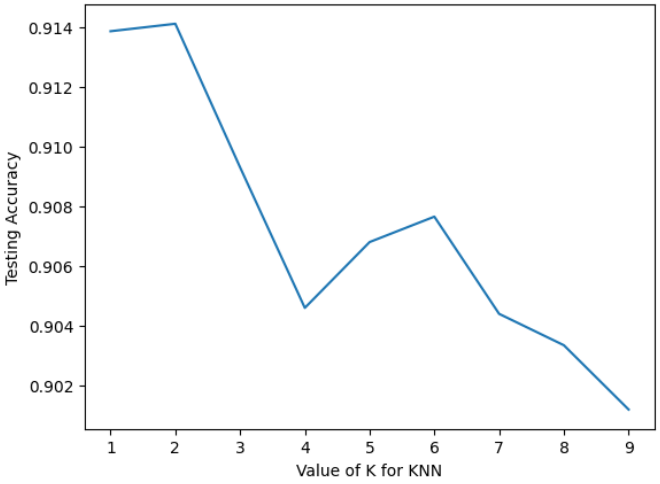

Fig. 2. K Parameter Tuning.

After analyzing the above graph for K from 1 to 10, it was determined that the most optimal value for K was 3. Although a K of 2 provided the highest accuracy, a value of 3 for K seemed more appropriate and close to the elbow of the curve. If we were to choose a K of 4, which is less accurate but also a potential elbow, we would be grouping too many neighbors during training and predictions, and overtime we would see a more significant loss in accuracy.

2) Performance: Overall, the performance of the KNN classifier was very high. Below is the confusion matrix values for the trained model with K equal to 3:

As we can see, the model performed with an accuracy of 95% which is quite high. As well the precision and recall were on average 95%. This signifies the model overall will be accurate in identifying and correctly classifying true positives,

TABLE I
KNN PERFORMANCE

Label	Measure		
	Precision	**Recall**	**F1-Score**
Legitimate	0.95	0.95	0.95
Phishing	0.95	0.94	0.95
	Overall Accuracy		0.95

and return a low rate of incorrect predictions.

3) Learning Curve: Although the performance statistics are all favorable, when examining the learning curves, we can see they are beginning to converge slightly. However, there is a large gap between the training and validation accuracies, which means we are overfitting our model to the dataset only.

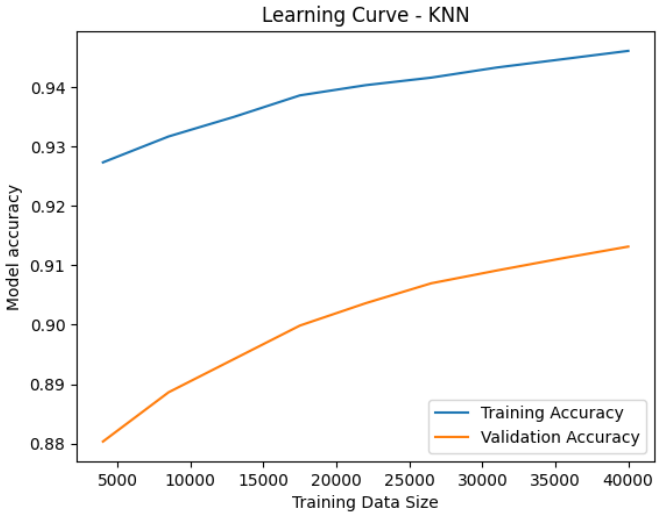

Fig. 3. KNN Learning Curve

As a result of this overfitting, we may see a significant dip in new data entering the model to be predicted and classified. In order to combat this, we need to increase the size of the dataset to increase the variance within the data. Increasing the size of the dataset would improve the learning curve of the model and begin to converge. In addition, cross validation was not performed during the training of this model unlike the others in this report.

4) Confusion Matrix: The confusion matrix presented is for the KNN model with K equal to 3. The matrix shows the distribution of predicted labels versus the actual labels. The first row indicates that the model correctly predicted 9,530 instances as legitimate and 517 were incorrectly classified as phishing. The second row shows that the model correctly predicted 9,400 instances as phishing, and 553 were incorrectly classified as legitimate. This indicates that the model again has a high accuracy and is well trained.

We can see, the model does well at distinguishing between legitimate and phishing instances. The high number of correct predictions demonstrates strong accuracy. However, the model does make some misclassifications. Overall, the model is well

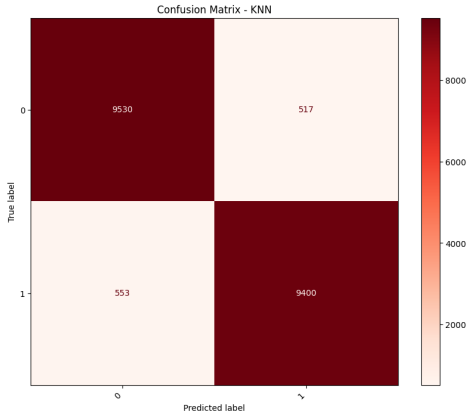

Fig. 4. KNN Confusion Matrix

trained, making it a reliable tool for distinguishing between legitimate and phishing sites, though further tuning could enhance its performance.

C. Decision Tree Classifier

1) Initial Learning Curve: The learning curve of the initially trained decision tree shows the same gap that we had prior on KNN. As the training data size increases, we see the accuracy improve, indicating that the model is learning effectively from the additional data. Additionally, we can see that overall we are predicting with a much higher accuracy than KNN. However, a large gap still exists between each curve, and the model is slightly overfitting the data.

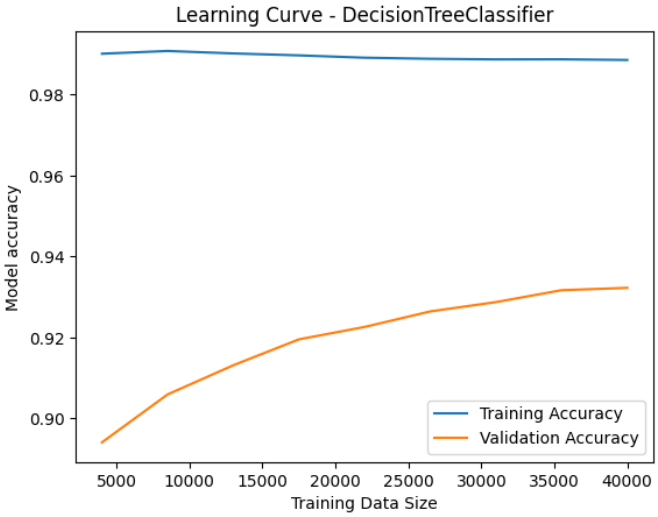

Fig. 5. Initial Decsion Tree Learning Curve

The gap between the training and validation accuracy is only a few percent, indicating that the model is not overfitting significantly. The learning curve suggests that the Decision Tree is achieving a good balance between bias and variance. As more data is added, the validation accuracy continues to improve, which is a good sign that the model's performance

can be further enhanced with more training samples. In general, the learning curve indicates that the model is able to predict accurately on new data.

2) Parameter Tuning: To find the most optimal parameters for a Decision Tree model, a grid search cross validation was used. The parameter grid that was used included searching over gini and entropy criterions, and a minimum sample split from 2 to 50. The grid search was then performed with a cross validation of 10 and based on an accuracy scoring.

After fitting the model using grid search it was determined that the best parameters for the decision tree was a minimum sample split of 3, and entropy as the learning criterion. The model was also trained with a random state of 4. The final model was then determined to have a 93% accuracy.

3) Performance: Overall, the performance of the Decision Tree classifier was very high. Below is the confusion matrix values for the trained model from the grid search:

TABLE II
DECISION TREE PERFORMANCE

Label	Measure		
	Precision	**Recall**	**F1-Score**
Legitimate	0.94	0.91	0.92
Phishing	0.91	0.95	0.93
	Overall Accuracy		0.93

As we can see, the model performed with an accuracy of 93% which is quite high. The precision and recall varied slightly as compared to the KNN classifier, but were both still very high. This signifies the model overall will be accurate in identifying and correctly classifying true positives, and return a low rate of incorrect predictions.

4) Learning Curve: Although the performance statistics of this tuned model from grid search are all favorable, when examining the learning curves, there is a still a large gap between the training and validation accuracies, which means we are still somewhat overfitting our model.

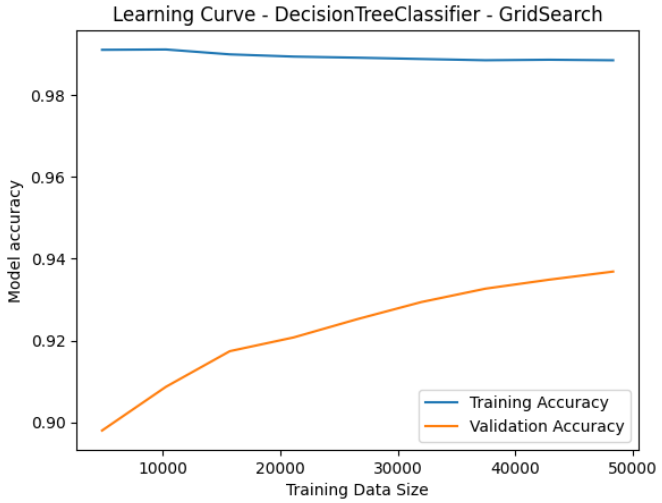

Fig. 6. Grid Search Decision Tree Learning Curve

However, compared to the initial learning curve from the basic unrefined decision tree to this grid search decision tree, we can see the accuracies converging at a slightly faster rate. However, because of this overfitting, once again new data may be incorrectly predicted and classified as the model is fitted to the dataset slightly too well. In order to combat this, we again would still need to increase the size of the dataset. Increasing the size of the dataset would improve the learning curve of the model and begin to converge.

5) Confusion Matrix: The confusion matrix below is for the grid searched decision tree. The matrix shows the distribution of predicted labels versus the actual labels. The first row indicates that the model correctly predicted 5,604 instances as legitimate and 567 were incorrectly classified as phishing. The second row shows that the model correctly predicted 5,726 instances as phishing, and 428 were incorrectly classified as legitimate. This indicates that the model again has a high accuracy and is well trained.

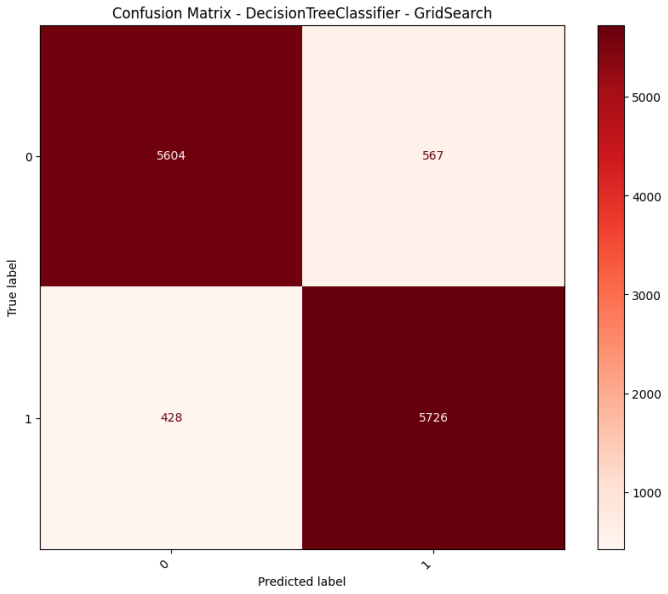

Fig. 7. Decision Tree Confusion Matrix

We can see, the decision tree also performs well at distinguishing between legitimate and phishing instances. The high number of correct predictions demonstrates strong accuracy. However, the model does make some misclassifications. Overall, the model is well trained, making it a reliable tool for distinguishing between legitimate and phishing sites, though increasing the size of the dataset and therefore reducing the overfitting would likely provide a strong improvement.

IV. CONCLUSIONS & DISCUSSION

This study explored the application of machine learning techniques, specifically K-Nearest Neighbors (KNN) and Decision Tree classifiers, for the identification of phishing and scam websites. The research aimed to determine whether the provided features of a webpage are sufficient to accurately classify a website as either phishing or legitimate. Both models

demonstrated strong performance, with the KNN classifier achieving an accuracy of 95% and the Decision Tree classifier achieving an accuracy of 93%. While the KNN model outperformed the Decision Tree in terms of accuracy, the Decision Tree provided valuable insights into feature importance through information gain, making it a more insightful model for this dataset.

A. Key findings

Both models exhibited high accuracy, precision, and recall, indicating their effectiveness in distinguishing between phishing and legitimate websites. However, the learning curves revealed signs of overfitting, particularly in the KNN model, suggesting that increasing the dataset size could further improve model generalization.

Feature analysis identified several attributes with little to no significance, which were subsequently removed from the training process. This simplified the models and improved their efficiency without compromising performance.

The Decision Tree classifier, while slightly less accurate, offered better insight and faster convergence in its learning curve, making it a more suitable choice for this dataset.

B. Discussion

The current dataset, while balanced, may not fully capture the variability of the features of phishing. Expanding the dataset to include more examples of evolving phishing strategies would improve model performance.

Although the dataset includes 43 features, additional attributes such as SSL certificate details, domain age, and traffic patterns could further enhance classification accuracy. Additionally, we could look at the URL of its site to gain more features.

C. Future Research

The high accuracy of both models suggests that the selected features are indeed effective in distinguishing phishing websites from legitimate ones. This finding is significant for the development of automated phishing detection systems, which can be integrated into web browsers or email clients to provide real-time protection for users. However, the observed overfitting in both models underscores the importance of using larger and more diverse datasets to enhance model generalization and robustness.

Future work could explore more sophisticated machine learning techniques, such as ensemble methods such as Random Forests, Gradient Boosting or deep learning models, to achieve even higher accuracy and robustness.

In conclusion, this study demonstrates the effectiveness of machine learning in identifying phishing and scam websites. While both KNN and Decision Tree classifiers performed well, the Decision Tree's insight and faster learning curve convergence make it a more practical choice for this application. Future research should focus on expanding the dataset, exploring advanced models, and developing real-time detection systems to further enhance online security.

REFERENCES

[1] https://www.kaggle.com/datasets/yuvistrange/content-based-features-phishing-and-legit-websites

[2] https://phishtank.org/stats.php