National College of Ireland

Comparative Analysis of Supervised Machine Learning Models for Phishing Detection

MSc Research Project
Cyber Security

Mamta Sawant
Student ID: X19221134

School of Computing
National College of Ireland

Supervisor: Dr Vanessa Ayala-Rivera

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Mamta Sanjay Sawant
Student ID:	X19221134
Programme:	MSc in Cyber Security **Year:** 2021-22
Module:	Research Project
Supervisor:	Dr Vanessa Ayala-Rivera
Submission Due Date:	16/12/2021
Project Title:	Comparative Analysis of Supervised Machine Learning Models for Phishing Detection
Word Count:	5373 **Page Count** 17

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Mamta Sanjay Sawant

Date: 16/12/2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Comparative Analysis of Supervised Machine Learning Models for Phishing Detection

Mamta Sawant
19221134

Abstract

Phishing is an illegitimate way of extracting information from the target computer system. The massive use of online services motivated the higher number of online fraudulent activities. Phishing is a technique that is used by individuals to do fraudulent activities, and this is considered one of the most dangerous cyber attacking techniques. Machine learning (ML) techniques are frequently used to solve real-world problems related to classification, detection, and regression. This report focused on a comparative analysis of the machine learning models to detect Phishing websites. Three major supervised machine learning algorithms such as KNN, Decision Tree, and Logistic Regression are used to build an ML model for detection of the Phishing website and their results are compared. The dataset is collected from the Mendeley Data site, and it comprises a total of 88647 observations. The result of this study shows that the highest accuracy is recorded for the Logistic Regression and the lowest accuracy is obtained for the KNN model with respect to parameter metrics such as accuracy and time taken to train the model.

1 Introduction

Phishing is an online attack which is done with an aim to steal the sensitive information from the victim users and organisations. The massive use of online services motivated the higher number of online fraudulent activities (Shirazi et al., 2018). The attackers often try to place a malicious website as a legitimate one. Phishing is a technique that is used by individuals to do fraudulent activities, and this is considered one of the most dangerous cyber attacking techniques. "Cyber Awareness" is the only way to prevent phishing cyber-attacks. Some of the frequently used phishing attacks are URL phishing, clone phishing, algorithmic phishing, and content injection phishing. Phishing detection techniques such as pattern matching filters, white-list filters, and Blacklist filters are extensively used to prevent these kinds of potential phishing attacks(Yi et al., 2018).

In order to detect the phishing, analysing the features of the website is important. Machine learning approaches can be used for analysing the data. Machine learning (ML) is the branch of computer science that entails on use of algorithms and data to imitate the human learning process.

This problem of phishing detection leads us to following research question: Do the supervised machine learning algorithms; Decision Tree, K-Nearest Neighbor(K-NN), Logistic Regression; show a significant difference in detection accuracy and speed for phishing attacks?

To address the research question this study is conducted to do the comparative analysis of the ML models used for the identification of phishing websites. In this report, a massive dataset containing the information related to the phishing attacks is analysed using the various machine learning techniques and a comparative analysis of employed machine learning techniques has also been done to select the highly accurate model. Decision tree, logistic regression, and K-NN algorithms are used for the classification of phishing URLs. These three machine learning techniques are the supervised machine learning models and hence the dataset is divided into 70:30 to create a training and testing dataset. I decided to used the 70-30 ratio as the training data could give a better classification model and while testing there would be more data available in 70-30 ratio which can help to give better error efficiency rate. The training dataset is used to train the machine learning parameters whereas the testing dataset is used for validation of the ML model. The dataset used in this analysis is collected from the cloud-based repository and is found under the name "dataset_full.csv"(Vrbančič, 2020). In this research the performance measurement of all three models is evaluated and compared with metrics such as recall, precision, accuracy, f1 measure and time taken to train the model.

The following areas will be covered in the rest of the report: Section 2 will explore similar studies done by other researchers in the past and compare and contrast their thoughts. Section 3 will go over the methodology and approach used to create the models required to do the comparative analysis. Section 4 will include the model's design specifications such as information about the dataset used for analysis and different machine learning algorithms. Section 5 will be about implementing the proposed system and Section 6 will be about evaluating the result of the model. Finally, section 7 will bring the research to an end with a final conclusion and the possibility of future work.

2 Literature Review

Phishing is a form of an attack in which sensitive information is stolen using social engineering and website forging tactics to mislead people and obtain personal information with financial worth. This section will examine previous research on the detection of phishing domain name using supervised machine learning approaches.

2.1 Related Work

Phishing detection has become challenging over time because to the validity that fraudulent websites display from the victim characteristics such as content in the body, DNS spoofing, and so on. Multiple significant studies are conducted in order to comprehend the feature selection in the dataset and the significance of the features in order to classify whether or not the website is malicious. As per the article (Dutta, 2021), advancements in internet and cloud technologies are resultant in a noteworthy increase in electronic trading which enforces customers to make some digital purchases as well as transactions. This success provides an opportunity for unauthorized access of users' sensitive data and damaging enterprise resources. Thus, phishing is one of the aware attacks which trick customers to access malicious content

for obtaining their data. As a result, IT experts are coming up with various techniques for detecting phishing websites like the blacklist, heuristic, and various supervised learning models.

Author (Ali et al., 2019) used entropy-based feature selection algorithms for classification of detecting phishing websites in their research. Their strategy included the use of URL-based characteristics such as URL length, sub-domain name, DNS features such as DNS records, domain name age, page ranking, and web-based elements such as iframes, and pop-up windows. The dataset comprising these features was subjected to three feature selection methodologies: wrapper feature selection (WFS), correlation-based feature selection (CFS), and entropy-based feature selection (ENFS).

Currently, hackers are installing malicious software on computers for misusing the credentials, frequently using the systems for intercepting the personal name of customers and passwords of digital accounts. Some of the methods used by phisher attackers are email, uniform resource locators, instant messages, forum postings, telephone calls, and text messages for misusing user data. Authors (Herland et al., 2019) depicts that supervised learning is one of the appropriate processes of offering input information and correct output information to the machine learning model. Thus, it is brought into sight that supervised learning models are the most suitable technique for assessing risk, classifying the image, fraud detection, filtering spam henceforth in the real world. It incorporates, logistic regression, decision tree, random forest, Ada Boost, SVM, KNN, neural networks, gradient, boosting and XGBoost.

In the study by (Odeh et al., 2021) implemented different phishing detection strategies such as user education, search engine-based techniques, supervised machine learning and deep learning techniques and their drawbacks. In this study Adaptive boosting classifier algorithm is used to for discrete dataset. The result of their works showed that the PhiBoost model performs best using the Logistic regression machine algorithm with the accuracy 98.40%**.** Authors (Marchal et al., 2014) examined the URLs of the websites and analysed the URL attributes that were extracted. Based on several inquiries using Google and Yahoo search engines, the writers discovered and discussed the URL-related elements for each website. The attributes gathered are then used in a machine learning classification system to identify phishing websites in a real-world dataset.

Machine learning tools are detecting phishing websites on the basis of markup visualization. It means, machine learning models are trained on the basis of website code visual representation which aids in enhancing the accuracy as well as speed of sensing phishing sites. Authors (Olegario et al., 2020) implemented the application of decision trees for disease identification in Japanese oak and pine trees using satellite imagery in this research. The results showed an average accuracy of 97.82 percent, which the researcher says was achieved by properly partitioning the information into well-ratioed training and test sets. K. Ohta, did a similar study in which the user used kNN to diagnose wilting and sick pine trees and achieved an average accuracy of 72 percent. It is believe that decision tree outperformed kNN because of its capacity to cope with missing data, as the existence of missing data in the dataset had no effect on prediction accuracy.

The authors (Vanhoenshoven et al., 2016) analyzed several algorithms such as MLP, Naive Bayes, SVM, Decision Trees, RF, and kNN using a very big dataset of 2.4

million URLs and 3.2 million attributes. These classifiers accuracy, precision, and recall were utilized as assessment measures. Random Forest achieved the maximum precision and recall, with 97.69% and 97.28% accuracy respectively. While the differences were minor, pairwise statistical analysis indicated that RF surpasses other approaches substantially.

A study conducted by (Alanezi et al., 2020) employed the use of machine learning to build a model to detect credit card frauds. Out of the different algorithms used by the researchers, logistic regression produced results of the highest accuracy of 94.6%. Although the author has stated that this accuracy differs through different datasets, it states that this is due to the algorithms ability to classify missing data efficiently.

In paper Detecting Phishing Websites Using Machine Learning determines phishing as a kind of cyber-crime where spammed texts and false websites appeal exploited people to provide sensitive information to phishers(Alswailem et al., 2019). Therefore, various techniques are used by specialist for detecting the phishing in order to control it as soon as possible. In this article, author believes that decision tree is highly used for data mining.

In a separate study conducted by ("A Comparison of Naive Bayes Methods, Logistic Regression and KNN for Predicting Healing of Covid-19 Patients in Indonesia | IEEE Conference Publication | IEEE Xplore," 2021) to predict the cure rate for the SARS-COV-2 virus, a similar approach was used where the author employed the use of kNN, logistic regression and Naive Bayes machine learning techniques. The study showed that the kNN technique was able to generate the highest accuracy of 75%. The author commented that the data for this research was limited and hence, they weren't able to achieve a higher accuracy. This is true because kNN is largely dependent on the quality of the dataset for providing better results.

In paper ("[2009.11116] Phishing Detection Using Machine Learning Techniques," 2020) explains the use of random forest supervised technique in phishing detection which consist of various individual decision tress that perform as a bunch to decide the outcome. Every tree of random forest stipulates the projection class, and the outcome is considered as a most assumed class amongst the decision trees. On the other hand, Ada-boost is somehow similar to Random Forest which make weak classification to create a powerful classifier. A single model might poorly segment the objects but if experts will integrate the multiple classifiers by choosing a bunch of samples in all iteration and allocate adequate weight to last vote then it will be considered as a suitable for entire classification.

Authors (Alanezi et al., 2020), used the K-NN (K-Nearest Neighbor) method for URL Identification model for phishing attack detection. The dataset in the study contains 1353 observations and 10 features that describe each observation. In this model, the performance measure is determined using the accuracy metric, and it was discovered that the value of k is critical for achieving higher model accuracy. The model was tested with several values of K, and the highest accuracy was obtained with k=10.

According to internet research, machine learning is commonly employed for phishing detection. The study conducted by (Alanezi et al., 2020) (Olegario et al., 2020) (Marchal et al., 2014) serves as the foundation for the research question addressed in this paper. This study intends to implement and compare supervised machine learning models such as KNN, Logistic Regression, and Decision Tree.

3 Research Methodology

The methodologies and analysis presented in this section were chosen after a thorough examination of the work done by researchers in the previous section. The study's goal is to conduct a comparison of machine learning algorithms such as Decision Tree, K-NN, and Logistic Regression in terms of several performance measures such as accuracy, precision, recall, f1 measure, and time necessary to train the model.
The section mainly focuses on feature selection and different machine learning algorithms that are used for phishing detection. Figure 1 illustrates the overview of the analysis and consolidates the steps involved in this process:

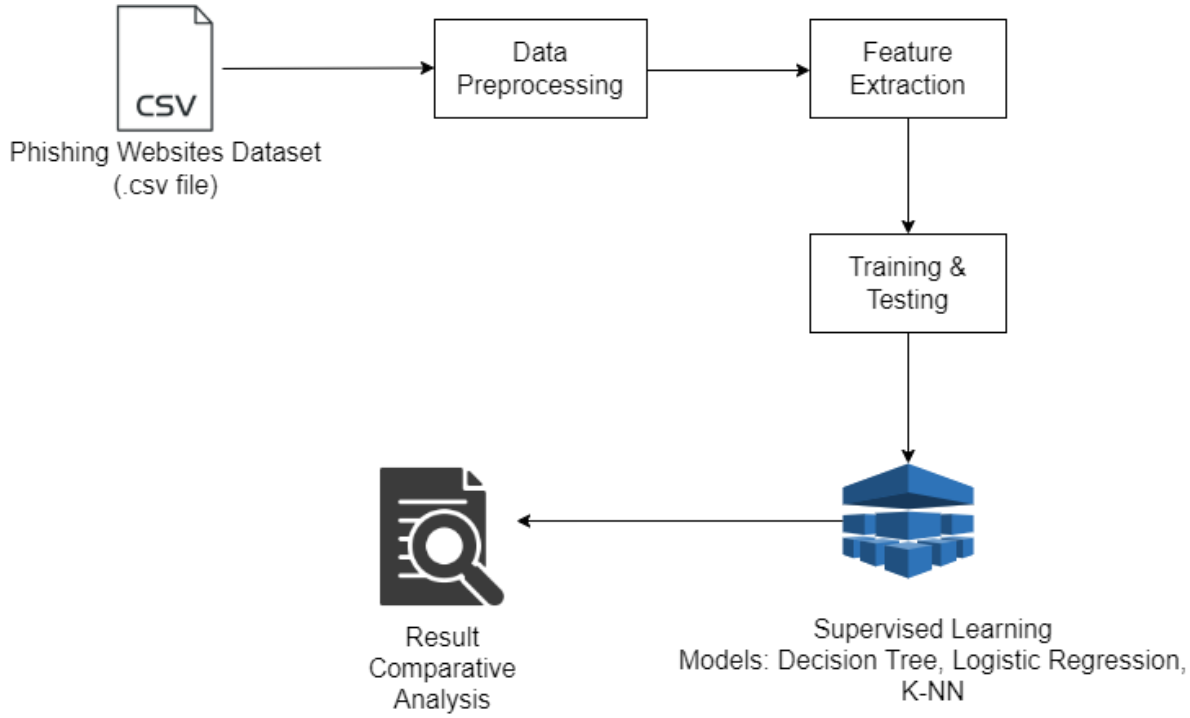

Figure 1: Research Methodology Work-flow diagram

The following subsections focuses on in detailed procedure involved in this research.

3.1 Data Understanding

The data is the primary source of any kind of analysis. For the comparative analysis of the machine learning models, the phishing dataset is collected from a cloud-based repository i.e. Mendeley Data. The dataset is present in .csv format with a total number of observations equal to 88647. The dataset comprises the data of phishing as well as legitimate websites. The total number of features in the 'Phishing Websites Dataset' is 111. In the following dataset the number of 'legitimate' instances of website are labelled as '0' whereas 'phishing' instances are labelled as '1'. Based on the major characteristics of the dataset, the website is labelled as a legitimate or phishing website. The complete dataset is treated as input to the machine learning model. The target observation in the "Phishing" column is the "1"(Vrbančič, 2020).

A pie chart is plotted in Figure 2 to analyse and illustrate the dataset in numerical proportion. It is observed that the percentage share of legitimate sites is 65.4 %, while the percentage share of phishing websites is 34.6%.

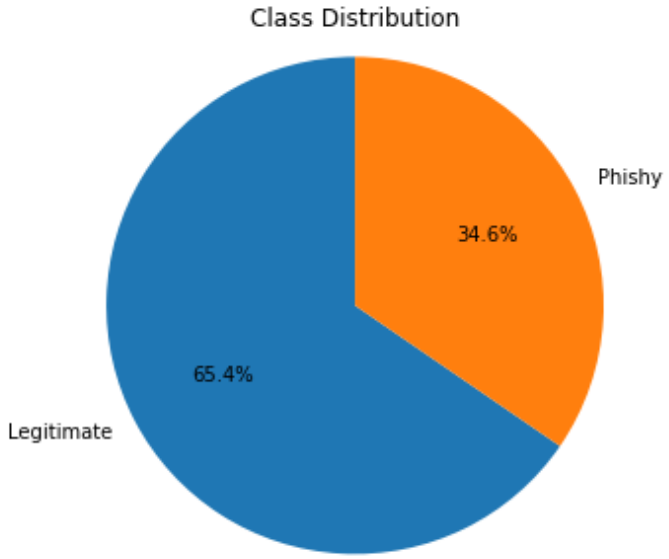

Figure 2: Pie Chart for the Percentage Distribution of Dataset

3.2 Data Pre-processing

The dataset that is collected using numerous resources is generally diagnosed with numerous a mbiguities such as missing values or duplication of the data. The model is implemented using the total number of rows in the dataset 88,647 in which count of legitimate Websites is 58000 and Phishy Websites is 30647.

To train the model without any noisy or unwanted data it is necessary to check the missing values null, values or NaN (Not a Number). So, to check if there were any such values in the dataset python code was implemented to with a dropna() method which removes the NA values but incase of this dataset no null or NaN values were identified (“Handling Missing Data | Python Data Science Handbook,” 2021).

3.3 Feature Extraction

The total number of in the 'Phishing Websites Dataset' was 111. The Correlation Coefficient feature selection technique was utilized to identify the important features that may be used for training and testing. The coefficient test was done using a python code to identify the linear relationship between the 2 or more variables. The idea behind the correlation for feature selection is that the good variables are highly correlated with the target (“What Is the Correlation Coefficient?,” 2021). Figure 3 shows the correlation graph which shows the best features that are related to each other.

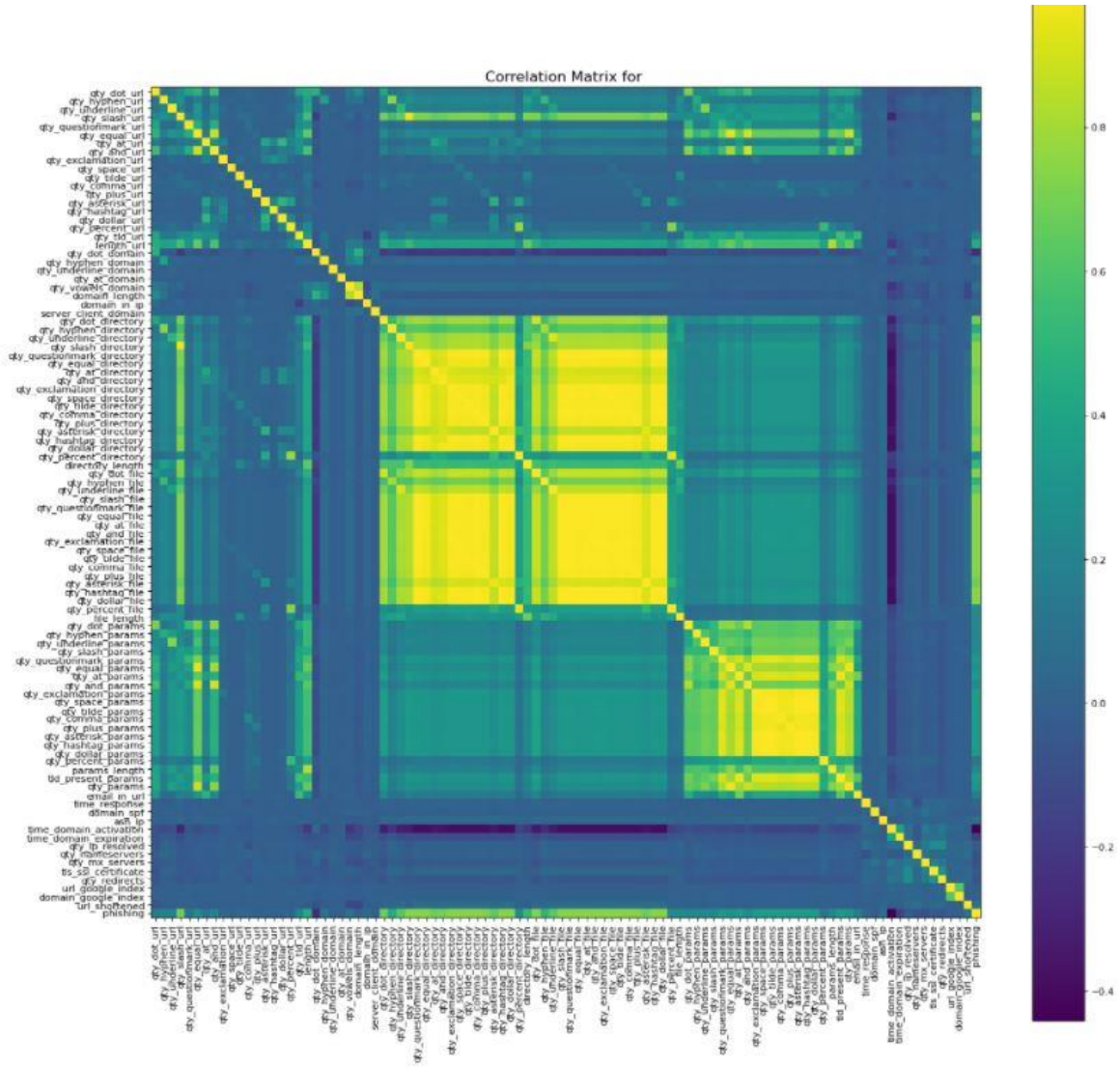

Figure 3: Correlation Matrix

In Figure 3, it shows the positive and negative correlations between the variables, allowing us to better understand their relationship. Because the number of variables included in the correlation study was large, a separate function in Python was written to export the features that had a strong association with the target variable 'Phishing.' The threshold value for filtering these features is set to 0.99. Following the execution of the test, 21 features were extracted, using which the dataset files was created which had 88647 values. The description is extracted features for eg, qty_exclamation_file is the number of "!" in the URL, qty_at_file is the number of "@"signs ("Datasets for phishing websites detection - ScienceDirect," 2020). Table 1 states the features extracted using correlation analysis:

Features Names	Data Type
qty_and_file	Int64
qty_asterisk_params	Int64
qty_at_file	Int64
qty_comma_directory	Int64
qty_comma_file	Int64

qty_dollar_file	Int64
qty_comma_params	Int64
qty_dollar_params	Int64
qty_equal_file	Int64
qty_exclamation_directory	Int64
qty_exclamation_file	Int64
qty_hashtag_directory	Int64
qty_hashtag_file	Int64
qty_hashtag_params	Int64
qty_plus_file	Int64
qty_questionmark_file	Int64
qty_slash_file	Int64
qty_space_directory	Int64
qty_space_file	Int64
qty_tilde_file	Int64
qty_tilde_params	Int64

Table 1: Feature Extraction using Coefficient Correlation

3.4 Training and Testing

In this phase the dataset is split into 70:30 ratio for training and testing sets respectively. This is implemented throughout the three algorithms. A function has been implemented to calculate how long it takes the model to train itself (in seconds). Once the model is train it is tested against test data. Confusion matrix is used for prediction analysis and to check the performance of a classification-based machine learning models. By implementing the matrix, it is easy to determine the accuracy of the model by observing the diagonal values for measuring the number of accurate classifications ("What is Confusion Matrix? | Analytics Steps,"2021).
This performance measurement method is used in the research for evaluating the metrics such as recall, precision, accuracy, f1 measure and time taken to train the model.

4 Design Specification

4.1 Machine learning algorithms

A total of three machine learning algorithms logistic regression, decision tree, and K-NN approaches are used in this analysis. All three are the supervised machine learning algorithm and are frequently used for classification as well as regression problem-solving. The working of these algorithms are as follows:

4.1.1 K-Nearest Neighbors

This is one of the fundamental supervised machine learning models frequently used for classification tasks. This algorithm is also suitable for missing value imputations and extracting a sample dataset from a huge dataset. The "K" in the algorithms stands for selecting the K nearest data points to predict the class of other variables. The Value of K is important to find the accuracy in the model. The model training and testing accuracy rises as the complexity of

the model increases. Value of k determines the complexity i.e., lower value means more complex ("K-nearest Neighbors (KNN) Classification Model," 2021). There are no predefined statistical methods to find the definite value of k. KNN is widely used in pattern recognition and analytical evaluation. So, in this research, the value of k is randomly defined as 5 to train the model(Band, 2021).

4.1.2 Decision Tree

The supervised ML model is trained by using the training dataset and the response of the validation dataset is used to predict the output. The decision tree algorithms are used for the classification of both categorical and continuous variables. It is a top-down approach as the root node lies above the tree and further splits into a variety of branch nodes. In simple terms decision trees are nothing more than a sequence of if-else statements. It examines if the condition is true, and if it is, it moves on to the next node in the decision sequence("Decision Tree Tutorials & Notes | Machine Learning," 2021).

To build the decision tree classifier model in Python, scikit-learn library is imported. To train and test the model, the dataset which have 21 features that are extracted using the coefficient correlation method and target observation column as 'Phishing'. DecisionTreeClassifier function is imported from sklearn library. The declaration of x and y is needed i.e. extracting the attribute variables. The values of x and y were determined using dataset in which x contains all data attributes except the target observation 'Phishing' and y has the target label 'Phishing'. As mentioned earlier in this report, the dataset is divided in 70:30 ratio for training and testing of data in the model. To produce the train and test sets scikit- learn's 'train_test_split' function is used in this model to evaluate performance metrics.

4.1.3 Logistic Regression

Logistic regression algorithm is also a supervised learning algorithm and used for classification & regression tasks in the case of binary variables. The logistic regression algorithm is used as a classification method for categorizing the phishing and legitimate website in the given dataset. The accuracy of the model is more when a large number of observations are present in the dataset ("Machine Learning - Logistic Regression," 2021).

The target variable, 'Phishing,' in this dataset, contains only two kinds of observations: 0 or 1, indicating that it subjects to binary logistic regression algorithm. The model is also evaluated by training and testing the dataset with a small number of observations to see how well it performs.

5 Implementation

This section will go over the steps taken to implement the proposed model into action. This section also discusses the hardware and software used, as well as the coding structure.

5.1 Hardware and Software Requirements:

The model is build using following hardware specifications:

Device name:	DESKTOP-FT5V163
Processor:	Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz
Installed RAM:	8.00 GB (7.76 GB usable)
System type:	64-bit operating system, x64-based processor

The following software requirements were used to build the proposed model:

Base OS:	Windows 10 64-bit
Development Environment:	Jupyter Notebook
Development Language:	Python 3
Libraries imported:	Numpy, Pandas, Matplotlib, sklearn, Seaborn, DecisionTreeClassifier, LogisticRegression, KNeighborsClassifier

5.2 Data files

80K_analysis.ipynb: This file contains the entire code needed to build the model, which was done in a Jupyter notebook using Python 3.

21Features_80kValues_Dataset.csv :This dataset is used for training and testing the model and it contains the 88647 number of observations, 21 features extracted by using coefficient correlation feature selection method and the target label ‘Phishing’.

5.3 Program and Development of the model

The model is built to train and test the dataset on the different machine learning algorithms which was performed in Jupyter Notebook, which is an IPython environment for programming. Firstly, start by importing the necessary Python libraries. These include Pandas, Numpy, Seaborn, Matplolib, Sklearn, DecisionTreeClassifier, LogisticRegression, and KNeighborsClassifier. After importing the dataset, preprocessing on the dataset is done in order to remove any NaaN, infinite, or missing values. After cleaning the data, a correlation test was run to determine the relationship between the various features. This is accomplished using the built-in correlation function, as 'matplotlib' and 'seaborn,' which allow the correlation to be plotted.The dataset was divided into X and Y, with X representing the independent variables and Y representing the dependent variables. The dataset is then divided into training and testing sets in a 70:30 ratio.The dataset is trained across the three algorithms. The elapsed_time function is used to calculate the time required for the models to train in seconds. Finally, ‘confusion matrix' is used to show the true positives, true negatives, false positives, and false negatives. Furthermore, the "accuracy score," "precision score," "recall score," and "f1 score" metrics are used in this study to calculate the accuracy, precision, recall, and f1-score.

6 Evaluation

The transformation and fundamental analysis of the dataset is successfully done in "Python" environment. Also, three machine learning model that are Logistic Regression, Decision Tree, and KNN are captured successfully. The test result of each model is recorded against the performance metrics. The accuracy score, precision score, f1 score, recall score, time taken by each model to train itself, and how the model performs with a smaller number of observations are recorded as a part of the performance metrics. Below are the metrics that are used in this study to evaluate the models ("Confusion Matrix, Accuracy, Precision, Recall, F1 Score | by Harikrishnan N B | Analytics Vidhya | Medium," 2021):

Accuracy refers to the ratio of the number of correctly predicted observations to total number of observations.

$$\text{Accuracy} = TP + TN / TN + FP + TP + FN$$

Recall is the ratio of the number of correct predictions to the total number of actual values.

$$\text{Recall} = TP / TP + FN$$

Precision is the ratio of the number of correctly identified positive observation to the total number of positive observations.

$$\text{Precision} = TP / TP + FP$$

F1 score is the value obtained from the weighted average of Precision and Recall. When two models have low precision and high recall or vice versa it is hard to compare so in that case f1 score is used.

$$\text{F1 Score} = 2 * \text{Recall} * \text{Precision} / \text{Recall} + \text{Precision}$$

The above scores will be calculated using the values generated by the confusion matrix, which include the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) (FN). These four metrics are discussed in more detail below:

True Positive: A true positive occurs when a model predicts that a value is true and it is absolutely true.

False Positive:A false positive occurs when a model predicts a value to be true when it is clearly false.

True Negative: When a model predicts a value to be false when it is totally true, this is referred to as a true negative.

False Negative: A false negative occurs when a model predicts a value to be true when it is in fact false.

6.1 Discussion

Three supervised machine learning methods were used in this study to identify phishing in the dataset. The comparison study for all three models was effectively implemented, and Table 2 shows the evaluation of all three models based on performance measures.

Machine Learning Models	Accuracy	Precision	F1	Recall	Time taken to train the Model (Seconds)
Decision Tree	86.58	73.22	86.87	97.407	0.062
K-Nearest Neighbor(KNN)	86.493	72.703	86.808	97.562	37.665
Logistic Regression	86.601	72.881	86.91	97.502	1.844

Table 2: Comparative Analysis of 3 Models

Accuracy describes the overall performance of the model. By analysing the overall accuracy obtained by all the supervised learning models such as decision tree, KNN and logistic regression, the logistic regression has the highest accuracy, i.e., 86.60%, while there is no significant difference in the accuracy of the other two models. Logistic regression models are highly suitable for binary classification tasks. KNN is referred to as the "lazy learning model" as there is no training involved. During testing, k neighbors with a minimum distance, will take part in classification and regression. Therefore, with the greater sample size, the time taken to train and predict by the model using KNN is much higher in comparison to other two supervised models (“Comparative Study on Classic Machine learning Algorithms | by Danny Varghese | Towards Data Science,” 2021).

Decision tree perform better for categorical values than logistic regression. Since the data used in the research was having categorical values, the decision tree was better in terms of the training speed of the model. The precision score calculated by the model shows that decision tree has the highest precision of 73.22%. As a result of the analysis, Logistic regression performed better in terms of phishing detection on the given dataset as compared to other two algorithms.

7 Conclusion and Future Work

A comparative analysis of a machine learning model that can detect the Phishing website dataset is included in this report. A large dataset containing features about phishing attacks is analyzed using various machine learning algorithms in this report, and a comparative analysis of the employed Machine learning models is also done to select the highly accurate model. Despite the fact that logistic regression has the highest accuracy (86.60%), no significant differences were found in the other two models(Decision tree, KNN). In terms of model training time, decision trees are faster than logistic regression and KNN. In the future, more

machine learning models will be compared to deep learning methods such as neural networks and auto encoders in order to improve phishing detection results.

References

[2009.11116] Phishing Detection Using Machine Learning Techniques [WWW Document], 2021. URL https://arxiv.org/abs/2009.11116 (accessed 12.01.21).

A Comparison of Naive Bayes Methods, Logistic Regression and KNN for Predicting Healing of Covid-19 Patients in Indonesia | IEEE Conference Publication | IEEE Xplore [WWW Document], 2021. URL https://ieeexplore.ieee.org/document/9431845 (accessed 12.04.21).

Alanezi, M.A., Homeed, M.T., Mohamed, Z.S., Zeki, A.M., 2020. Comparing Naïve Bayes, Decision Tree and Logistic Regression Methods in Fraudulent Credit Card Transactions, in: 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI). Presented at the 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI), pp. 1–5. https://doi.org/10.1109/ICDABI51230.2020.9325705

Ali, S., Shahbaz, M., Jamil, K., 2019. Entropy-Based Feature Selection Classification Approach for Detecting Phishing Websites, in: 2019 13th International Conference on Open Source Systems and Technologies (ICOSST). Presented at the 2019 13th International Conference on Open Source Systems and Technologies (ICOSST), pp. 1–6. https://doi.org/10.1109/ICOSST48232.2019.9044042

Alswailem, A., Alabdullah, B., Alrumayh, N., Alsedrani, A., 2019. Detecting Phishing Websites Using Machine Learning, in: 2019 2nd International Conference on Computer Applications Information Security (ICCAIS). Presented at the 2019 2nd International Conference on Computer Applications Information Security (ICCAIS), pp. 1–6. https://doi.org/10.1109/CAIS.2019.8769571

Assegie*, T.A., 2021. K-Nearest Neighbor Based URL Identification Model for Phishing Attack Detection. IJAINN 1, 18–21. https://doi.org/10.35940/ijainn.B1019.041221

Band, A., 2021. How to find the optimal value of K in KNN? [WWW Document]. Medium. URL https://towardsdatascience.com/how-to-find-the-optimal-value-of-k-in-knn-35d936e554eb (accessed 12.5.21).

Comparative Study on Classic Machine learning Algorithms | by Danny Varghese | Towards Data Science [WWW Document], 2021. URL https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222 (accessed 12.5.21).

Confusion Matrix, Accuracy, Precision, Recall, F1 Score | by Harikrishnan N B | Analytics Vidhya | Medium [WWW Document], 2021. URL https://medium.com/analytics-

vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd (accessed 12.14.21).

Datasets for phishing websites detection - ScienceDirect [WWW Document], 2021. URL https://www.sciencedirect.com/science/article/pii/S2352340920313202 (accessed 12.14.21).

Decision Tree Tutorials & Notes | Machine Learning [WWW Document], 2021. HackerEarth. URL https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/ml-decision-tree/tutorial/ (accessed 12.7.21).

Dutta, A.K., 2021. Detecting phishing websites using machine learning technique. PLOS ONE 16, e0258361. https://doi.org/10.1371/journal.pone.0258361

Handling Missing Data | Python Data Science Handbook [WWW Document], 2021. URL https://jakevdp.github.io/PythonDataScienceHandbook/03.04-missing-values.html (accessed 11.20.21).

Herland, M., Bauder, R.A., Khoshgoftaar, T.M., 2019. The effects of class rarity on the evaluation of supervised healthcare fraud detection models. Journal of Big Data 6, 21. https://doi.org/10.1186/s40537-019-0181-8

K-nearest Neighbors (KNN) Classification Model [WWW Document], 2021 . ritchieng.github.io. URL http://www.ritchieng.com/machine-learning-k-nearest-neighbors-knn/ (accessed 11.15.21).

Machine Learning - Logistic Regression [WWW Document], 2021. URL https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_logistic_regression.htm (accessed 11.15.21).

Marchal, S., Francois, J., State, R., Engel, T., 2014. PhishStorm: Detecting Phishing With Streaming Analytics. IEEE Transactions on Network and Service Management 11, 458–471. https://doi.org/10.1109/TNSM.2014.2377295

Odeh, A., Keshta, I., Abdelfattah, E., 2021. PhiBoost- A novel phishing detection model Using Adaptive Boosting approach. Jordanian Journal of Computers and Information Technology (JJCIT) 07, 65–74. https://doi.org/10.5455/jjcit.71-1600061738

Olegario, T.V., Baldovino, R.G., Bugtai, N.T., 2020. A Decision Tree-based Classification of Diseased Pine and Oak Trees Using Satellite Imagery, in: 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). Presented at the 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), pp. 1–4. https://doi.org/10.1109/HNICEM51456.2020.9400002

Shirazi, H., Bezawada, B., Ray, I., 2018. *"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. https://doi.org/10.1145/3205977.3205992

Vanhoenshoven, F., Nápoles, G., Falcon, R., Vanhoof, K., Köppen, M., 2016. Detecting malicious URLs using machine learning techniques, in: 2016 IEEE Symposium Series on Computational Intelligence (SSCI). Presented at the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–8. https://doi.org/10.1109/SSCI.2016.7850079

Vrbančič, G., 2020. Phishing Websites Dataset 1. https://doi.org/10.17632/72ptz43s9v.1

What is Confusion Matrix? | Analytics Steps [WWW Document], 2021. URL https://www.analyticssteps.com/blogs/what-confusion-matrix (accessed 12.07.21).

What Is the Correlation Coefficient? [WWW Document], 2021. Investopedia. URL https://www.investopedia.com/terms/c/correlationcoefficient.asp (accessed 12.07.21).

Yi, P., Guan, Y., Zou, F., Yao, Y., Wang, W., Zhu, T., 2018. Web Phishing Detection Using a Deep Learning Framework. Wireless Communications and Mobile Computing 2018, e4678746. https://doi.org/10.1155/2018/4678746