Covenant Journal of Informatics & Communication Technology. Vol. 9 No. 2, December 20
ISSN: *print* 2354 – 3566 *electronics* 2354 – 3507 DOI:

An Open Access Journal Available Online

Classification of Cybersecurity Incidents in Nigeria Using Machine Learning Methods

Onyinye Onyekpeze, Olumide Owolabi, Bisalla Hashim Ibrahim

Department of Computer Sciences, Faculty of Science, University of Abuja, FCT, Nigeria.
danielonyinye@gmail.com, olumide.owolabi@uniabuja.edu.ng,
hashim.bisallah@unabuja.edu.ng

Received: 27.06.2021 Accepted: 27.11.2021
Date of Publication: December 2021

Abstract—Cybercrime has become more likely as a result of technological advancements and increased use of the internet and computer systems. As a result, there is an urgent need to develop effective methods of dealing with these cyber threats or incidents to identify and combat the associated cybercrimes in Nigerian cyberspace adequately. It is therefore desirable to build models that will enable the Nigeria Computer Emergency Response Team (ngCERT) and law enforcement agencies to gain valuable knowledge of insights from the available data to detect, identify and efficiently classify the most prevalent cyber incidents within Nigeria cyberspace, and predict future threats. This study applied machine learning methods to study and understand cybercrime incidents or threats recorded by ngCERT to build models that will characterize cybercrime incidents in Nigeria and classify cybersecurity incidents by mode of attacks and identify the most prevalent incidents within Nigerian cyberspace. Seven different machine learning methods were used to build the classification and prediction models. The Logistic Regression (LR), Naïve Bayes (NB), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), K-Nearest Neighbor (KNN), Decision Tree (CART) and Random Forest (RF) Algorithms were used to discover the relationship between the relevant attributes of the datasets then classify the threats into several categories. The RF, CART, and KNN models were shown to be the most effective in classifying our data with accuracy score of 99% each while others has accuracy scores of 98% for SVM, 89% for NB, 88% for LR, and 88% for LDA. Therefore, the result of our classification will help organizations in Nigeria to be able to understand the threats that could affect their assets.

Keywords/Index Terms—Cybersecurity, threats, incidents, cybercrimes, classification, machine learning

1. Introduction

Countries and organizations must be prepared to deal with cyber incidents as they become more complex, damaging and harmful. A study by PriceWaterhouseCooper, in The Global State of Information Security 2015 explains how cybercrime has evolved to the point where there are over 117,000 attacks per day. Hakak et al. (2020) explained that as the public transitions from physical to online activities, the possibility of cyberattacks victimization rises, potentially resulting in service disruption, financial loss, data breaches, and individual and institutional anxiety. According to Clough (2015), dating back to the 60s until the present, cybercrime is gradually updating as technology develops. Isah et al. (2016) conducted a survey that identified some common cybercrime in Nigeria to include online advance-fee fraud, pornography, software piracy, software cracking, ATM fraud, spam e-mail, website hacking, and personal identification theft (PIT) with a framework to combat the crimes through a proposed National Cybercrime Control Center (NCCC).

However, from the number of incidents received by the Nigeria Computer Emergency Response Team, which is the National CERT saddled with the responsibility of ensuring a safe, secure and resilient cyberspace in Nigeria, we shall be able to identify the most prevalent incidents, and predict the trend of future incidents. Raw data needs to be gradually refined into helpful information and subsequently into knowledge to become valid. The process begins with pre-processing of the raw data, application of the machine learning methods, analysis and interpretation of the results which is used for decision making to functional areas.

Given the increase in cyber incidents and the associated cybercrimes in Nigeria, law enforcement agencies are facing considerable difficulties in intercepting, arresting, and prosecuting cybercriminals. Also, the dependency of much of society on information and communication technologies makes them highly vulnerable to attacks. It is therefore desirable to build a model that will allow ngCERT, and law enforcement agencies in Nigeria to get helpful knowledge of insights from the available data to detect or identify the most prevalent cyber incidents, the trends of cyber incidents and predict future trends.

The aim of this research, therefore, is to utilize machine learning techniques to classify and understand cybercrime incidents recorded by the Nigeria Computer Emergency Response Team (ngCERT). It also attempts to build models to characterize cyber incidents in Nigeria, identify prevalent threats within Nigerian cyberspace, and identify the suitable Machine Learning Algorithms to classify the cyber incidents in Nigeria.

2. Methodology

This study seeks to introduce the use or application of various machine learning methods in classifying cyber incidents in Nigeria using the data generated from the ngCERT cyber threats intelligence platforms. It utilizes machine learning techniques to study and understand cybercrime incidents or threats recorded by the Nigeria Computer Emergency Response Team (ngCERT) to build models to characterize cybercrime incidents in Nigeria, classify cybersecurity incidents by mode of

attacks and identify the most prevalent incidents within the Nigerian cyberspace. To proceed, we employ the clustering technique to determine the type of cybercrime. Clustering was useful in grouping data with similar characteristics. This grouping aids in the discovery of similar data patterns that occur frequently. Our classification algorithms then supervise or "train" a model with specific data to provide predictions of the target variable, denoted y. Training a dataset was carried out by selecting some essential features or predictors and combining them with a response y (labelled data) that is the observed value of the target variable.

2.1 Feature Extraction

The dataset for the project was obtained from the Nigeria Computer Emergency Response Team (ngCERT) Intelligence Platform between the years 2019 to the year 2021. The dataset comprises mainly categorical features, dates and IP addresses. Data were retrieved from the Nigeria Computer Emergency Response Team (ngCERT) Cyber Intelligence Platform in a Comma-Separated Values (CSV) format for ease of access and manipulation using the various python packages. The columns in the dataset include IP Addresses, ASNName, Activity Date, Country, Infection, Type, Category, amongst other irrelevant features.

- IP Addresses: This lists the IP Addresses of the ASN used in the attack.
- ASNName: This is the name of the ISPs in Nigeria involved in the attack.
- Activity Date: This is the date the attack was recorded.
- Country: This is the country the attack was targeted at.
- Infection: This column records the name of the threat or incident such as Conficker, gamut, extortion, andromeda, iotmicrosoftds, sendsafe, zeroaccess, etc
- Type: This column records the type of attack as either Spam or SpamBot.
- Category: This records the categories of the different types of incidents, such as Botnet, Spam, Vulnerability, Web Defacement, etc

From our dataset, only three features are relevant to the target variable.

2.2 Model Training and Algorithms Used

We train the models by providing our learning algorithm with training data to learn from. The target or target attribute must contain the right answer and must be included in the training data. In this study, the target attribute is the Category Column. The learning algorithm looks for patterns in the training data that map the attributes of the input data to the target, and it produces an ML model that captures these patterns. The model is then used to make predictions on new data for which you do not know the target. We trained the dataset and fit different models using the Logistic Regression, Support Vector Machine, Decision Tree, K-Nearest Neighbor, Nave Bayes, and Random Forest algorithms. Then we used the KModes Clustering Algorithm to group the data that has similar features.

2.2.1: KModes Clustering Algorithm

The KModes Clustering Algorithm is a form of unsupervised learning method which breaks the data points by dividing them into various categories in such a way that from each of the divided groups, every data point in the same group resembles and differs from data

points in other groups. It is essentially a collection of objects based on their similarities and differences.

In Bonthu (2021), the KModes algorithm helps us define clusters based on the number of matching categories between data points. We use KModes clustering when we want to cluster categorical variables. For categorical data points, we cannot calculate the distance, so we go for the KModes algorithm. It uses the dissimilarities between the data points. This type of algorithm uses Modes instead of Means through the following process:

1. Pick K observations as leaders/clusters at random.
2. Calculate the dissimilarities and assign each observation to its closest cluster.
3. Define new modes for the clusters

Repeat 2–3 steps until there are is no re-assignment required.

4. Use Elbow curve to find optimal K value.

2.2.2: Random Forest Algorithm (RF)

The Random Forest comprises multiple decision trees. The Random Forest algorithm can be described as follows:

Step 1: Select k features randomly from the dataset and build a decision tree using those features where k < m. Where m denotes the total number of features.

Step 2: Repeat this n times to obtain n decision trees from various random combinations of k features.

Step 3: To acquire a total of n outcomes from n decision trees, take each of the n Decision Trees and forecast the outcome with a random variable.

Step 4: Each tree in the forest predicts which category the new record belongs to and the category to which the new record is placed with the most votes.

Step 5: It is feasible to utilize and tune a Random Forest model based on established conditions that will offer instructions to the algorithm to create the trees that make up the forest using Python's scikit learn module.

2.2.3:Decision Tree Algorithm (CART)

The Decision Tree Algorithm divides a dataset into smaller subsets using if-then-else decision rules within the data's features. The basic principle behind a decision tree is that the algorithm evaluates each characteristic and uses it to split the tree based on how well it can explain the target variable. The characteristics could be categorical or continuous variables. The algorithm selects the most critical features in a top-down approach while building the tree, creating decision nodes and branches, and making predictions at points where the tree cannot be expanded further.

2.2.4: Support Vector Machine Algorithm (SVM)

Based on their properties and a set of previously classified examples, the Support Vector Machine divides new, unseen objects into two distinct groups. The feature space is divided into two subspaces by the algorithm. Following the establishment of these subspaces, previously unseen data can be classified in some of these locations. The program uses a technique known as the kernel trick to convert the data and identify an optimal boundary between the available outputs when dealing with non-linear connections. Essentially, these are techniques for projecting data into a higher dimension so that a linear separator is sufficient to split the feature space.

2.2.5: K-Nearest Neighbor (KNN)

The K-Nearest Neighbor algorithm learns from training data, then, based on the labels of its nearest neighbours in the training data, predicts the label of any category. As a result, the features used to describe the structure of the data points are most relevant to their labels, bringing them closer to the points with the same label. KNN is a straightforward machine learning classification algorithm based on the assumption that items that look alike must be the same. One of the main advantages of the KNN technique is that it is effective for extensive training data and robust to noisy training data. Scaling KNN queries across massive high-dimensional multimedia datasets presents an exciting challenge for KNN classifiers. A high-performance multimedia KNN query processing system was created to address this issue.

2.2.6: Logistic Regression (LR)

Using logistic regression, we discovered a relationship between input features and output labels. While in logistic regression, we consider the label's category, in linear regression, we find the label's value. For example, predicting the number of attacks is a regression problem, but predicting whether the attack is Spam or Botnet is a classification problem.

2.2.7: Naive Bayes (NB)

The Naive Bayes method forecasts a target variable using some characteristics. Naive Bayes differs from previous classification algorithms in that it assumes that features are unrelated to one another and have no correlation. As a result, this hypothesis is not evaluated in the context of real-world issues. As a result of this naive assumption that features are uncorrelated, this algorithm is called Naive Bayes. We make predictions with the Naive Bayes algorithm by assuming that the given characteristics are independent.

2.2.8: Linear Discriminant Analysis (LDA)

This algorithm is a linear classification machine learning algorithm. Based on the unique distribution of observations for each input variable, the algorithm generates a probabilistic model for each class. Then, for each class, the conditional likelihood of a new example is computed, and the class with the highest probability is picked. You can use linear discriminant analysis to divide a response variable into two or more classes when you have a collection of predictor variables. LDA make predictions based upon the probability that a new input dataset belongs to each class. The class which has the highest probability is considered the output class and then the LDA makes a prediction (Priyankur, 2019)

2.3: Model Evaluation

The models were evaluated using the following performance metrics:

Accuracy: It is the ratio of the number of correct predictions to the total number of input samples. In other words, it is the proportion of the total number of correct predictions.

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

Precision: It is the number of correct positive results divided by the number of positive results predicted by the classifier.

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

F1-Score: F1-Score is used to measure a test's accuracy. F1 Score is the Harmonic Mean between precision and recall. The range for the F1 Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances).

$$\text{F1 Score} = \frac{2 * (\text{precision} * \text{Recall})}{\text{precision} + \text{Recall}}$$

Recall: It is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive). It is the proportion of actual positive cases which are correctly identified.

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

Confusion Matrix: Confusion Matrix gives us a matrix as output and describes the complete performance of the model.

3.0: System Design

The models are built using the python programming language. Python provides a huge number of data analysis and visualization packages used for classification and predictions. Some of the packages include but are not limited to scikit-learn, pandas, numpy, matplotlib, etc. The program is deployed in a Windows Operating System running any version of python version 3. After installation of python, import the required python packages with associated libraries and the dataset. Then the program is deployed and run on pycharm Integrated Development Environment (IDE).

4.0: Performance Evaluation

Evaluation of the model was carried out using the seven (7) identified machine learning algorithms. This section shows how the model performed during implementation. A descriptive explanation of the meaning of each performance metric has been described in section 3.8. The precision result explains what percentage of the items predicted to be relevant by the classifier are relevant. The percentage of items found by the truly relevant classifier is indicated by the recall. Here our X_train and Y_train are fit into the model, then our X_test is used to make the prediction. The outcome is evaluated by showing the accuracy_score, confusion_matrix and the classification report comprising precision, recall, f1_score and support. These are used to determine the performance of our model.

4.1: Random Forest Algorithm

This model was able to achieve an accuracy score of 99%. 52, 605 was correctly classified according to the confusion matrix. The result for the evaluation of the Random Forest is shown in Figure 1.

```
Accuracy:  0.9853566447838362
Confusion Matrix:
 [[52605   211     0   272]
 [    9  5013     0    56]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       1.00      0.99      1.00     53088
          Spam       0.96      0.99      0.97      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.33      1.00      0.50       319

      accuracy                           0.99     58798
     macro avg       0.57      0.74      0.62     58798
  weighted avg       0.99      0.99      0.99     58798
```

Figure 1: Model Evaluation for Random Forest

4.2 Decision Tree (CART)

The performance evaluation of the Decision Tree model is shown below. It can be seen that it has the same performance as the Random Forest Model based on the data we have. It showed an accuracy of 99%.

```
Accuracy:  0.9853566447838362
Confusion Matrix:
 [[52605   211     0   272]
 [    9  5013     0    56]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       1.00      0.99      1.00     53088
          Spam       0.96      0.99      0.97      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.33      1.00      0.50       319

      accuracy                           0.99     58798
     macro avg       0.57      0.74      0.62     58798
  weighted avg       0.99      0.99      0.99     58798
```

Figure 2. Model Evaluation for Decision Tree

4.3 Linear Discriminant Analysis

The performance evaluation of the Discriminant Analysis model is shown below. It can be seen that it performed lesser than the Random Forest and the Decision Tree models. It has an accuracy of 88%.

```
Accuracy:  0.8846729480594578
Confusion Matrix:
 [[51051   243     0  1794]
 [ 4080   647     0   351]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       0.93      0.96      0.94     53088
          Spam       0.73      0.13      0.22      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.11      1.00      0.21       319

      accuracy                           0.88     58798
     macro avg       0.44      0.52      0.34     58798
  weighted avg       0.90      0.88      0.87     58798
```

Figure 3. Model Evaluation for Linear Discriminant Analysis

4.4 Logistic Regression Model

By referring to the probability distribution that defines multi-class probabilities as a multinomial probability distribution, we modified the logistic regression model to predict the probability that an input example belongs to each known class label directly. The performance evaluation of the Logistic Regression model shows that the model did not perform well on the data compared to others. It has an accuracy of 88%.

4.5 K-Nearest Neighbors

The performance evaluation of the K-Nearest Neighbors model is shown below. It can be seen that it performed exactly as the Random Forest and the Decision Tree models. It has an accuracy of 99%.

```
Accuracy:  0.9852886152590224
Confusion Matrix:
 [[52604   212     0   272]
 [   12  5010     0    56]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       1.00      0.99      1.00     53088
          Spam       0.96      0.99      0.97      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.33      1.00      0.50       319

      accuracy                           0.99     58798
     macro avg       0.57      0.74      0.62     58798
  weighted avg       0.99      0.99      0.99     58798
```

Figure 5. Model Evaluation for K-Nearest Neighbors

4.6: Gaussian Naive Bayes

The performance evaluation of the Naive Bayes model is shown below. It can be seen that it performed lesser than the Random Forest and the Decision Tree models. It has an accuracy of 89%.

```
Accuracy:  0.8846729480594578
Confusion Matrix:
 [[51051   243     0  1794]
 [ 4080   647     0   351]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       0.93      0.96      0.94     53088
          Spam       0.73      0.13      0.22      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.11      1.00      0.21       319

      accuracy                           0.88     58798
     macro avg       0.44      0.52      0.34     58798
  weighted avg       0.90      0.88      0.87     58798
```

Figure 6: Model Evaluation for Gaussian Naive Bayes

4.7 Support Vector Machine

The performance evaluation of the Support Vector Machine model is shown below. It can be seen that it performed lesser than the Random Forest and the Decision Tree models. It has an accuracy of 98%. In this case, our data is projected in a higher-dimensional Hilbert space using a kernel function.

```
Accuracy:  0.9753222898738052
Confusion Matrix:
 [[52332   482     0   274]
 [  318  4696     0    64]
 [    0     0     0   313]
 [    0     0     0   319]]
                precision    recall  f1-score   support

        Botnet       0.99      0.99      0.99     53088
          Spam       0.91      0.92      0.92      5078
 Vulnerability       0.00      0.00      0.00       313
Web Defacement       0.33      1.00      0.49       319

      accuracy                           0.98     58798
     macro avg       0.56      0.73      0.60     58798
  weighted avg       0.98      0.98      0.98     58798
```

Figure 7: Model Evaluation for Support Vector Machine

5 Performance Comparison of Different Classifiers

The performance of a model is primarily dependent on the nature of the data. The performances for the seven different Classifiers are shown in Table 1:

.

Table 1: Comparison table for the different Performance Classifiers.

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1_score (%)	Mean Score	Standard Deviation
LR	88	90	97	93	0.876810	0.001560
LDA	88	93	96	94	0.883990	0.001915
KNN	99	100	99	100	0.981858	0.006636
CART	99	100	99	100	0.985218	0.000967
NB	89	0.93	100	94	0.889056	0.002142
RF	99	100	99	100	0.985218	0.000967
SVM	98	99	100	0.99	0.985204	0.000982

5.1 Model Comparison

The chart in Figure 8 shows the performance comparison for the different algorithms when all the Machine Learning algorithms were evaluated.

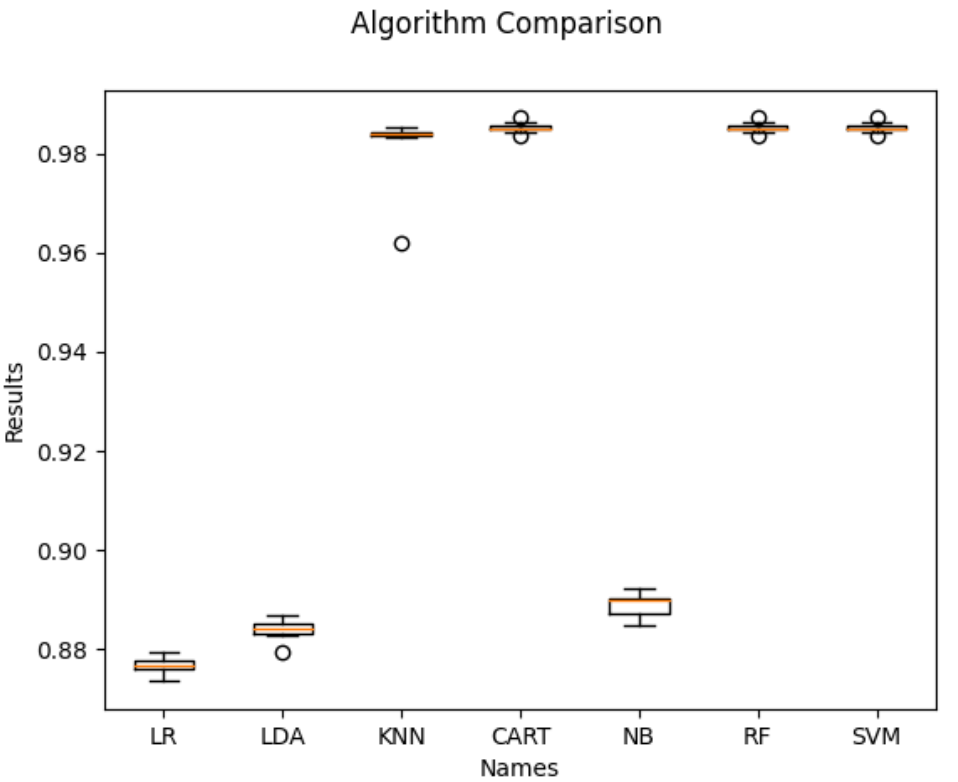

Fig 8: Model performance comparisons

4. Related Work

The current level of practice in the cyber-world in terms of technical capabilities to monitor and trace internet-based attacks is, at best, crude. With existing techniques, tracing sophisticated attacks to their proper source can be nearly difficult (Lipson, 2002). Salas-Fernández et al. (2021) conducted some review studies on Metaheuristic in Attack and/or Defense (MAD) using a systematic review associated with Swarm and Evolutionary algorithms in Intrusion Detection System (IDS) and based on PRISMA methodology by proposing a two-way classification, to determine which of them applied to attack and which to defense, and the second to identify the solved problem (Tactics or Procedure). The main goal was to improve the efficiency of the Intrusion Detection System models. Some of the reviewed algorithms are Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABC), Firefly Algorithm (FA), Bat Algorithm (BAT) and Flower Pollination Algorithm (FPA). The Web of Science search engine was consulted achieving a total of 44 articles. SCOPUS indexing engine, achieving a total of 52 articles. Finally, other indexed sources of articles were consulted achieving a total of 30 articles. A total of 126 documents related to MAD were obtained.

In an attempt to investigate and analyze cyber incidents, several authors proposed various models.

Recently, Aniche et al. (2021) studied Nigeria's current voting system, identified serious flaws in the current paper-based voting system, and designed an E-Voting Biometric system while employing cybersecurity to protect the proposed system from cyberattacks via encryption and decryption algorithms.

According to Mepham et al. (2014), despite the rapidly changing environment and associated risks, standard computer security incident response models have remained essentially unchanged since the 1990s. A review of 90 works claiming to use quantified security investigation and analysis revealed that most of these works' validity was questionable when used in an operational setting. (Verendel, 2009).

The principle of machine learning was used by Prithi et al. (2020) to construct a model utilizing a training dataset that had gone through data cleaning, data transformation, and data reduction using sampling and correlation. The study compares the results of various supervised machine learning methods to predict accuracy. Python is used to start the analytical process, including data cleaning and preparation, missing meanings, experimental analysis, and model construction and evaluation. The Logistic Regression (LR) algorithm employs a linear equation and a prediction model to forecast a value. After that, algorithms such as Logistic Regression (LR), Random Forest (RF), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Decision Tree (DT) were compared. By comparing the improved accuracy, the Logistic Regression generated higher precision prediction results. The Indian Police Department provided the Crime Dataset. By comparing the best accuracy, the logistic regression model produces a more excellent precision prediction result.

Ch et al. (2020) used machine learning techniques to discover and classify attacks that exploit security weaknesses. They constructed a model using a cybercrime dataset acquired from Kaggle and CERT-In.

Two thousand records with attributes: Incident, harm, year, location, offender, victim, age of the offender and cybercrime. Support Vector Machine and Linear Regression were used to build the model and compare with NB, RF. The proposed model has a 99 per cent accuracy rate.

Lekha & Prakasam (2018) developed a model for implementing data mining techniques for cybercrime detection using a cybercrime dataset from an unknown source. SVM, DT, K-mean clustering, and hybrid approach were used to build the model and compare.

Stephen et al. (2020) developed a model using data mining techniques and R software to analyze crime data in Kenya. The crime dataset was extracted from the country's ICT authority website using the APRIORI algorithm, K-Means algorithm and mapping. Multiple crimes are linked, according to the APRIORI method.

Lekha and Prakasam (2017) developed a model using data mining techniques in detecting and predicting cybercrimes in the banking sector. The dataset used was the cybercrime dataset composed of news, feeds, articles, blogs, police department websites and the banking sector. K-mean clustering algorithm and Influenced associative algorithm was used to boost the classification competition and accuracy.

Zolfi et al. (2019) developed a model to investigate and classify cybercrimes through IDS and SVM algorithm, and cyber-attacks datasets were collected from petrochemical companies with 27 features. The NB, DT, LR and SVM algorithm was applied in the classification process, with the SVM providing the best accuracy. Pre-processing and normalization were also discussed and introduced. The techniques are executed using SVM, NB, DT, and LR in tandem. Each of these techniques is used, and the results are presented in various modules. SVM is the most accurate classification technique, with a 99 per cent accuracy rate, allowing for reasonable cybercrime detection in cyber threats. The following algorithms have high accuracy: NB 84 per cent, DT 80 per cent, Logistic Regression 63 per cent, and SVM 99 per cent. As a result, SVM was the most accurate.

Singh and Silakari (2013) proposed a model of Cyber Attack Detection System and its generic framework, which has been found to perform well for all the classes of attack. In this framework, the authors used four tiers architecture to enhance the adaptability of the cyber-attack detection system. The first tier is dedicated to data collection and pre-processing of the data. The Second tier is meant for the feature extraction technique, the third tier is dedicated to classifying cyber-attacks, and the fourth tier is dedicated to the user interface for reporting the events.

Nguyen and Cheng (2011) proposed a new feature selection algorithm for distributed cyber-attack detection and classification. Different sorts of attacks, as well as the network's normal state, are modeled as different classes of network data. Local sensors employ binary classifiers to identify each class from the others. The proposed technique generates a collection of pairwise feature subsets for each local binary classifier, which differentiates that class from the other classes. Unlike traditional feature selection algorithms, which choose a different feature subset for each local binary classifier, this method selects a unique feature subset for each local binary classifier. The novel feature selection technique is more capable of choosing all relevant features, resulting in improved detection and

classification accuracy.

Mishra and Saini (2009) employed a cyber-attack classification approach that uses characteristics metrics and a game-theoretic approach to classifying attacks into their closest categories. To put cyber-attacks into the appropriate group, the standard weights of the metrics were used as a baseline. The method is simple and flexible, as new characters from newly discovered attacks can be added to the attack characteristic metrics, and the suggested formula provides the character with a unique weight. Aside from that, the proposed approach depicts the cause-and-effect link for all possible attacks, assisting us in determining the best way to limit them on the Internet.

Singh et al. (2011) proposed an improved Support Vector Machine (SVM) algorithm for the classification of a cyber-attack dataset. The result shows that SVM gives 100% detection accuracy for Normal and Denial of Service (DOS) classes and is comparable to false alarm rate, training, and testing times. The performance of classic SVM is improved in this study by using conformal mapping to widen the spatial resolution around the border of the Gaussian kernel, increasing the separability of attack classes. It is based on the kernel function's induction of a Riemannian geometrical structure.

5. Conclusion

The increase in cyber incidents and the associated cybercrimes in Nigeria makes this work vital in the fight against cybercriminals. The models will properly classify both present and future cyber incidents into any new categories of threats that are prevalent in Nigerian cyberspace. Therefore, this work used seven different classifiers to obtain the best performing machine learning classification algorithm in building our model: The Logistic Regression (LR), Naïve Bayes (NB), Support vector machine (SVM), Linear Discriminant Analysis (LDR), K-Nearest Neighbor (KNN), Decision Tree (CART) and Random Forest (RF) Algorithms. The Trained model will be very relevant to easily identify and properly classify cyber threats. This will also enhance the development of an efficient incident response plan and ease of identification and response to emerging cyber threats within Nigeria's cyberspace.

The RF, CART, and KNN models were shown to be the most effective in classifying our data with accuracy score of 99% each while others has accuracy scores of 98% for SVM, 89% for NB, 88% for LR, and 88% for LDA.

However, This research will spawn further researches in this area.

This study identified the appropriate models that will be applicable in the Nigerian context based on other research contributions and the identified gaps. The accuracy of the result is measured in the context of Nigeria, and it is the first to be done in Nigeria using data collected from the Nigeria Computer Emergency Response Team (ngCERT) Cyber Monitoring Platform. This study has identified the most appropriate classification models in the Nigerian context, which will help identify and provide a better understanding of the nature of threats within Nigerian cyberspace, allowing for the development of appropriate tactics and information security decisions to prevent or mitigate their impacts. One of such tactics is the work by Alhassan et al. (2020) that created a model to help computer professionals and users stay informed about Cyberethics,

Cyber acts, computer professional codes of conduct, and guidelines.

References

Alhassan, J. K., Abba, E., Misra, S., Ahuja, R., Damasevicius, R., & Maskeliunas, R. (2020). A Framework for Cyber Ethics and Professional Responsibility in Computing. In Advances in Electrical and Computer Technologies (pp. 299-307). Springer, Singapore.

Aniche, C., Yinka-Banjo, C., Ohalete, P., & Misra, S. (2021). Biometric E-Voting System for Cybersecurity. In Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities (pp. 105-137). Springer, Cham.

Arévalo, C., Ramos, I., Gutiérrez, J., & Cruz, M. (2019). Practical experiences in the use of pattern-recognition strategies to transform software project plans into software business processes of information technology companies. Scientific Programming, 2019.

Bonthu, H. (2021). KModes Clustering Algorithm for Categorical data [Online] available from https://www.analyticsvidhya.com/blog/2021/06/kmodes-clustering-algorithm-for-categorical-data/

Brownlee, J. (2016). Machine Learning Mastery [online] available from https://machinelearningmastery.com/

Ch, R., Gadekallu, T. R., Abidi, M. H., & Al-Ahmari, A. (2020). Computational system to classify cybercrime offences using machine learning. Sustainability, 12(10), 4087.

Clough, J., 2015. Principles of cybercrime. Cambridge University Press.

Hakak, S., Khan, W. Z., Imran, M., Choo, K. K. R., & Shoaib, M. (2020). Have you been a victim of COVID-19-related cyber incidents? Survey, taxonomy, and mitigation strategies. Ieee Access, 8, 124134-124144.

Isah, A. O., Alhassan, J. K., Misra, S., Idris, I., Crawford, B., & Soto, R. (2016). Network system design for combating cybercrime in Nigeria. In International Conference on Computational Science and Its Applications (pp. 497-512). Springer, Cham.

Lekha, K. C., & Prakasam, S. (2018). Implementation of data mining techniques for cybercrime detection. International Journal of Engineering, Science and Mathematics, 7(4), 607-613.

Lekha, K. C., & Prakasam, S. (2017). Data mining techniques in detecting and predicting cybercrimes in the banking sector. International Conference on Energy, Communication, Data Analytics, and Soft Computing (ICEC 2017) was held in 2017. (ICECDS) (pp. 1639-1643). IEEE.

Lewis, J. (2018). Economic Impact of Cybercrime-No Slowing Down. Centre for Strategic and International Studies. February 2018.

Lipson, H. F. (2002). Tracking and tracing cyber-attacks: Technical challenges and global policy issues. Carnegie-Mellon Univ Pittsburgh Pa

Software Engineering Inst.

Mepham, K., Louvieris, P., Ghinea, G., & Clewley, N. (2014, June). Dynamic cyber-incident response. The 6th International Conference on Cyber Conflict was held in 2014. (CyCon 2014), (pp. 121-136). IEEE.

Mishra, B. K., & Saini, H. (2009). Cyber attack classification using game-theoretic weighted metrics approach. World Applied Sciences Journal, 7, 206-215.

Nguyen, H. D., & Cheng, Q. (2011, March). An efficient feature selection method for distributed cyber-attack detection and classification. In 2011 45th Annual Conference on Information Sciences and Systems (pp. 1-6). IEEE.

PriceWaterhouseCooper. The Global State of Information Security Survey 2015 [Online]. Annual State of Information Security Survey available from https://www.pwc.com/gx/en/consulting-services/information-security-survey/assets/the-global-state-of-information-security-survey-2015.pdf [accessed 20/01/21].

Prithi, S., Aravindan, S., Anusuya, E., & Ashok-Kumar, M. (2020). GUI Based Prediction of Crime Rate Using Machine Learning Approach. International Journal of Computer Science and Mobile Computing, Vol.9 Issue.3, March-2020, pg. 221-229.

Salas-Fernández, A., Crawford, B., Soto, R., & Misra, S. (2021). Metaheuristic Techniques in Attack and Defense Strategies for Cybersecurity: A Systematic Review. Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities, 449-467.

Singh, S., & Silakari, S. (2013). An ensemble approach for cyber-attack detection system: a generic framework. In 2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (pp. 79-84). IEEE.

Singh, S., Agrawal, S., Rizvi, M. A., & Thakur, R. S. (2011). Improved Support Vector Machine for Cyber Attack Detection. In Proceedings of the World Congress on Engineering and Computer Science (Vol. 1).

Stephen M.W., Joseph N. K., Rachael K., Noah M., (2020). Using Data Mining Techniques and R Software to Analyze Crime Data in Kenya, International Journal of Data Science and Analysis. Vol. 6, No. 1, 2020, pp. 20-31. DOI: 10.11648/j.ijdsa.20200601.13

Zolfi, H., Ghorbani, H., & Ahmadzadegan, M. H. (2019). Investigation and classification of cyber crimes through IDS and SVM algorithm. In 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 180-187). IEEE.