International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

Applying Artificial Intelligence Techniques on Cyber Security Datasets: Detecting Cyber Attacks.

Imad Salah
Computer Science Department
The University of Jordan
isalah@ju.edu.jo

Bayan Alfayoumi
Computer Science Department
The University of Jordan
balfayoumi@ju.edu.jo

Amani Alghareeb
Computer Science Department
The University of Jordan
amanealgareeb1994@gmail.com

Basima Elshqeirat
Computer Science Department
The University of Jordan
B.shoqurat@ju.edu.jo

Mohammad Alshraideh
Artificial Intelligence Department,
The University of Jordan
mshridah@ju.edu.jo

Abstract— The rapid expansion of government and corporate services to the online sphere has spurred a notable surge in internet usage among individuals. However, this increased connectivity also amplifies the risks posed by cyber threats, as hackers exploit external networking avenues and corporate networks for personal activities. Consequently, proactive measures must be taken to mitigate potential financial losses and resource drain from cyber attacks. To this end, numerous machine-learning techniques have been developed for cybercrime detection and threat mitigation. This study evaluates several prominent machine learning methods to identify and address significant cyber threats. The research scrutinizes the effectiveness of five techniques: Random Forest, Decision Tree, Convolutional Neural Network (CNN), K-Nearest Neighbors (KNN), and Naive Bayes. Among these, Random Forest demonstrates superior performance with an accuracy rate of 99.69%, outperforming ensemble models such as Decision Tree, CNN, KNN, and Naive Bayes.

Keywords- Information security, machine learning, Detection process.

I. INTRODUCTION

The process of protecting information by reducing its risks is called information security, more commonly known as information security [1]. This is an essential part of information risk management. This means preventing or restricting unlawful or inappropriate access to data, as well as misuse, disclosure, interception, deletion, destruction, modification, authentication, recording or neglect. It includes measures to reduce the impact of such situations [2]. Information security systems can use machine learning to analyze and learn from trends to help prevent recurring attacks and adapt to changes in behaviour [3,4]. This can help information security teams be more proactive when preventing threats and responding to real-time attacks [5]. Reducing the time spent on routine tasks can help companies allocate their resources more intelligently.

In conclusion, he said machine learning can make information security more accessible, efficient, successful, and affordable [6,7]. However, this can only be done if the underlying machine-learning data accurately represents the oceans [8]. Many attackers using various tactics can target any network traffic to steal or access sensitive data. Only large organizations and companies involved in illegally obtaining information will be used in attacks [9]. Any internal or external network can be a target for these attacks. It is difficult to understand the nature of these attacks because each attack has distinct characteristics that make it difficult to distinguish between them. In addition, people often have difficulty identifying many information security attacks on their own without the help of numerous machine learning algorithms developed by computers using a variety of programming languages[10]. The main objective of this research is to detect and classify different information security attacks, which can target any network, internal or external, using a detection model based on three machine learning (ML) methods.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

II. BACKGROUND AND RELATED WORK

Cybersecurity remains a primary concern in the current infrastructure and technology ecosystems era. Continuous attempts to penetrate organizations and countries' network systems have indicated that traditional firewall and antivirus protections can fail in advanced digital cyberattacks, putting systems at risk. Any network traffic can be attacked by a large number of attackers using a variety of techniques in an attempt to steal or access important information [11]. Attacks can be organized by an individual attacker or large organizations or companies seeking to steal or illegally obtain specific information. These attacks can target any internal or external network, regardless of whether the attackers are internal or external. Because each attack has unique characteristics that make it difficult to differentiate between them, it is difficult to understand the nature of these attacks. Furthermore, without the help of various machine learning (ML) algorithms created by computers using multiple programming languages, humans face a very high level of difficulty detecting many types of information security attacks on their own [12].

Protected information can be intangible (such as paper documents) or physical (like electronic data). Moreover, it focuses on effective policy implementation without sacrificing organizational productivity. An attack that represents an information security threat, such as threats, is an attempt to obtain, alter, destroy, delete, plant, or disclose information without consent or authority. It affects both individuals and organizations [13].

The main goal of information security is to balance data availability, confidentiality, and integrity [14]. Active attacks aim to turn off system resources or disrupt operations. They involve producing false information or manipulating the data flow. The following active attacks are possible: masquerade, message modification, deauthorization, replay, and denial of service (DoS).

In active attacks, attackers attempt to collect information from systems or use them without affecting their resources, unlike passive attacks, where attackers spy or sniff transmissions because the adversary needs information. Therefore, attackers in such a situation want to listen to the conversations, thus obtaining the required information, such as broadcast message content and traffic analysis [15](as shown in Fig 1).

Attack Name	Description	Attack by (Packets, Tools, etc.)
Active Attacks		
Denial of Service (DoS) Attacks		
Jamming Attack	By using the channel that they are communicating on, it prohibits other nodes from accessing it to connect.	Radio frequency noise.
Flooding	A DoS attack in which a server receives many connection requests but does not reply to complete the handshake. (ICMP Flood, SYN Flood, HTTP Flood).	Unbound number of requests without acknowledgment of packet after receiving it.
Smurf Attack	A network layer DDoS attack caused due to the network tools misconfiguration.	Source IP fooling victim IP.
Teardrop Attack	A DoS attack that bombards a network with many Internet Protocol (IP) data fragments, then the network is unable to recombine the fragments back into their original packets.	Sending fragmented packets to the target machine.
Man in the Middle Attacks		
Ransomware	A form of malware that infiltrates and encrypts important files and systems, preventing a person from accessing their own data.	BOt0nt0k (encryption ransomware), Mado (malicious program)
Session Hijacking	To obtain unauthorized access to the Web Server, the Session Hijacking attack disrupts the session token by stealing or guessing a valid session token (e.g., predictable session token).	Malicious JavaScript Codes, XSS, Session Sniffing.
Passive Attacks		
Active Reconnaissance	An intruder is engaged in targeting the system to acquire information about vulnerabilities (e.g., port scanning).	Nmap, Metasploit.
Passive Reconnaissance	Gathering information about computers and networks without actively engaging with them (e.g., eavesdropping, OS fingerprinting).	Wireshark, Shodan.
Traffic Analysis	A method to gather and monitor wireless frames, packets, or messages to drive information for communication patterns.	Sniffing tools.
War Driving	Mapping the wireless access points with wireless networks with vulnerabilities in moving cars.	iStumbler, Global Positioning System (GPS), antenna, Wifiphisher.

Fig 1, Type of Attacks

A. Deep Learning Methodology

Deep learning was introduced according to the progress of feature learning research, large-scale labelled data availability, and hardware [3]. In 2006, G. Hinton initiated representation learning research with the idea of greedy layer-wise pre-training and fine-tuning deep belief networks. This resulted in higher performance than state-of-the-art algorithms on MNIST handwritten digit recognition and document retrieval. Later, many deep learning algorithms were proposed and successfully applied to many domains.

Convolutional Neural Network (CNN) [13] and Recurrent Neural Network (RNN) [13]. While AE and DBN are unsupervised learning models, CNN and RNN are supervised learning models. This paper considers using supervised learning models: CNN, RNN and stacked RNN [14]. An introduction to each model is described below.

1)Convolutional Neural Network (CNN):

Convolutional Neural Networks (**CNNs),** often called ConvNets, are a specialized class of deep learning models designed to excel in tasks involving visual and spatial data, such as image and video analysis. CNNs have revolutionized the field of computer vision and significantly improved the accuracy of tasks like image classification, object detection, facial recognition, and more [29].

The key feature that sets CNNs apart from traditional neural networks is their ability to learn hierarchical features automatically and adaptively from data. This is particularly important when dealing with complex visual data, as CNNs can recognize patterns at multiple levels of abstraction.

Here are some essential characteristics and components of CNNs :

Convolutional Layers: CNNs utilize convolutional layers to apply a set of learnable filters (kernels) to the input data. These filters systematically scan the input, capturing features like edges, textures, and shapes. Convolutional layers are instrumental in preserving the spatial relationships within the data.

Pooling Layers: Pooling layers, typically implemented as max-pooling or average-pooling, reduce the spatial dimensions of the data, which helps reduce computational complexity while retaining the most essential information. This down-sampling operation helps make CNNs computationally efficient.

Activation Functions: Activation functions like ReLU (Rectified Linear Unit) are used to introduce non-linearity into the network, allowing it to model complex relationships within the data.

Fully Connected Layers: These layers connect every neuron in one layer to every neuron in the next, a typical architecture for the final layers of a CNN. Fully connected layers are typically used for classification tasks.

Dropout: Dropout is a regularization technique used in CNNs to prevent overfitting. It randomly drops a fraction of neurons during training, which helps the network generalize better to unseen data.

Transfer Learning: CNNs can leverage pre-trained models on large datasets, like ImageNet, and fine-tune them for specific tasks. This transfer learning approach saves training time and resources.

The success of CNNs can be attributed to their ability to automatically learn and extract features from data, making them well-suited for tasks where feature engineering would be

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

challenging and time-consuming. Their applications extend beyond computer vision into natural language processing (NLP) and speech recognition, where they are adapted to process sequential data effectively.

Overall, Convolutional Neural Networks have revolutionized the field of artificial intelligence, making significant strides in pattern recognition, image analysis, and a wide range of real-world applications. They continue to be a driving force in advancing technology and reshaping how we interact with the visual world.

2)Recurrent Neural Network (RNN): It is a sequence model of neural networks. RNNs have the property of reusing information that has already been given. An RNN is a neural network layer that takes the input and the previous state as inputs applies the tanh operation, and outputs the new state.

3)Stacked Recurrent Neural Network (Stacked RNN): It enhances the abilities of RNN. Each cell of a Stacked RNN can store more information throughout the hidden states between the input and output layers.

Random Forests (R.F.):

Random Forests are an ensemble learning approach that combines multiple decision trees to make predictions. Each decision tree is constructed using a random subset of training data and features, reducing the risk of overfitting and enhancing generalization. Random Forests are adept at handling high-dimensional data, capturing complex feature interactions and providing rankings of feature importance. They have found widespread use in heart disease detection due to their strong performance and interpretability; the Algorithm is as follows:

Algorithm Random Forest
Input: m samples from the Dataset with **n** target classes
H: hypothesis
I: Iterations
1. for all m
2. Do am ← 0
3. for all m
4. for all I
5. x1 ← random sample from m
6. p ← H(x1)
7. x2 ← random sample from m
8. x1[m] ← x2[m]
9. **if** H(x1) ≠ p **then**
10. am ← am + 1

Decision Trees (D.T.):

Decision Trees represent a straightforward yet potent machine-learning algorithm that employs a tree-like structure for decision-making. Internal nodes correspond to features, while leaf nodes signify classes or predictions. Decision Trees recursively split the data based on features, creating decision rules for predicting the target variable. They offer intuitiveness, ease of interpretation and versatility in handling categorical and continuous features. Decision trees are applied in heart disease detection, offering transparent decision-making processes and insights into the importance of features. The algorithms are as follows:

Algorithm 1 Decision Tree Classifier

Input: m samples from the Dataset with target classes
for all attributes, **do**
for each record, **do**
Perform Decision Tree Classifier algorithm.
end for
classify the attribute space.
end for
determine the total leaf nodes n1, n2, n3, …., nm
divide the samples into m1, m2, m3,…, mm according to the leaf noc
Output: Partitioned samples m1, m2, m3,…, mm

Naive Bayes:

Naive Bayes is a probabilistic machine learning technique founded on Bayes' theorem. It operates under the assumption of conditional independence among features given the class label; despite this simplifying assumption, Naive Bayes classifiers have demonstrated effectiveness across various domains, including text classification and medical diagnosis. Naive Bayes algorithms classify patients into distinct disease categories based on features in heart disease detection. The Algorithm is as follows:

Input:
Training Dataset T,
F= (f_1, f_2, f_3…,f_n)//value of the predictor variable in Testing dataset
Output:
A class of Testing datasets.
Step:

1. Read the Training dataset T,
2. Calculate each class's mean and standard deviation of the predictor variables.
3. Repeat

Calculate the probability of f_i using the Gauss density equation in each class.

Until the probability of all predictor variables (f_1, f_2, f_3,…, f_n) has been calculated.

4. Calculate the likelihood for each class.
5. Get the greatest likelihood.

K-Nearest Neighbors (KNN):

K-Nearest Neighbors is a non-parametric machine learning method suitable for classification and regression tasks. Using a specified distance metric, KNN classifies a new instance by identifying the K nearest neighbours within the training data. The class label of the new instance is determined through a majority vote among its K nearest neighbours. KNN is straightforward to implement, although its performance may be sensitive to the choice of distance metric and the value of K. In heart disease detection, KNN is applied, mainly when dealing with non-linear class distributions; the Algorithm is as follows:

Algorithm K-Nearest Neighbors
Input: m samples from a dataset with target classes
1. Calculate Euclidean Distance dis (x, x_i) and arrange it in ascending order.
2. Take the first value from Step 1 and name it k.
3. Find the distance corresponding to the k-points.
4. If $k_j > k_i$, where $j \neq i$ then put x in class j.

Some previous works suggest using different types of machine learning algorithms to detect and classify different types of network attacks in information security based on various network intrusion datasets. Some examples of such datasets are CIDC-2017, NSL-KDD, UNSW-NB15, KDD99 and others.

Many machine learning algorithms are applied to the datasets mentioned above to perform the detection task, which leads to the development of many models to protect the network from internal and external attacks.

Kumar et al. [16] proposed a machine-learning model to identify network threats from suspicious activities. This Dataset, the four-type decision tree (D.T.), is used to test this model (C5, CHAID, CART, QUEST). They used the UNSW-NB15 dataset as an offline dataset containing nine attack models (DoS, Reconnaissance, Backdoor, Fuzzers, Analysis, Exploits, Worms, Shellcode, and Generic) and regular attacks containing 44 features. Then, at the NIT Patna CSE lab, they produce a real-time dataset known as RTNITP18. The offline Dataset also underwent a series of preprocessing techniques, such as feature reduction and dataset reduction (to reduce the size of the Dataset by removing duplicates) (to remove unimportant features). They measured these trees' effectiveness using various evaluation criteria, including recall and precision. Using the UNSW-NB15 benchmark dataset and the RTNITP18 real-time dataset, they showed that the recommended model outperforms existing models, demonstrating higher accuracy (90.74%), attack detection rate and F-measure, Average, Average Accuracy, Attack Accuracy And error. . . . Alarm rate. .

Amaizo et al. [17] Detect network intrusions (DNN) using a deep learning neural network model. They leveraged datasets from NSL-KDD, UNSW-NB15, and CSECIC-IDS2018. Before sending the datasets to DNN, two preprocessing procedures were performed on the three datasets: dimensionality reduction and principal component analysis (PCA) to extract features. For DNNs to work optimally, they specify the number of layers and their locations. The data set is passed to the first hidden layer (dense 1: dense) after the DNN input layer. Three additional coarse layers (Dense2:Dense, Dense2:Dense, and Dense2:Dense) whose outputs are sent to each session using the Rectified Linear Unit (ReLU) session function. Like other hidden layers in the model, this layer is activated using the ReLU activation function. To avoid overfitting, we added dropout to each of the three layers. Finally, the output is provided as input to the output layer, driven by the sigmoid function of the last layer, which communicates at connection 1: the communication layer. The model classifies network traffic as 0 for secure traffic or 1 for attack traffic. The final step is to test the new network traffic for anomalies after the model is fully trained and tested. The system sets the value to 0 for all legitimate network traffic and sets the value to 1 when an anomaly is encountered. Model performance is evaluated using four metrics: precision, recall, and F1 score. The results on the NSL-KDD, UNSW-NB15, and CSECIC-IDS2018 datasets show that the DNN model exceeds 97.89%, 89.99%, and 76.47% on each Dataset, respectively.

Kasongo et al. [18] used a machine learning-based intrusion detection system to detect cyber threats. They used a popular dataset called UNSW-NB15, which contains multiple cyberattacks. They then performed three preprocessing operations on this Dataset: cleaning up errors such as missing values and normalizing to scale using the Min-Max scale feature selection using the XGBoost method to identify key features. The Dataset contains nine attack examples (DoS, Reconnaissance, Backdoor, Fuzzers, Analysis, Exploit, Worms, Shellcode, Generic) and regular attacks with 44 signatures. These algorithms are support vector machines (SVM), maximum gradient boosting (XGBoost), artificial neural networks (ANN), k-nearest neighbours (kNN), logistic regression (L.R.), and decision trees (D.T.). They showed that these methods can identify objects with 90.85% higher accuracy when using XGBoost as a feature selection strategy.

Xiao et al. [19] proposed a new 5-layer autoencoder (A.E.)-based model that is more suitable for network anomaly detection. They took advantage of the well-known NSL KDD dataset in cyber attack research. The Dataset contains 148,517 samples divided into training (125,973) and testing (22,544) datasets. The Dataset contains five categories, divided into two categories (regular attacks and abnormal attacks), each with multiple types. R2L (Xlock, Snmpguess, Httptunnel, Sendmail, Named, Ftp write, Phf, Multihop, Imap, Warezmaster, Warezclient, Snmpgetattack, Spy) and DoS (Smurf, Back, Land, Process table, Neptune, Pod, Apache2, Udpstorm, Worm, Teardrop), U2R (Xterm, Buffer_overflow, Sqlattack, Rootkit, Perl, Loadmodule, Ps) and Probe (Portsweep, Stan, Nmap, Ipsweep, Mscan, Saint) are unusual attacks. They showed that the model gave an accuracy value of 90.61% during intrusion detection.

Koisha et al. [20] proposed a new algorithm for detecting intrusions in network attacks called class support vector machines (OCSVM). They used the well-known NSL-KDD dataset, which contains 148,517 samples divided into training (125,973) and testing (22,544) datasets. The Dataset contains five categories, divided into two categories (regular attacks and abnormal attacks), each with multiple types. R2L (Xlock, Snmpguess, Httptunnel, Sendmail, Named, Ftp write, Phf, Multihop, Imap, Warezmaster, Warezclient, Snmpgetattack, Spy) and DoS (Smurf, Back, Land, Process table, Neptune, Pod, Apache2, Udpstorm, Worm, Teardrop), U2R (Xterm, Buffer overflow, Sqlattack, Rootkit, Perl, Loadmodule, Ps) and Probe (Portsweep, Stan, Nmap, Ipsweep, Mscan, Saint) are unusual attacks. They showed that the model's accuracy reached 81.29% during intrusion detection. In this study, several machine learning algorithms were used to improve the performance of the detection process of different types of information security attacks that attack any network.

Research work [21] proposed the design of a Hybrid Intrusion Detection System that uses Snort and Hadoop. This work aims to integrate Snort with Hadoop and auto-generate new Snort rules for better detection performance. This work focuses on specific types of attacks, such as ICMP attacks, Smurf attacks, SYN flood attacks, UDP attacks, and Port scanning. The researchers found that Snort rules can be generated by adding options for event filtering so that alerts will be created if the number of packets from the source exceeds a particular amount. New rules were generated with the output of the analysis done by Hadoop. The new Snort rules efficiently detected ICMP attacks, Smurf attacks, SYN flood attacks, UDP attacks, and Port scanning.

Sasanka Potluri et al.[22] evaluated the performance of a detection mechanism by combining deep learning techniques with machine learning techniques. They used Theano deep learning library with MATLAB and NSL-KDD DATASET as the input data. The Dataset contains different types of attacks, such as DoS, Probe, R2L, and U2R. They evaluated the performance of the hybrid deep learning approach in combination with stacked autoencoders, DBN, softmax regression, and SVM. They found that the model that gave the best detection accuracy was stacked autoencoders with SVM.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Zhang et al. [23] created a semi-supervised model by exploiting a small number of labelled data with many unlabeled data for intrusion detection. In the first stage, they applied information gain-based feature selection to the NSL-KDD datasets to reduce the redundant features. In the second stage, they used the new training dataset to train a LapSVM-based learning model and obtained the best accuracy of 97.8%.

Yin et al.[24] proposed a deep learning approach for intrusion detection using recurrent neural networks (RNN-IDS) and the NSL-KDD dataset. They also studied the model's performance in binary classification and multiclass classification, and they investigated the impact of the number of neurons and different learning rates on the model's performance. They used machine learning algorithms such as J48, Naive Bayesian, Random Forest, Multilayer Perceptron, Support Vector Machine, and others to train models through the training set by Weka. The results showed that the model had a higher accuracy on the KDDTest+ when there were 80 hidden nodes in the RNN-IDS model, the learning rate was 0.5, and the training was performed 80 **times.**

Vinayakumar[25] evaluated the essential synthetic I.D. dataset, such as KDDCup 99. The researchers analyzed various MLP, CNN, CNN-RNN, CNN-LSTM, and CNN-GRU with their topologies, network parameters, and network structures to select the optimal network architecture. The models in each experiment were run up to 1000 epochs with a learning rate in the range [0.01-0.5]. CNN and its variant architectures performed significantly better than the standard machine learning classifiers. This is mainly because CNN can extract high-level feature representations representing the abstract form of low-level network traffic connections feature sets.

The study [26] introduces a new method to detect web phishing, a significant security concern. It combines four machine learning algorithms, including K-means and supervised techniques, to weigh their outputs strategically. Using a dataset of 111 web features, the study highlights the importance of feature correlation in improving detection accuracy. This research offers an innovative approach to combating web phishing.

The paper [27] introduces two Genetic Programming-based methods, GPM and GPMP, for predicting malware, compared to three popular feature selection techniques. Results show that the proposed methods outperform existing techniques in accuracy and F-score, with faster computation times. Evaluation across four datasets using Random Forest and Decision Tree classifiers confirms their efficiency, particularly with Random Forest achieving superior performance.

The study [28] develops a malware detection model using machine learning classifiers and a new feature selection technique based on genetic programming. Results show that Random Forest, Random Forest (4), and Random Tree classifiers perform the best, while Hoeffding Tree and Decision Stump perform poorly. The proposed feature selection method, GPMP, outperforms Filter-based, with higher accuracy and F1-score and fewer features, reducing computational complexity.

III. PROPOSED METHODOLOGY

A method used in this theory is proposed to identify several network attacks, which can attack any network based on the information specified: dataset description, preprocessing stage, and feature extraction usage—methods and construction of eight methods for machine learning. Fig 2 shows the flow chart of the proposed method.

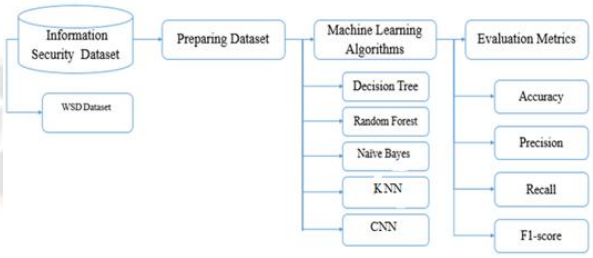

Fig2: Proposed Methodology Flow-chart

A. Dataset Description

This section presents the Dataset used in this thesis to detect specific network attacks in information security from a well-known source. Therefore, this Dataset has 23 features and 374,661 instances with a classification label containing several types of network attacks, as shown below.

1. **Node ID**: a unique I.D. that distinguishes the sensor node in any round and at any stage. For example, node number 25 in the third round and the first stage will be symbolized as 001 003 025.
2. **Time**: the current simulation time of the node.
3. **Is CH?** A flag to distinguish whether the node is C.H. with value one or normal node with value 0.
4. **Who CH?** The ID of the C.H. in the current round.
5. **RSSI**: Received Signal Strength Indication between the node and its C.H. in the current round.
6. **Distance to CH**: the distance between the node and its C.H. in the current round.
7. **Max distance to CH**: the maximum distance between the C.H. and the nodes within the cluster.
8. **Average distance to CH**: the average distance between nodes in the cluster to their C.H.
9. **Current energy**: the current energy for the node in the current round.
10. **Energy consumption**: the energy consumed in the previous round.
11. **ADV_CH send**s the number of advertise C.H.'s broadcast messages to the nodes.
12. **ADV_CH receives**: the number of advertising C.H. messages received from CHs
13. **Join_REQ send**s the number of join request messages the nodes send to the C.H.
14. **Join_REQ receive** the number of join request messages received by the C.H. from the nodes.
15. **ADV_SCH send**s the number of advertise TDMA schedule broadcast messages to the nodes.
16. **ADV_SCH receives** the number of TDMA schedule messages received from C.H.s.
17. **Rank**: the order of this node within the TDMA schedule.
18. **Data sent**: the number of data packets sent from a sensor to its C.H.
19. **Data received**: the number of data packets received from C.H.
20. **Data sent to BS**: the number of data packets sent to the B.S.
21. **Distance CH to BS**: the distance between the C.H. and the B.S.
22. **Send Code**: the cluster sending Code.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

23. **Attack Type**: type of the node. It is a class of five possible values, namely, Blackhole, Grayhole, Flooding, and Scheduling, in addition to normal if the node is not an attacker.

Table 1 and Figure 3 show that the class label has two types of malicious attack (Grayhole, Blackhole, TDMA, and Flooding) and Normal.

Table 1. Attack Classes in the WSN-DS Dataset

Attack	Label	Frequency
Normal	Normal	340066
Grayhole	Malicious	14596
Blackhole	Malicious	10049
TDMA	Malicious	6638
Flooding	Malicious	3312

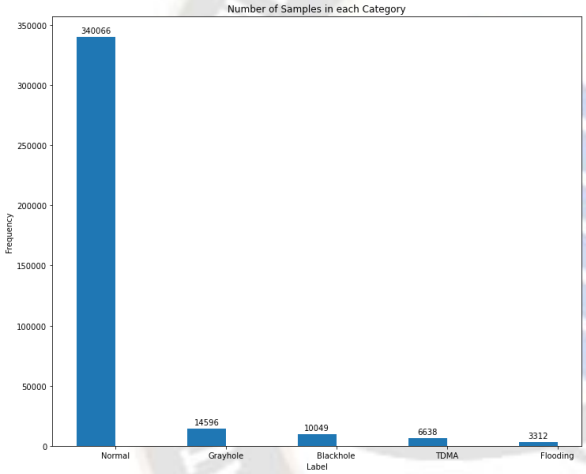

Fig 3. Attack Classes in the WSN-DS Dataset

B. Preparing Dataset

Using a popular encoding technique, we used different machine learning algorithms on this Dataset to convert the non-numerical features to numerical features [16]. This method, known as "Label Encoder," turns non-numerical data into machine-readable form by replacing each value with a unique number starting at 0 [17]. All the features except the Attack type (label) are numerical values, so we converted them to numerical type using this method.

C. Preprocessing

The research aims to build a well-behaved traffic model for different types of attacks by going beyond traditional intrusion detection methods; a multi-cluster approach was used. The procedure performed during the experiments is shown in preprocessing divided into three steps such as "Data Initialization", "Data Preparation", and "Segmentation and Normalization". At this stage, the method shown in Figure 4 is used.

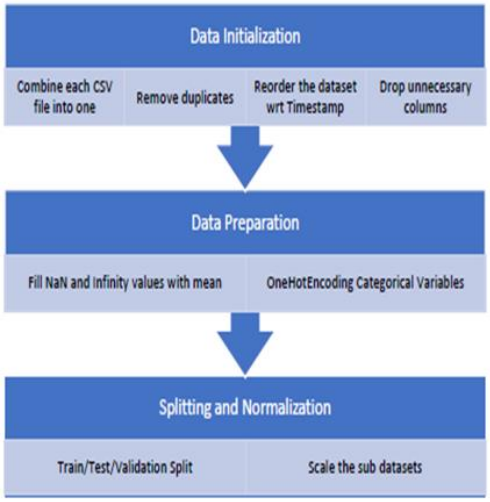

Fig 4, preprocessing methods

D. Evaluation Matrix

Several assessment measures were used to examine the ML algorithms, including accuracy, precision-recall, and f1-score. These matrixes' formulas are shown below, where T.P. = True Positives, TN = True Negcheekye, FP cheeky False Positives, and F.N. = False Neative:Accuracy: The ratio of correctly paggressivected samples to all samples, or simply the ratio of correctly predicted samples to all samples, is the most intuitive performance matrix.

1- Accuracy: The ratio of correctly predicted samples to all samples, or simply the ratio of correctly predicted samples to all samples, is the most intuitive performance metric[29].

$$Accuracy = \frac{\text{TP+TN}}{\text{TP + TN + FP + FN}} \qquad (1)$$

2-Precision: the ratio of correctly predicted positive samples to the total predicted positive samples [30].

$$Precision = \frac{TP}{TP+FP} \qquad (2)$$

3-Recall the proportion of accurately anticipated positive samples to the predicted positive samples [31].

$$Recll = \frac{TP}{TP+FN} \qquad (3)$$

4- F1-score: is the weighted average of Precision and Recall [32].

$$F1\text{-}score = 2 * \frac{Precision * Recall}{Precision + Recall} \qquad (4)$$

IV. RESULTS

In this experiment, machine learning algorithms are applied to detect whether a sample in this Dataset is expected or some malicious attack: Blackhole, Grayhole, TDMA, and Flooding. Figure 5 compares classification algorithms on this Dataset, which reveals impressive performance across the board. The standout performer is the Random Forest model, achieving an accuracy of 99.67%, closely followed by the Decision Tree at 99.47%. While Naive Bayes performs well with 98.32%

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

accuracy, it falls slightly behind these leaders. CNN and K-NN show respectable accuracies of 98.97% and 96.20%, respectively, demonstrating the effectiveness of various approaches for this task. It is important to note that choosing the best Algorithm might depend on factors beyond just accuracy, such as interpretability, computational efficiency, and specific data characteristics.

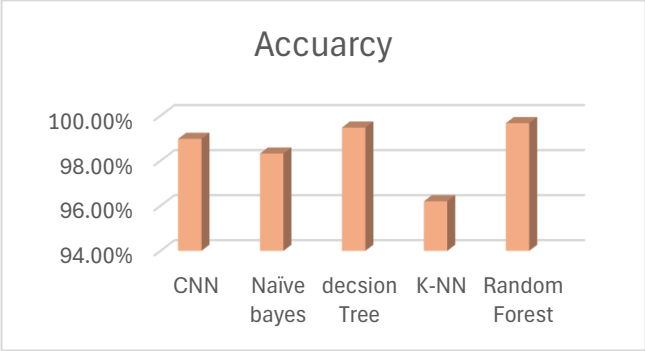

Fig 5: Accuracy of ML Algorithm.

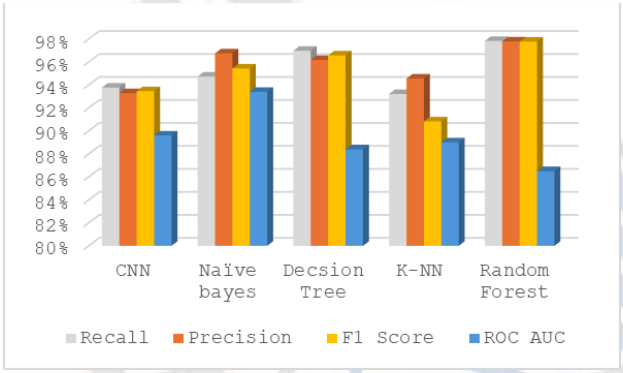

Fig 6: The performance matrix of ML Algorithms.

Figure 6 shows the performance of five different machine learning algorithms on a classification task, as measured by three different metrics: recall, precision, and F1 score. It also includes an additional metric, ROC AUC, which is not included in your question.

- Recall: This metric measures the model's ability to identify positive cases. In other words, it tells you what proportion of the actual positive cases were correctly identified by the model. The table shows that Random Forest has the highest recall of 98%, followed by Decision Tree (97%) and Naive Bayes (95%). CNN and K-NN have the lowest recall, at 94% and 93%, respectively.
- Precision: This metric measures the ability of the model to identify only actual positive cases and avoid false positives. In other words, it tells you what proportion of the cases the model identified as positive were positive. Here, we see that Random Forest again has the highest precision of 98%, followed by Decision Tree (96%) and K-NN (95%). Naive Bayes and CNN have the lowest precision, at 93% and 93%, respectively.
- F1 Score: This metric is a harmonic mean of precision and recall, combining both metrics into a single score. It provides a balanced view of the model's performance, taking into account both its ability to correctly identify true positives and avoid false positives. Consistent with the previous two metrics, Random Forest has the highest F1 score of 98%, followed by Decision Tree (97%) and Naive Bayes (95%). CNN and K-NN have the lowest F1 scores, at 93% and 91%, respectively.

Overall, the results suggest that Random Forest is the best-performing Algorithm, achieving the highest scores in all three metrics. Decision Tree also performs well, closely following Random Forest in all metrics. Naive Bayes performs reasonably well, with slightly lower scores than the top two. CNN and K-NN have the lowest performance among the five algorithms compared here.

It is important to note that these results are based on a single data set, and the performance of these algorithms may vary depending on the specific task and data characteristics. The choice of the best Algorithm may also depend on factors beyond just performance, such as interpretability, computational efficiency, and the application's specific requirements.

A confusion matrix is a table used to determine the performance of a classification algorithm. Visualizes the confusion matrix and summarizes the performance of the classification algorithm[33-34]. This measure used four terms: Output "T.N." Denotes True Negative, which shows the exact number of negative examples classified. Likewise, "T.P." stands for True Positive and indicates the number of accurately classified positive examples. The term "F.B." shows a false positive value, i.e. the number of actual negative examples classified as positive, and the "F.N." Mean false negative value, which is the number of actual positive examples classified as negative.

Figure 7 shows the confusion measure for the R.F. algorithm in the first experiment. Therefore, the R.F. performance results are 100, which means the error rating is zero, as shown in the values of F.P. and F.N. in the figure. In other words, the R.F. model did not make any errors during the classification task in the training and testing processes.

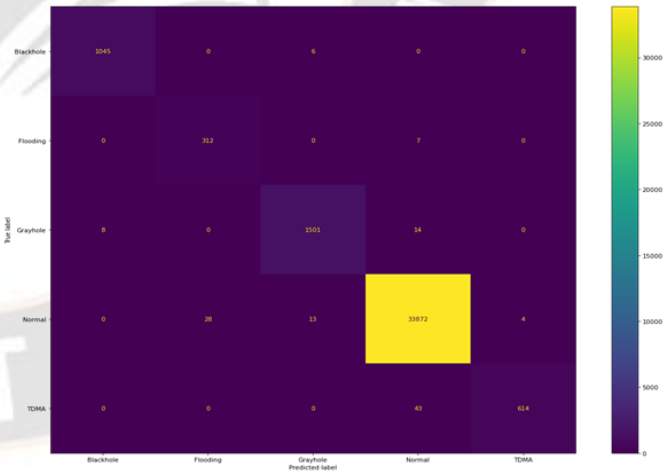

Fig 7, R.F. Confusion Metric.

V. Conclusion

Based on a well-known dataset, five machine learning algorithms are applied to detect network attacks in cybersecurity. To use different machine learning algorithms in this Dataset, we converted the non-numeric features into numeric ones using a standard encryption technique. Random Forest produced the best performance of 99.69% compared to the ensemble models Decision Tree, CNN, KNN, and Naive Bayes. All methods used

to evaluate the models showed that random forest remains the best Algorithm among others. Therefore, it is recommended that it be used to classify scenarios related to detecting cyber-attacks and controlling system operations. However, increasing the amount of data can increase accuracy and time complexity. In future work, we will apply these algorithms to different datasets to verify whether these algorithms can detect multiple attacks using different network datasets. We also used deep learning algorithms on this Dataset and another dataset to see if deep learning can detect various types of networks.

ACKNOWLEDGEMENT

This research was done during the sabbatical leave from the University of Jordan for the academic year 2023-2024.

REFERENCES

[1] G. Breda and M. Kiss, "Overview of Information Security Standards in the Field of Special Protected Industry 4.0 Areas & Industrial Security", Procedia Manufacturing, vol. 46, pp. 580-590, 2020. Available: 10.1016/j.promfg.2020.03.084 [Accessed 10 September 2022].

[2] M. Singh, "An Overview of Automotive Vehicles and Information Security", Information Security of Intelligent Vehicles Communication, pp. 1-13, 2021. Available: 10.1007/978-981-16-2217-5_1 [Accessed 10 September 2022].

[3] F. Alkhudhayr, S. Alfarraj, B. Aljameeli and S. Elkhdiri, "Information Security: A Review of Information Security Issues and Techniques", 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), 2019. Available: 10.1109/cais.2019.8769504 [Accessed 10 September 2022].

[4] N. Zhang, R. Wu, S. Yuan, C. Yuan and D. Chen, "RAV: Relay Aided Vectorized Secure Transmission in Physical Layer Security for Internet of Things Under Active Attacks", IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8496-8506, 2019. Available: 10.1109/jiot.2019.2919743 [Accessed 10 September 2022].

[5] J. Ning, J. Xu, K. Liang, F. Zhang and E. Chang, "Passive Attacks Against Searchable Encryption", IEEE Transactions on Information Forensics and Security, vol. 14, no. 3, pp. 789-802, 2019. Available: 10.1109/tifs.2018.2866321 [Accessed 10 September 2022].

[6] J. Lee, J. Shin and M. Realff, "Machine learning: Overview of the recent progress and implications for the process systems engineering field", Computers & Chemical Engineering, vol. 114, pp. 111-121, 2018. Available: 10.1016/j.compchemeng.2017.10.008 [Accessed 10 September 2022].

[7] Mohammad Alshraideh, Najwan Alshraideh, Abedalrahman Alshraideh, Yara Alkayed, Yasmin Al Trabsheh, Bahaaldeen Alshraideh, "Enhancing Heart Attack Prediction with Machine Learning: A Study at Jordan University Hospital", *Applied Computational Intelligence and Soft Computing*, vol. 2024, Article ID 5080332, 16 pages, 2024. https://doi.org/10.1155/2024/5080332

[8] M. Goryunov, A. Matskevich and D. Rybolovlev, "Synthesis of a Machine Learning Model for Detecting Computer Attacks Based on the CICIDS2017 Dataset", Proceedings of the Institute for System Programming of the RAS, vol. 32, no. 5, pp. 81-94, 2020. Available: 10.15514/diasporas-2020-32 (5)-6 [Accessed 12 September 2022].

[9] Nancy Shaar, Mohammad Alshraideh, Lara Shboul & Iyad AlDa jani (2023) Decision support system (DSS) for traffic prediction and building a dynamic internet community using Netnography technology in the city of Amman, Journal of Experimental & Theoretical Artificial Intelligence, DOI: 10.1080/0952813X.2023.2165716

[10] Kurniabudi, D. Stiawan, Darmawijoyo, M. Bin Idris, A. Bamhdi and R. Budiarto, "CICIDS-2017 Dataset Feature Analysis With Information Gain for Anomaly Detection", IEEE Access, vol. 8, pp. 132911-132921, 2020. Available: 10.1109/access.2020.3009843 [Accessed 12 September 2022].

[11] G. Breda and M. Kiss, "Overview of Information Security Standards in the Field of Special Protected Industry 4.0 Areas & Industrial Security", Procedia Manufacturing, vol. 46, pp. 580-590, 2020. Available: 10.1016/j.promfg.2020.03.084 [Accessed 10 September 2022].

[12] R. Shree and K. Sandhu, "A Multi-Objective Decision Assistance System for Selecting Security Controls Based On Simulation," 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2023, pp. 1754-1756, doi: 10.1109/ICACITE57410.2023.10183342.

[13] F. Alkhudhayr, S. Alfarraj, B. Aljameeli and S. Elkhdiri, "Information Security: A Review of Information Security Issues and Techniques", 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), 2019. Available: 10.1109/cais.2019.8769504 [Accessed 10 September 2022].

[14] S. Mishra and V. K. Chaurasiya, "Ethereal Networks and Honeypots for Breach Detection," 2022 International Conference on Machine Learning, Computer Systems and Security (MLCSS), Bhubaneswar, India, 2022, pp. 309-317, doi: 10.1109/MLCSS57186.2022.00063.

[15] J. Ning, J. Xu, K. Liang, F. Zhang and E. Chang, "Passive Attacks Against Searchable Encryption", IEEE Transactions on Information Forensics and Security, vol. 14, no. 3, pp. 789-802, 2019. Available: 10.1109/tifs.2018.2866321 [Accessed 10 September 2022].

[16] V. Kumar, D. Sinha, A. Das, S. Pandey and R. Goswami, "An integrated rule-based intrusion detection system: analysis on UNSW-NB15 data set and the real-time online dataset", Cluster Computing, vol. 23, no. 2, pp. 1397-1418, 2019. Available: 10.1007/s10586-019-03008-x [Accessed 11 September 2022].

[17] G. Amaizu, C. Nwakanma, J. Lee and D. Kim, "Investigating Network Intrusion Detection Datasets Using Machine Learning", 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020. Available: 10.1109/ictc49870.2020.9289329 [Accessed 11 September 2022].

[18] Alfayoumi, Bayan et al. "Analyzing the Sentiments of Jordanian Students Towards Online Education in the Higher Education Institutions." *International Journal of Advanced Computer Science and Applications* (2021): n. pag.

[19] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei and F. Sabrina, "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset", IEEE Access, vol. 9, pp. 140136-140146, 2021. Available: 10.1109/access.2021.3116612 [Accessed 11 September 2022].

[20] K. Dr.R.Venkatesh, "Network Anomaly Detection for NSL-KDD Dataset Using Deep Learning", INFORMATION TECHNOLOGY IN INDUSTRY, vol. 9, no. 2, pp. 821-827, 2021. Available: 10.17762/itii.v9i2.419 [Accessed 11 September 2022].

[21] A. Ferriyan, A. Thamrin, K. Takeda and J. Murai, "Generating Network Intrusion Detection Dataset Based on Real and Encrypted Synthetic Attack Traffic", Applied Sciences, vol. 11, no. 17, p. 7868, 2021. Available: 10.3390/app11177868 [Accessed 12 September 2022].

[22] I. Qaddara, "APPLYING MACHINE LEARNING TECHNIQUES ON CYBER SECURITY DATASETS: DETECTING CYBER ATTACKS". Harbin Gongye Daxue Xuebao/Journal of Harbin Institute of Technology, vol. 54, no. 7, pp. 95-110, 2022.

[23] Ketsbaia, L., Issac, B., & Chen, X. (2020, December). Detection of hate tweets using machine learning and deep learning. In 2020 IEEE 19th International Conference on Trust, Security and

Privacy in Computing and Communications (TrustCom) (pp. 751-758). IEEE.

[24] Luo, Y., Zhang, X., Hua, J., & Shen, W. (2021, August). Multi-featured Cyber-bullying Detection Based on Deep Learning. In 2021, the 16th International Conference on Computer Science & Education (ICCSE) (pp. 746-751). IEEE.

[25] Mahlangu, T., & Tu, C. (2019, November). Deep learning cyber-bullying detection using stacked embeddings approach. In 2019, the 6th International Conference on Soft Computing & Machine Intelligence (ISCMI) (pp. 45-49). IEEE.

[26] P. P.G and D. E. D, "Design of a Hybrid Intrusion Detection System using Snort and Hadoop," Int. J. Comput. Appl., vol. 73, no. 10, pp. 5–10, Jul. 2013, doi: 10.5120/12775-9226.

[27] S. Potluri, N. F. Henry, and C. Diedrich, "Evaluation of hybrid deep learning techniques for ensuring security in networked control systems," in 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Sep. 2017, pp. 1–8. doi: 10.1109/ETFA.2017.8247662.

[28] X. Zhang, P. Zhu, J. Tian, and J. Zhang, "An effective semi-supervised model for intrusion detection using feature selection based LapSVM," in 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), Jul. 2017, pp. 283–286. doi: 10.1109/CITS.2017.8035323.

[29] Alfayoumi B., Alshraideh M., Martin Leiner, Iyad Muhsen Aldajani.(2021). Machine Learning Predictions For The Advancement Of Online Education in The Higher Education Institutions in Jordan, Journal of Hunan University Natural Sciences, 48(9)

[30] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Sep. 2017, pp. 1222–1228. doi: 10.1109/ICACCI.2017.8126009.

[31] M. M. ALkharabsheh, M. Alshraideh, and I. Salah, "Enhancing Cybercrime Deterrence with Artificial Intelligence," Int. J. Adv. Netw. Appl., vol. 15, no. 04, pp. 6015–6027, 2023, doi: 10.35444/IJANA.2024.15404.

[32] H. Harahsheh, M. Alshraideh, S. Al-Sharaeh, and R. Al-Sayyed, "Improving Classification Performance for Malware Detection Using Genetic Programming Feature Selection Techniques," J. Appl. Secure. Res., vol. 18, no. 3, pp. 627–647, Jul. 2023, doi: 10.1080/19361610.2022.2067459.

[33] H. Harahsheh, M. Shraideh, and S. Sharaeh, "Performance of Malware Detection Classifier Using Genetic Programming in Feature Selection," Informatica, vol. 45, no. 4, Dec. 2021, doi: 10.31449/inf.v45i4.3819.

[34] M. Alshraideh, A. A.-J. Abu-Zayed, M. Leiner, and I. M. AlDajani, "Beyond the Scoreboard: A Machine Learning Investigation of Online Games' Influence on Jordanian University Students' Grades," Appl. Comput. Intell. Soft Comput., vol. 2024, pp. 1–11, Jan. 2024, doi 10.1155/2024/1337725.