International Journal of Computers and Informatics (IJCI)

Vol. (2), No. (8)

المجلة الدولية للحاسبات والمعلوماتية

الإصدار (2)، العدد (8)

December 2023

"Intrusion Detection in IoT Networks Using Machine Learning Techniques"

Wasan Abdallah Alawsi
Collage of Science, Al Qadisayah University, Al Dewanyah, Iraq
wasan.alawsi@qu.edu.iq

Abstract:

The advent of the IoT has undeniably transformed the manner in which we connect with the world innovative solutions across a multitude of sectors. However, the proliferation of IoT applications has brought to the fore a complex and multifaceted problem that warrants immediate attention: the security and energy efficiency challenges within IoT networks. As the number of IoT devices continues to surge and their applications diversify, the demand for ubiquitous and secure data access has never been greater. These applications, ranging from healthcare monitoring to industrial automation, require not only real-time data processing but also the assurance of data security and privacy. So, the research explores the machine learning techniques application for intrusion detection in IoT networks, aiming to enhance security in the rapidly evolving landscape of interconnected devices. The research emphasizes the significance of proactive measures in safeguarding IoT ecosystems and highlights the potential of machine learning to detect and mitigate intrusions effectively. In this research, a model was developed for detecting anomalies in Internet of Things networks using artificial neural networks. The research aims to develop and evaluate an effective intrusion detection model for IoT networks by leveraging machine learning techniques. The purpose is to enhance the security posture of interconnected devices, addressing the unique challenges posed by the dynamic and heterogeneous nature of IoT environments. The study tries to

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

contribute important experiences and functional answers to reinforce the strength of IoT networks against developing digital dangers.

Keywords: IoT, IoMT, Anomalies in IoT Networks, Artificial Neural Networks, Intrusion Detection**.**

1- Introduction

The coming of the IoT has introduced a new connectivity era, transforming the way devices interact and communicate in various domains such as healthcare, smart cities, and industrial automation [1]. As billions of devices become interlinked, creating a vast and intricate network, the potential for innovative applications has grown exponentially. However, this surge in connectivity has concurrently introduced unprecedented security challenges, as the heterogeneity and dynamic nature of IoT environments create vulnerabilities that can be exploited by malicious actors. In light of these challenges, this research endeavors to address a critical aspect of IoT security—intrusion detection—by harnessing the power of machine learning techniques [2].

The inherent characteristics of IoT networks, characterized by a multitude of devices with diverse capabilities, limited resources, and often deployed in uncontrolled environments, amplify the complexity of ensuring robust security measures [3]. Traditional security mechanisms, while effective in conventional networks, often fall short when applied to the unique challenges posed by the IoT paradigm [4]. Intrusion detection emerges as a crucial line of defense, aiming not only to identify and thwart potential threats but also to adapt dynamically to the ever-evolving nature of cyber-attacks in IoT ecosystems.

Machine learning, with its capacity to discern patterns, anomalies, and trends from vast datasets, presents a promising avenue for enhancing intrusion detection

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

capabilities in IoT networks [5]. This research explores the machine learning techniques, delving into their application for detecting intrusions in real-time within the intricate fabric of IoT. By leveraging the inherent capabilities of machine learning algorithms.

The significance of this research extends beyond theoretical exploration, as the outcomes hold the potential to shape the future landscape of IoT security. As society increasingly relies on interconnected devices for critical functions, ranging from healthcare monitoring to industrial control systems, the need for robust security measures becomes paramount. By establishing a foundation for effective intrusion detection in IoT networks, this research endeavors to mitigate the risks associated with unauthorized access, data breaches, and potential disruptions to the seamless operation of IoT-enabled systems.

2- Literature Reviews

The proliferation of IoT devices introduces a myriad of security challenges stemming from their heterogeneity, resource constraints, and often open deployment environments. Previous studies (Al-Fuqaha et al., 2015; Roman et al., 2013) [6] [7] highlight the diverse attack surfaces presented by IoT devices, emphasizing the need for tailored security mechanisms to address specific threats such as unauthorized access, data manipulation, and denial-of-service attacks.

Intrusion detection is a pivotal component of cybersecurity, aiming to identify and respond to malicious activities promptly. Traditional intrusion detection systems (IDS) have been adapted for IoT; however, their efficacy is often hindered by the unique characteristics of IoT environments. These researches (Miettinen et al., 2017; Kolias et al., 2017) [8] [9] explore the challenges associated with designing intrusion

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

detection systems specifically tailored to the diverse and dynamic nature of IoT networks.

Machine learning techniques have gained prominence in enhancing the capabilities of intrusion detection systems. Studies (Wang et al., 2017; Abdullah et al., 2020) [10] [11] demonstrate the effectiveness of machine learning algorithms in discerning patterns and anomalies from large datasets, providing a dynamic and adaptive approach to intrusion detection. The application of machine learning in traditional networks has shown promising results, motivating its exploration in IoT security. While the machine-learning integration in intrusion detection for IoT networks presents promising opportunities, challenges such as scalability, resource constraints, and the interpretability of machine learning models need careful consideration. Understanding these challenges is crucial for developing effective and practical solutions that can be seamlessly integrated into the diverse IoT landscape.

The time-sensitive nature of IoT applications demands real-time intrusion detection capabilities. Ensuring low-latency responses to security threats is vital, especially in critical applications such as healthcare and industrial control. Previous studies (Mahmood et al., 2019; Javed et al., 2021) [12] [13] have explored the challenges of achieving real-time intrusion detection in resource-constrained IoT devices. Addressing these challenges requires a nuanced understanding of the trade-offs between accuracy, speed, and resource utilization in the context of machine learning-based intrusion detection.

As IoT devices collect and process vast amounts of personal and sensitive data, privacy concerns become paramount. Balancing the need for effective intrusion detection with privacy and ethical considerations is a critical aspect of IoT security. Examining the ethical implications of deploying machine learning algorithms for intrusion detection in IoT ecosystems (Dwork et al., 2017; Mittal, 2019) [14] [15]

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

adds depth to the discussion, emphasizing the importance of responsible and transparent practices in securing IoT networks.

This survey [16] offers a comprehensive overview of intrusion detection systems (IDS) tailored for the Internet of Things (IoT) landscape. It explores the challenges associated with securing IoT environments and provides a detailed analysis of existing IDS solutions. The study categorizes IDS approaches into signature-based, anomaly-based, and hybrid methods, discussing their strengths and limitations. The authors emphasize the necessity of adaptive and scalable IDS solutions for the diverse and dynamic nature of IoT networks.

3- Methodology

Artificial neural networks are considered as a computational system consisting of a number of interconnected processing units and are characterized by their dynamic and parallel nature in processing the data entering them [17]. The artificial neuron is the building unit of the artificial neural network. The cell consists of an arithmetic unit with multiple inputs and an output signal, and for each output signal, there is a weight that modifies the input signal and stimulates the cell to produce a response signal when the value is positive or suppress it when its value is negative [18].

An artificial neural network (ANN) and a biological neural network (BNN) share the same basic principle of information processing, but they are fundamentally different in their implementation and purpose [19].

The following figure (1) represents the architecture of the ANN:

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

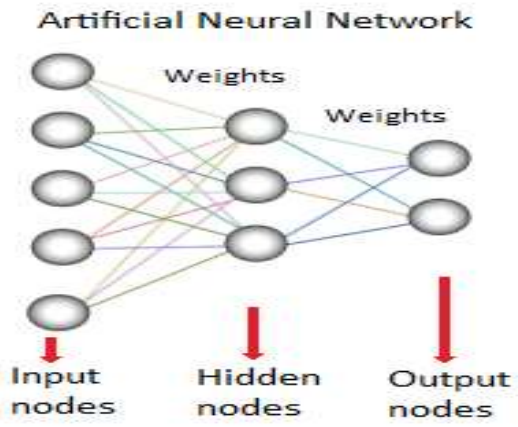

Figure (1): Architecture of the ANN

The input nodes process information in a digital format, represented by activation values. These values, assigned to each node, indicate the degree of activation. As the information travels through the network, connection strength, inhibition/excitation functions, and transmission affect how activation values move from one node to another. Each node gathers received activation values, adjusting them based on its transfer function. This process continues through hidden layers until reaching output nodes, which convey meaningful information externally. The network learns by backpropagating the difference between expected and actual values, adjusting weights accordingly [19].

In the context of Artificial Neural Networks (ANNs), a transfer function (also known as an activation function) is a mathematical function applied to the output of each neuron (or node) in a neural network layer. The primary purpose of a transfer function is to introduce non-linearity into the network, allowing the neural network to model complex relationships in the data it is learning from [20].

Figure (2) shows forward propagation and back propagation networks.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

International Journal of Computers and Informatics (IJCI)

Vol. (2), No. (8)

IJCI

December 2023

المجلة الدولية للحاسبات والمعلوماتية

الإصدار (2)، العدد (8)

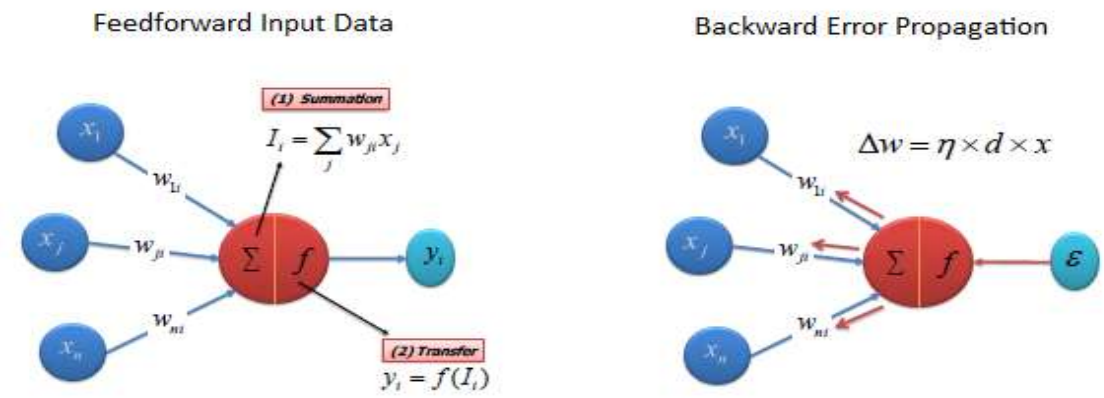

Figure (2): Forward Propagation and Back Propagation Networks

The proposed methodology is based on building a multilayer perceptron neural network to detect intrusion in Internet of Things networks. The proposed methodology differs from previous methodologies in determining the type of attack, as most studies that dealt with the subject of intrusion detection depend on detecting the attack in general, that is, either 0 there is no attack or 1 there is an attack, while in our proposed methodology we will rely on determining the type of attack, and this enables specialists Dealing with every type of attack in the appropriate manner to prevent or stop it. Figure (3) shows the general framework of this work.

3-1 Data Collection:

UNSW-NB15 is a network intrusion dataset. The dataset contains raw network packets. The number of records in the training set is 175,341 records and the testing set is 82,332 records from the different types, attack and normal. It contains nine different attacks, namely:

1) Fuzzers: Trying to disrupt a program or network by supplying it with randomly generated data.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

2) Analysis: It encompasses various attacks such as port scanning, spam, and HTML file penetrations.
3) Backdoors: A method wherein a computer or its data is accessed clandestinely by circumventing the system's security mechanisms.
4) DOS: A harmful effort to render a server or network resource inaccessible to users, typically achieved by temporarily disrupting or suspending the services of an Internet-connected host.
5) Exploit: The assailant is aware of a security issue in an operating system or software and exploits this knowledge to take advantage of the vulnerability.

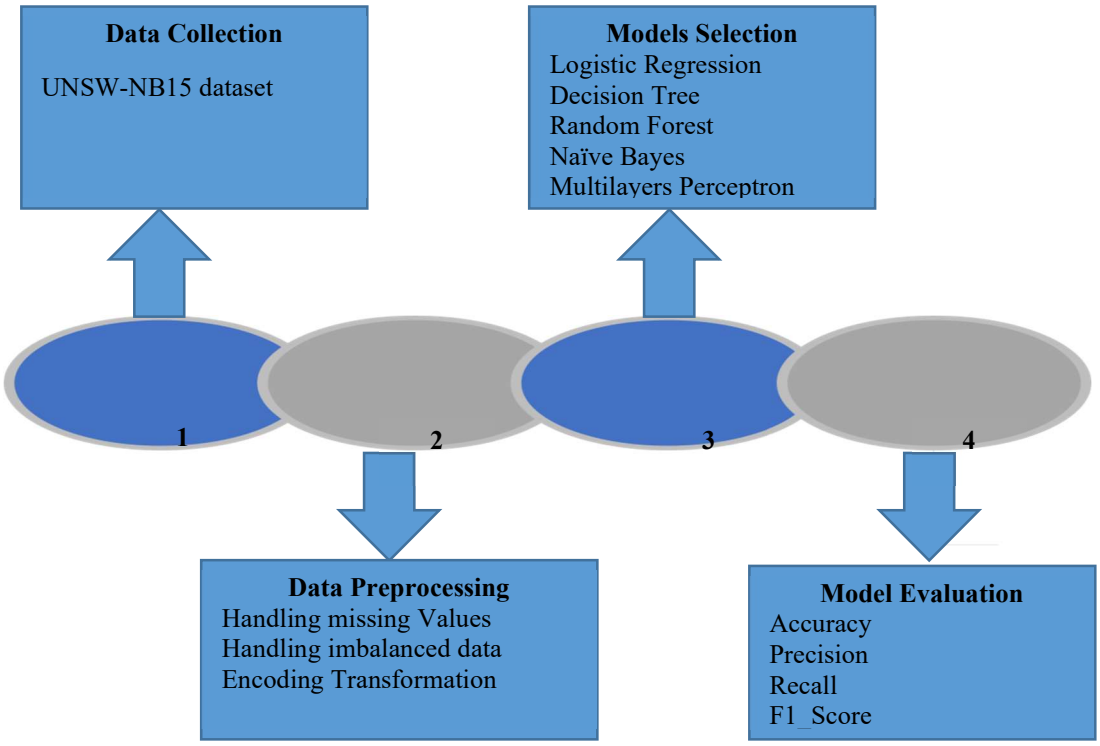

Figure (3): The General Framework of this Work

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

6) Generic: This technique is effective against all block ciphers (with a specified block and key size), without taking into account the structure of the block cipher.
7) Reconnaissance: Encompasses all offensive maneuvers capable of simulating information-gathering attacks.
8) Shell code: A concise code snippet employed as the payload in exploiting a software vulnerability.
9) Worms: The assailant duplicates its presence to propagate to other computers, frequently utilizing a computer network for dissemination and exploiting security weaknesses in the target system for access.

3-2 Data Preprocessing:

1. **Handling missing values:** Due to various factors, datasets frequently include missing values [21]. These datasets pose challenges for machine learning estimators, which expect numerical values and meaningful content in all array elements. While a common approach is to eliminate rows or columns with missing values, this results in data loss, potentially sacrificing valuable information. A more effective strategy involves utilizing available data for imputing missing values. The SimpleImputer python is a class that provides a straightforward way to handle missing values in a dataset. It allows you to replace missing values with a specified strategy, such as mean, median, most frequent, or a constant value. In our proposed methodology, we relied on calculating missing values by taking the mean value of the available data based on class label.
2. **Handling imbalanced dataset:** Addressing imbalanced datasets often involves the widely adopted technique of resampling data, which broadly falls into two categories: i) Undersampling and ii) Oversampling [23]. Typically,

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

oversampling is favored over undersampling methods because undersampling involves discarding instances that may contain valuable information. We used SMOTE method which is an oversampling method that produces synthetic samples for the minority class [24]. Unlike random oversampling, it addresses overfitting issues by concentrating on the feature space. This algorithm generates new instances by interpolating between closely positioned positive instances [25].

3. **Encoding Transformation:** In ML projects, it's common to encounter datasets containing diverse categorical columns. Some columns may feature elements categorized as ordinal variables, such as an "income level" column with values like low, medium, or high. In such instances, a straightforward approach is to substitute these categorical labels with numerical equivalents, like 1 for 'low', 2 for 'medium', and 3 for 'high'. We used “Label Encoding method” which is employed to convert categorical columns into numerical format, enabling compatibility with ML models that exclusively handle numerical data.

3-3 Model Selection:

1) **K-Nearest Neighbors (KNN):** K-Nearest Neighbors (KNN) is a supervised machine learning algorithm used for classification and regression tasks. In KNN, the prediction for a new data point is determined by the majority class (for classification) or the average of the neighboring points' values (for regression) among its k nearest neighbors in the feature space. The distance metric, such as Euclidean or Manhattan distance, is typically used to measure the proximity between data points. KNN is a non-parametric and instance-based algorithm, meaning it doesn't make assumptions about the underlying data distribution and relies on the entire training dataset for predictions.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

2) **Random Forest:** It is an ensemble learning algorithm widely used for both classification and regression tasks in machine learning. It is an ensemble of decision trees, where each tree is constructed using a random subset of the training data and a random subset of features for each split. The final prediction in classification is determined by a majority vote among the trees, and in regression, it's the average prediction of individual trees. This ensemble approach improves generalization and robustness by reducing overfitting and capturing complex relationships within the data. It is known for its flexibility, scalability, and effectiveness across various types of datasets.
3) **Multi-Layer Perceptron:** The multi-layer perceptron, commonly referred to as MLP, comprises fully connected dense layers that enable the transformation of input dimensions to the desired form. Essentially, it is a neural network with multiple layers. Constructing a neural network involves connecting neurons so that the outputs of some neurons become inputs for others. A typical multi-layer perceptron includes an input layer with one neuron per input, an output layer with a single node for each output, and the flexibility to incorporate any number of hidden layers, each accommodating any desired number of nodes.

In this research, to improve forecast accuracy, we have developed an artificial neural network architecture of the type MLP. the experimental approach was adopted to obtain the structure of the proposed neural network. In the beginning, one layer consisting of ten neurons was relied upon and the results extracted, then the number of neurons was increased by ten by ten until we reached the number of neurons of 100, after which the accuracy began to decrease. A second hidden layer was added and the same scenario as the first layer was used, so the best accuracy we got was when the number of neurons was 125 neurons. When adding a third layer, we noticed a decrease in the classification accuracy, so we deleted the third layer. So the result

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

that achieved the best classification accuracy was a neural network architecture with two-hidden-layers:

- The first layer has 100 neurons.
- The second layer has 125 neurons.

We used the ADAM training algorithm which refers to a stochastic gradient-based optimizer proposed in [26]. ADAM works pretty well on relatively large datasets (with thousands of training samples or more) in terms of both training time and validation score.

In order to utilize the stochastic gradient descent with the errors backpropagation for training ANN, we need the activation function. This function should be providing more sensitivity to the activation sum input and avoid easy saturation so, we will utilize RELU [27].

Based on the above mentioned we used ReLU (Rectified Linear Unit) transfer function. ReLU (Rectified Linear Unit) is the most recently function has become popular. It is neither smooth nor bounded, but works well in applications that have very large numbers of data.

ReLU is one of the most commonly used activation functions in deep learning, particularly in the hidden layers of deep neural networks. It is well-suited for tasks like image recognition, object detection, and natural language processing, where complex and hierarchical patterns need to be learned from the data.

4) **Naive Bayes:** It is a probabilistic machine learning algorithm commonly used for classification tasks. It is a simple and fast algorithm based on Bayes' theorem. It assumes that features are conditionally independent given the class label, which is why it's called "naive." Despite this simplifying assumption, Naive Bayes often performs well in practice and is particularly useful for text

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

classification and spam filtering. It calculates the probability of each class given a set of input features and selects the class with the highest probability as the predicted class. It is efficient and requires a relatively small amount of training data to make predictions.

5) **Logistic Regression:** It is a statistical method used for binary classification, predicting the probability that an instance belongs to a particular class. Despite its name, logistic regression is employed for classification rather than regression tasks. It models the probability of a binary outcome using the logistic function (sigmoid function). It calculates a weighted sum of input features, and the result is transformed into a probability between 0 and 1. The decision boundary is set to classify instances based on whether the predicted probability is above or below a threshold (commonly 0.5). Logistic Regression is widely used due to its simplicity, interpretability, and effectiveness in scenarios where the relationship between features and the binary outcome is approximately linear.

3-4 Model Evaluation:

We used hold-out method is a technique in machine learning where the dataset is divided into two subsets: one for training the model and another for evaluating its performance. We divided dataset to 80%, is used for training, while the remaining 20% is reserved for testing. This approach allows the model to learn from one subset and then assess its performance on unseen data, helping to gauge its generalization capabilities. The hold-out method is a straightforward and commonly employed strategy for model validation and performance assessment [28].

A confusion matrix is used for evaluating the classification performance in ML, [29]. It provides a summary of the predictions by compare them to the actual true labels

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

of the data. The matrix is particularly useful for tasks with multiple classes or categories, allowing us to understand how well the model is classifying instances into each class.

A confusion matrix (CM) is a square matrix representing the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions made by a classification model for each class in the dataset. Each row of the matrix corresponds to the actual class, and each column corresponds to the predicted class [30]. Figure (4) shows the architecture of the confusion matrix:

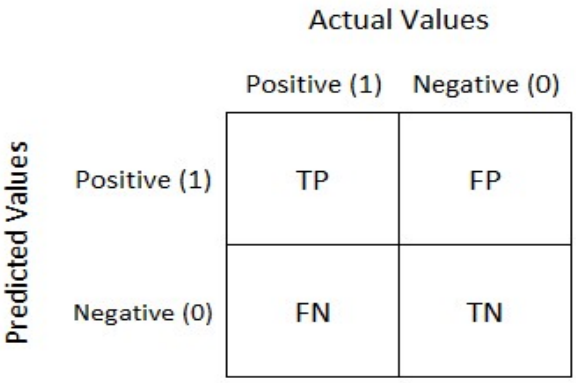

Figure (4): Confusion Matrix Architecture

It is extremely useful for measuring Recall, Precision and Accuracy.

- TP (True Positive): It refers that we predicted positive and it is true.
- TN (True Negative): It refers that we predicted negative and it is true.
- FP (False Positive): It is type one error and refers that predicted positive and it is false.
- FN (False Negative): It is type two error and refers that predicted negative and it is false.

These parameters are used to calculate recall, precision and accuracy.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

$$\text{Re } call = \frac{TP}{TP + FN} \quad \text{…….. (1)}$$

In equation (1), it can be discovered as "for all classes which are positive, how many correctly predicted classes [31]. Recall should be high as possible.

$$\text{Pr}ecision = \frac{TP}{TP + FP} \quad \text{……..(2)}$$

In equation (2), it can be discovered as "from all predicted positive classes, how many are positive actually [32]. Precision should be high as possible.

The accuracy is defined as the percentage of true classifications that a trained ML model achieves [33]. The equation (3) of accuracy is given as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{…….(3)}$$

In equation (4) The F1 score is a metric that combines precision and recall to provide a balanced measure of a model's performance. It's calculated as the harmonic mean of precision and recall, providing a single value that considers both false positives and false negatives. The formula is:

$$F1\_Score = 2 \times \frac{\text{Pr}ecision \times \text{Re}call}{\text{Pr}ecision + \text{Re}call} \quad \text{…….. (4)}$$

4- Results and Discussions

In this section, we review the confusion matrices resulting from the simulation process for each of the algorithms RF, KNN, NB, LR, proposed model. The results of these algorithms will then be compared with the performance metrics mentioned in Section 3.4.

Table 1 shows the confusion matrix (CM) resulting from applying the RF algorithm to the test data. The colored values within the confusion matrix, represented by the elements of the main diagonal, show the number of predictions that the algorithm

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

made correctly for each type of attack, while the elements that are outside the main diagonal represent the algorithm's incorrect predictions. For example, for the Analysis attack type, the algorithm correctly predicted 7,349 samples while incorrectly predicting 4 samples as a DOS attack and 6 samples as a Reconnaissance attack.

Table 1: CM Resulting from Applying the RF Algorithm to The Test Data

	Analysis	Backdoors	DOS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shell code	Worms
Analysis	7349	0	4	0	0	0	0	6	0	0
Backdoors	0	6694	356	134	111	0	0	47	63	5
DOS	0	939	4811	1081	103	0	0	202	184	26
Exploit	1	651	1371	4409	220	0	0	389	177	138
Fuzzers	0	1217	380	54	5574	0	0	76	45	0
Generic	0	6	9	173	31	7260	0	10	29	0
Normal	0	0	4	15	140	0	7200	61	28	0
Reconnaissance	0	46	597	33	16	0	5	6553	158	13
Shell code	0	0	0	0	0	0	0	1	7444	0
Worms	0	0	0	0	0	0	0	0	100	7251

In the same way, we will review below the CM values for each of the NB, LR, and KNN algorithms, the proposed model, from table (2) we shows the CM resulting from applying the NB algorithm to the test data.
Table 3 shows the CM resulting from applying the LR algorithm to the test data.
Table 4 shows the CM resulting from applying the KNN algorithm to the test data.
Table 5 shows the CM resulting from applying the proposed model to the test data.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

International Journal of Computers and Informatics (IJCI)

Vol. (2), No. (8)

IJCI

المجلة الدولية للحاسبات والمعلوماتية

الإصدار (2)، العدد (8)

December 2023

Table 2: CM Resulting from Applying the NB Algorithm to The Test Data

	Analysis	Backdoors	DOS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shell code	Worms
Analysis	6532	179	11	578	59	0	0	0	0	0
Backdoors	6069	438	142	158	593	10	0	0	0	0
DOS	4305	647	266	1445	656	15	8	0	0	4
Exploit	1446	464	117	3540	1164	56	20	450	66	33
Fuzzers	220	1542	20	219	4106	540	0	370	329	0
Generic	0	10	1	51	32	7263	27	21	31	82
Normal	0	45	0	70	888	429	3815	866	1066	269
Reconnaissance	0	79	0	0	48	351	34	1658	5236	15
Shell code	0	0	0	0	4	94	0	35	7311	1
Worms	0	0	0	0	113	138	0	181	1476	5443

Table 3: CM Resulting from Applying the LR Algorithm to the Test Data

	Analysis	Backdoors	DOS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shell code	Worms
Analysis	462	4632	1633	0	15	0	588	0	0	29
Backdoors	532	5469	504	13	113	0	295	134	334	16
DOS	134	974	4009	657	106	87	686	48	152	403
Exploit	72	642	1534	2037	323	38	1284	215	208	1003
Fuzzers	508	1819	502	303	1007	4	711	544	1140	808
Generic	29	2753	5	47	9	4517	118	6	8	26
Normal	67	891	51	577	777	141	4000	202	472	270
Reconnaissance	435	1061	1555	42	347	446	227	2684	343	281
Shell code	846	1358	1376	67	372	174	259	426	2270	297
Worms	146	140	529	1117	215	188	985	0	27	4004

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

International Journal of Computers and Informatics (IJCI)

Vol. (2), No. (8)

IJCI

المجلة الدولية للحاسبات والمعلوماتية

الإصدار (2)، العدد (8)

December 2023

Table 4: CM Resulting from Applying the KNN Algorithm to the Test Data

	Analysis	Backdoors	DOS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shell code	Worms
Analysis	5022	1987	215	43	54	0	0	29	9	0
Backdoors	2337	4786	102	28	102	0	0	30	24	1
DOS	428	522	5797	326	121	13	0	84	37	18
Exploit	244	391	1134	4864	304	10	11	229	96	73
Fuzzers	274	422	275	124	5846	106	6	166	104	23
Generic	13	6	49	111	64	7181	2	58	26	8
Normal	35	8	21	66	293	3	6918	73	13	18
Reconnaissance	92	78	354	87	86	69	1	6503	141	10
Shell code	44	35	24	6	26	21	0	175	7110	4
Worms	4	1	13	4	8	4	8	5	26	7278

Table 5 : CM Resulting from Applying the Proposed Model to The Test Data

	Analysis	Backdoors	DOS	Exploit	Fuzzers	Generic	Normal	Reconnaissance	Shell code	Worms
Analysis	7415	48	3	0	0	0	0	0	0	0
Backdoors	6	7200	161	115	0	0	0	0	0	0
DOS	10	90	6464	737	42	1	4	0	0	0
Exploit	2	8	136	6836	283	15	13	8	0	0
Fuzzers	0	2	1	82	7357	7	13	28	0	0
Generic	0	0	1	40	144	7224	1	8	0	0
Normal	0	0	0	22	105	3	6980	162	67	28
Reconnaissance	0	0	0	8	1	4	15	7412	21	23
Shell code	0	0	0	0	0	0	0	420	6863	25
Worms	0	0	0	0	0	0	0	0	5	7331

The measurement parameters mentioned in paragraph 4.3 represent a summary of the confusion matrix and show the performance of each algorithm based on the resulting confusion matrix.

The following figure 5 shows a comparison between the previous algorithms and the proposed methodology in terms of: precision, recall, F1_Score and accuracy.

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

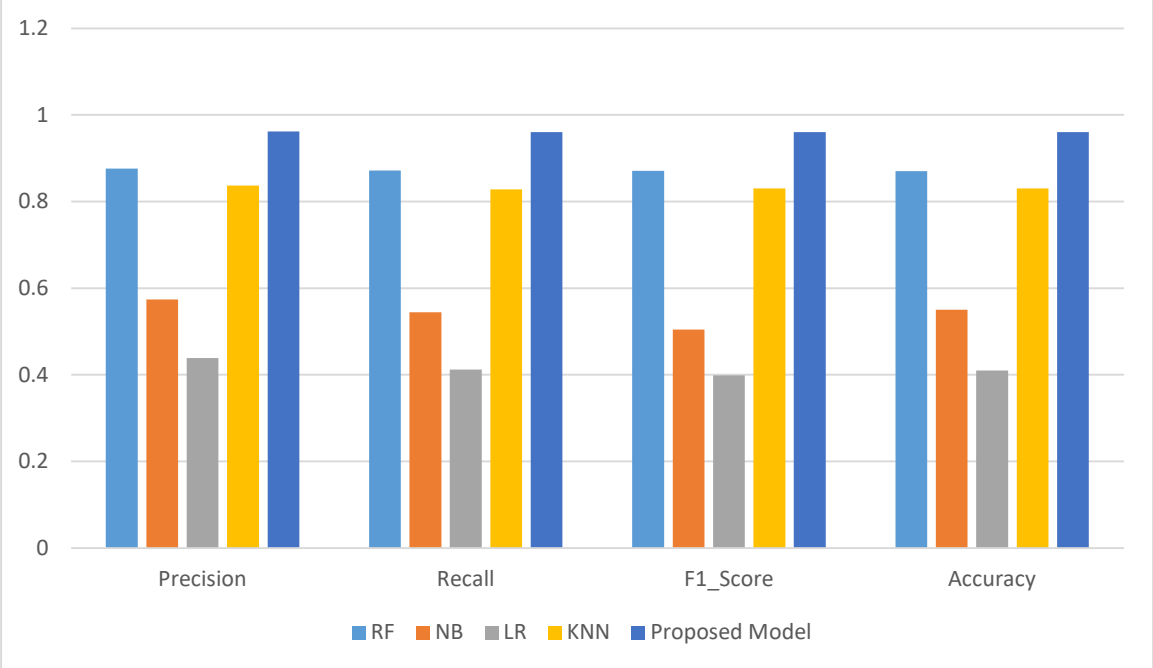

Figure (5): Shows A Comparison Between The Previous Algorithms and The Proposed Methodology

We note from figure (5) that:

1) Precision:

- The proposed model achieved the highest precision (0.962), indicating a low rate of false positives. This suggests that when the model predicts an intrusion, it is highly likely to be correct.
- Random Forest also performed well in precision (0.8758), indicating a good ability to avoid false positives.

A high precision value for the proposed model (0.962) is crucial in scenarios where false positives have significant consequences. This indicates that the

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

model is adept at minimizing the instances of incorrectly classifying normal behavior as an intrusion.

2) Recall:
   - The proposed model again excelled in recall (0.9603), demonstrating a high ability to correctly identify intrusions. This is crucial for comprehensive threat detection.
   - Random Forest also showed strong recall (0.8716), indicating its effectiveness in capturing actual positive instances.

   The high recall of the proposed model (0.9603) is essential for capturing a substantial proportion of actual intrusions. This suggests that the model is effective in identifying malicious activities, minimizing false negatives.

3) F1 Score:
   - The F1 score balances precision and recall. The proposed model achieved a high F1 score (0.9605), indicating a good overall balance between precision and recall.
   - Random Forest also demonstrated a competitive F1 score (0.8707).

   The balanced F1 score (0.9605) of the proposed model is a positive sign, indicating a strong compromise between precision and recall. This balance is crucial in situations where both false positives and false negatives need to be minimized.

4) Accuracy:
   - The proposed model outperformed all other models in terms of accuracy (0.96), showcasing its effectiveness in overall classification accuracy.
   - Random Forest had the second-highest accuracy (0.87).

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

The overall accuracy of the proposed model (0.96) is very good, highlighting its ability to correctly classify instances across all classes. However, it's essential to be cautious about overemphasizing accuracy, especially in imbalanced datasets.

In summary, RF demonstrates solid performance across all metrics, particularly in precision (0.8758) and recall (0.8716). This suggests its effectiveness in minimizing false positives and capturing actual intrusions. While not surpassing your proposed model, RF's competitive results make it a noteworthy alternative, especially considering its ensemble nature that often provides robustness. NB shows lower performance compared to other models, with lower precision (0.5736), recall (0.5447), and F1 score (0.5047). This indicates limitations in accurately classifying both normal and intrusive instances. LR exhibits moderate performance, falling between RF and NB. Its precision (0.4387) and recall (0.4121) suggest a trade-off between minimizing false positives and capturing intrusions. KNN performs decently, with precision (0.8371) and recall (0.8279) values comparable to Random Forest. Its results indicate effectiveness in classifying instances, though not reaching the level of your proposed model.

5- Conclusions and Recommendations

In this paper, we explored the application of machine learning techniques for intrusion detection in IoT networks, aiming to enhance security in the rapidly evolving landscape of interconnected devices. We developed a model for detecting anomalies in Internet of Things networks using artificial neural networks. The research on intrusion detection in IoT networks employing machine learning techniques has yielded compelling insights. The proposed multilayer perceptron neural network emerged as the frontrunner, showcasing superior precision, recall, F1

IJCI, VSRP Publishing, UK
https://ijci.vsrp.co.uk
E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

score, and accuracy compared to Random Forest, Naive Bayes, Logistic Regression, and k-Nearest Neighbors. The model's exceptional ability to minimize false positives and effectively capture intrusions positions it as a promising solution for real-world deployment in IoT security.

Moving forward, it is recommended to explore ensemble approaches, considering the competitive performance of Random Forest, to enhance the system's robustness. While recognizing the complexity of the proposed neural network, efforts should be made to balance interpretability without compromising its efficacy. Continuous evaluation, adaptation to evolving threats, and collaboration with cyber security experts will be integral for the sustained success and relevance of the intrusion detection system in the dynamic landscape of IoT security.

References:

[1] Qadri, Y. A., Nauman, A., Zikria, Y. B., Vasilakos, A. V., & Kim, S. W. (2020). The future of healthcare internet of things: a survey of emerging technologies. *IEEE Communications Surveys & Tutorials*, *22*(2), 1121-1167.

[2] Yang, N., Wang, L., Geraci, G., Elkashlan, M., Yuan, J., & Di Renzo, M. (2015). Safeguarding 5G wireless communication networks using physical layer security. *IEEE Communications Magazine*, *53*(4), 20-27.

[3] Berger, C., Eichhammer, P., Reiser, H. P., Domaschka, J., Hauck, F. J., & Habiger, G. (2021). A survey on resilience in the iot: Taxonomy, classification, and discussion of resilience mechanisms. *ACM Computing Surveys (CSUR)*, *54*(7), 1-39.

[4] Singh, A., Payal, A., & Bharti, S. (2019). A walkthrough of the emerging IoT paradigm: Visualizing inside functionalities, key features, and open issues. *Journal of Network and Computer Applications*, *143*, 111-151.

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

[5] Da Costa, K. A., Papa, J. P., Lisboa, C. O., Munoz, R., & de Albuquerque, V. H. C. (2019). Internet of Things: A survey on machine learning-based intrusion detection approaches. *Computer Networks*, *151*, 147-157.

[6] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, *17*(4), 2347-2376.

[7] Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed internet of things. *Computer networks*, *57*(10), 2266-2279.

[8] Miettinen, M., Marchal, S., Hafeez, I., Asokan, N., Sadeghi, A. R., & Tarkoma, S. (2017, June). Iot sentinel: Automated device-type identification for security enforcement in iot. In *2017 IEEE 37th international conference on distributed computing systems (ICDCS)* (pp. 2177-2184). IEEE.

[9] Kolias, C., Kambourakis, G., Stavrou, A., & Gritzalis, S. (2017). Intrusion Detection in 6LoWPAN Networks: Vulnerabilities, Threats, and Countermeasures. IEEE Transactions on Industrial Informatics, 13(2), 757-766.

[10] Feng, H., Llorca, J., Tulino, A. M., Raz, D., & Molisch, A. F. (2017, May). Approximation algorithms for the NFV service distribution problem. In IEEE INFOCOM 2017-IEEE Conference on Computer Communications (pp. 1-9). IEEE.

[11] Abdullah, A. A., Rawat, D. B., & Yan, G. (2020). Machine Learning Techniques in IoT Security: A Comprehensive Review. IEEE Access, 8, 186589-186615.

[12] Mahmood, A., Ramzan, N., Javaid, N., & Khan, A. R. (2019). Real-time Intrusion Detection in IoT-Based Smart Grid: A Machine Learning Approach. In 2019 21st International Conference on Advanced Communication Technology (ICACT), 1-6.

[13] Javed, M. A., Javaid, N., Baker, T., & Alrajeh, N. (2021). Real-Time Intrusion Detection Systems in the Internet of Things: A Comprehensive Survey. IEEE Internet of Things Journal, 8(4), 2538-2555.

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

[14] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2017). Fairness and Abstraction in Sociotechnical Systems. In Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT/ML), 59-68.

[15] Mittal, P. (2019). A Survey of Techniques for Achieving Differential Privacy in IoT. Journal of King Saud University - Computer and Information Sciences.

[16] Zarpelão, B. B., Miani, R. S., Kawakani, C. T., & de Alvarenga, S. C. (2017). A survey of intrusion detection in Internet of Things. *Journal of Network and Computer Applications*, *84*, 25-37.

[17] Singha, A. K., Zubair, S., Malibari, A., Pathak, N., Urooj, S., & Sharma, N. (2023). Design of ANN Based Non-Linear Network Using Interconnection of Parallel Processor. *Computer Systems Science & Engineering*, *46*(3).

[18] Yang, J. Q., Wang, R., Ren, Y., Mao, J. Y., Wang, Z. P., Zhou, Y., & Han, S. T. (2020). Neuromorphic engineering: from biological to spike-based hardware nervous systems. *Advanced Materials*, *32*(52), 2003610.

[19] Tang, J., Yuan, F., Shen, X., Wang, Z., Rao, M., He, Y., ... & Wu, H. (2019). Bridging biological and artificial neural networks with emerging neuromorphic devices: fundamentals, progress, and challenges. *Advanced Materials*, *31*(49), 1902761.

[20] Atangana, A., & Akgül, A. (2020). Can transfer function and Bode diagram be obtained from Sumudu transform. *Alexandria Engineering Journal*, *59*(4), 1971-1984.

[21] Chehal, D., Gupta, P., Gulati, P., & Gupta, T. (2023). Comparative Study of Missing Value Imputation Techniques on E-Commerce Product Ratings. *Informatica*, *47*(3).

[22] Chakrabarti, S., Biswas, N., Karnani, K., Padul, V., Jones, L. D., Kesari, S., & Ashili, S. (2023). Binned Data Provide Better Imputation of Missing Time Series Data from Wearables. *Sensors*, *23*(3), 1454.

[23] Shelke, M. S., Deshmukh, P. R., & Shandilya, V. K. (2017). A review on imbalanced data handling using undersampling and oversampling technique. *Int. J. Recent Trends Eng. Res*, *3*(4), 444-449.

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1

[24] Mohammed, A. J., Hassan, M. M., & Kadir, D. H. (2020). Improving classification performance for a novel imbalanced medical dataset using SMOTE method. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(3), 3161-3172.

[25] Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing* (pp. 878-887). Berlin, Heidelberg: Springer Berlin Heidelberg.

[26] Zhou, P., Feng, J., Ma, C., Xiong, C., & Hoi, S. C. H. (2020). Towards theoretically understanding why sgd generalizes better than adam in deep learning. *Advances in Neural Information Processing Systems*, *33*, 21285-21296.

[27] Daubechies, I., DeVore, R., Foucart, S., Hanin, B., & Petrova, G. (2022). Nonlinear approximation and (deep) ReLU networks. *Constructive Approximation*, *55*(1), 127-172.

[28] Kim, J. H. (2009). Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational statistics & data analysis*, *53*(11), 3735-3745.

[29] Haghighi, S., Jasemi, M., Hessabi, S., & Zolanvari, A. (2018). PyCM: Multiclass confusion matrix library in Python. *Journal of Open Source Software*, *3*(25), 729.

[30] Markoulidakis, I., Kopsiaftis, G., Rallis, I., & Georgoulas, I. (2021, June). Multi-class confusion matrix reduction method and its application on net promoter score classification problem. In *The 14th pervasive technologies related to assistive environments conference* (pp. 412-419).

[31] Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., & Gehler, P. (2022). Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 14318-14328).

[32] MacEachern, S. J., & Forkert, N. D. (2021). Machine learning for precision medicine. *Genome*, *64*(4), 416-425.

[33] Fu, G., Sun, P., Zhu, W., Yang, J., Cao, Y., Yang, M. Y., & Cao, Y. (2019). A deep-learning-based approach for fast and robust steel surface defects classification. *Optics and Lasers in Engineering*, *121*, 397-405.

E-ISSN 2976-9361
https://doi.org/10.59992/IJCI.2023.v2n8p1