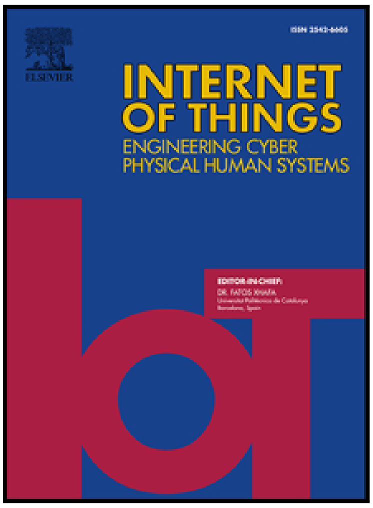

Improve Quality of Service for the Internet of Things using Blockchain & Machine Learning Algorithms.

Lawrence Nforh CheSuh , Ramón Ángel Fernández Díaz ,
Jose Manuel Alija Perez , Cármen Benavides Cuellar ,
Héctor Alaiz Moretón

PII: S2542-6605(24)00065-9
DOI: https://doi.org/10.1016/j.iot.2024.101123
Reference: IOT 101123

To appear in: *Internet of Things*

Received date: 24 April 2023
Revised date: 11 February 2024
Accepted date: 12 February 2024

Improve Quality of Service for the Internet of Things using Blockchain & Machine Learning Algorithms,

Lawrence Nforh CheSuh
lnforc00@estudiantes.unileon.es
Área de Ingeniería de Sistemas y Automática.
Dpto. Ingeniería Eléctrica y de Sistemas y Automática.
Edificio Tecnológico - Campus de Vegazana s/n 24071.
Universidad de León. León.

Ramón Ángel Fernández Díaz
ramon.fernandez@unileon.es
Área de Lenguajes y Sistema Informáticos.
Dpto. Ingeniería Ing. Mecánica, Informática y Aeroespacial.
Edificio Tecnológico - Campus de Vegazana s/n 24071.
Universidad de León. León.

Jose Manuel Alija Perez
jmalip@unileon.es
Área de Lenguajes y Sistema Informáticos.
Dpto. Ingeniería Ing. Mecánica, Informática y Aeroespacial.
Edificio Tecnológico - Campus de Vegazana s/n 24071.
Universidad de León. León.

Cármen Benavides Cuellar
carmen.benavides@unileon.es
Área de Ingeniería de Sistemas y Automática.
Dpto. Ingeniería Eléctrica y de Sistemas y Automática.
Edificio Tecnológico - Campus de Vegazana s/n 24071.

Héctor Alaiz Moretón
hector.moreton@unileon.es
Área de Ingeniería de Sistemas y Automática.
Dpto. Ingeniería Eléctrica y de Sistemas y Automática.
Edificio Tecnológico - Campus de Vegazana s/n 24071.

Abstract: The quality of service (QoS) parameters in IoT applications plays a prominent role in determining the performance of an application. Considering the significance and popularity of IoT systems, it can be predicted that the number of users and IoT devices are going to increase exponentially shortly. Therefore, it is extremely important to improve the QoS provided by IoT applications to increase their adaptability. Majority of the IoT systems are characterized by their heterogeneous and diverse nature. It is challenging for these systems to provide high-quality access to all the connecting devices with uninterrupted connectivity. Considering their heterogeneity, it is equally difficult to achieve better QoS parameters. Artificial intelligence-based machine learning (ML) tools are considered a potential tool for improving the QoS parameters in IoT applications. This research proposes a novel approach for enhancing QoS parameters in IoT using ML and Blockchain techniques. The IoT network with Blockchain technology is simulated using an NS2 simulator. Different QoS parameters such as delay, throughput, packet delivery ratio, and packet drop are analyzed. The obtained QoS values are classified using different ML models such as Naive Bayes (NB), Decision Tree (DT), and Ensemble, learning techniques. Results show that the Ensemble classifier achieves the highest classification accuracy of 83.74% compared to NB and DT classifiers.

Keywords: Quality of Service (QoS), Internet of Things (IoT), Blockchain, Machine Learning, Classification accuracy, Naive Bayes, Decision Tree, Ensemble Learning

1. Introduction

The Internet of Things (IoT) is considered one of the advanced visions of digital network systems that have transformed the phenomenon of data transmission in recent times (Rajab & Cinkelr, 2018) [1]. IoT is an integration of several devices into a single network that enables communication between various heterogeneous devices. The implementation of IoT-based applications has gained huge significance in recent times because of its capacity to transform human lives and make them simple and easier (Shah & Yaqoob, 2016) [2] (Agarwal, & Alam, 2020) [3]. Past decades have seen a sharp rise in the research related to IoT-based applications. However, the majority of the existing works have focused on different aspects of IoT and very minimal attention is given to the improvement of Quality of Service (QoS) in IoT applications (Nauman et al., 2020) [4]. It must be ensured that every layer in the IoT architecture is updated to achieve QoS and thereby guarantees a continuous and sustainable service for critical applications (Alhasan et al., 2019) [5]. Several critical issues might arise due to poor QoS. This issue can be preposterous in applications such as smart healthcare systems and autonomous vehicles wherein the delay in the response of physical sensors can lead to lethality. To overcome such scenarios, the QoS must be maintained in all network layers and each layer should provide active feedback for effective communication (Duan et al., 2011) [6] (Singh, M., & Baranwal, 2018) [7]. Maintaining QoS is extremely important in applications that demand high-quality data transmission with a minimum amount of delay.

Apart from delay, several other QoS factors such as packet delivery ratio, maximum throughput, security, and reliability also influence the efficiency of IoT systems. These factors are determined according to the user requirements and are evaluated based on the service requested by a specific application node (Khan et al., 2012) [8]. As discussed previously, a high-level IoT architecture incorporated with several network layers is expected to provide high-quality services with better assistance and maintenance. In addition, IoT systems are also expected to perform complex computation tasks in application-specific domains. In such cases, it is challenging to maintain the QoS considering the involvement of different quality factors (Alhasan et al., 2019) [5] (Liang et al., 2013) [9]. Various approaches and methodologies have been proposed for improving QoS in a high-level IoT architecture which includes the design of an efficient architecture with high-quality service components, suitable protocols, and robust access networks. These factors play an important role in developing a service-oriented IoT architecture (Sosa-Reyna et al., 2018) [10] (Varga et al., 2018) [11].

Despite the availability of several IoT architectures for improving QoS, there is still a lack of an effective technique that signifies the importance of QoS in different IoT applications such as smart cities, smart healthcare systems, financial services, etc (Huang et al., 2017) [12] (Patan et al., 2020) [13] (Dineshreddy & Gangadharan, 2016) [14]. For effective management of QoS in IoT applications, it is essential to cover all aspects of QoS metrics namely reliability, scalability, security, optimal resource allocation, etc. QoS helps in managing the responsibilities of the system effectively and to provide IoT services. In this context, this paper presents a novel approach to improving QoS in IoT. Various performance metrics are considered for improving the QoS which are discussed in the next sections. This paper emphasizes the application of ML and Blockchain for maximizing the performance of QoS parameters. Since IoT deals with real-time data, Blockchain combined with IoT helps in authorizing and validating the data transactions (Lau et al., 2018) [15]. Blockchain-powered IoT systems are characterized by high security and resilience to potential security threats

(Khalid et al., 2020) [16]. On the other hand, ML models help in classifying the QoS parameters for better performance.

The main contributions of this paper are summarized as follows:

- This paper presents a novel system architecture that combines Blockchain and ML algorithms for the enhancement of QoS parameters.
- The QoS parameters are analyzed through proper data communication flows, a guarantee of the service and performance of the network, and improved security through Blockchain authentication against unknown IoT nodes.
- This paper focuses on improving QoS parameters such as delay, packet loss, and throughput. The improved QoS parameters are deployed as input to the ML classifiers for classification. Furthermore, the QoS is enhanced using Blockchain technology.
- The classification of QoS parameters is validated using different ML classifiers such as Naive Bayes, Decision Tree, and Ensemble learning techniques.

The rest of the paper is further structured as follows: Section II discusses the existing literary works related to QoS improvement in IoT. Section III presents the deployment of the IoT smart network in the NS2 simulator. Section IV discusses the classification process using ML classifiers. Section V presents the results of the simulation analysis and Section VI concludes the paper with prominent research observations and future research directions.

2. Related Works

Several research works have been presented in the past decades related to QoS improvement in different IoT applications (Zafar et al., 2019) [17] (Jaiswal & Anand, 2021) [18] (Shankhpal & Savadatti Hanumantha, 2022) [19]. The heterogeneous nature of IoT devices makes it difficult for the applications to deliver a reliable QoS. This problem motivates the researchers to explore different techniques to improve the QoS by transforming conventional layered solutions. This paper intends to comprehensively analyze the solutions available for improving QoS. In this context, this paper surveys various existing related works and summarizes the contributions of the research works. (Simiscuka & Muntean, 2018) [20] proposed a relay and mobility-based approach for improving QoS in IoT communication systems. The proposed IoT architecture consists of different IoT devices which provide diversified services such as data monitoring, analyzing the sensor data, and interconnection of various smart devices. A cloud-based platform is deployed for obtaining optimal resource management with an appropriate decision-making process to enhance network quality. A novel algorithm is proposed for improving the quality of IoT services by managing service-relevant metrics and QoS parameters. The devices and objects were attached to the relays for evaluation and the performance of underperforming clusters was improved.

Most of the existing research works have emphasized on other aspects in IoT with few research works intend to improve the security of IoT data along with QoS enhancement. The deployment of IoT can be considered as successful if the amount of data that is either being transferred or received over the networks meets the desired QoS requirements (Sheikh et al., 2019) [21]. Since IoT deals with several devices and shares data over multiple communication networks, it is highly susceptible to security threats and attacks. Hence it becomes essential to secure the IoT devices and protect the confidentiality of data that is transmitted across different networks. Different algorithms have been analyzed in the work proposed by (Sheikh et al., 2019) [21] for protecting the data from being exploited. Algorithms such as energy efficient secure route adjustment (ESRA) (Jain et al., 2019) [22], double level unequal clustering algorithm (DLUC) (Farahani et al., 2019) [23], Stochastic and diffusive routing algorithm (SDR) (Manjula & Datta, 2018) [24], sector-based random routing scheme (SRR) (He et al., 2019) [25], and source location privacy based ring loop routing (SLPRR) (Wang et al., 2019) [26] are proven to be efficient in providing the security to IoT data along with the improvement in the QoS metrics. The study suggests the application of ML algorithms for securing and improving QoS in IoT applications. (Sood et al., 2019) [27] aimed to alleviate the heterogeneity in software-defined network (SDN) based IoT systems. A novel approach is proposed in this study which exploits the heterogeneous controllers for improving the response time of SDN-IoT systems. A mathematical approach along with a proof of concept (POC) is proposed and the performance of the approach is evaluated. It was observed that the PoC approach reduces the heterogeneity which in turn improves the QoS and security of the IoT data. In addition to the QoS, the network security of the system in terms of handling the dynamic nature of the SDN- IoT system was also evaluated. Results validate the efficacy of the proposed approach. The application of ML for improving QoS in IoT is discussed by (Alsamhi et al., 2021) [28]. ML plays a significant role in improving communication and QoS in IoT-based smart applications. An empirical analysis of the application of ML algorithms for maximizing the performance of IoT applications was presented in the paper. It can be inferred from the study that ML algorithms are capable of transforming the operational process in different IoT applications such as smart cities, healthcare services, and other smart applications. ML algorithms are advantageous in terms of their superior prediction performance and ability to extract and classify data patterns while handling large-scale datasets. An artificial intelligence (AI) based approach is proposed by (Sheikh et al., 2022) [29] for improving the QoS in IoT networks. The study focuses on monitoring the changes in the QoS parameters in IoT systems with varying numbers of nodes. Different parameters such as energy consumption, jitter, packet delivery ratio, and throughput are evaluated for different nodes. The performance of QoS parameters with and without AI was compared. Results show that the values of most of the QoS parameters were improved significantly with AI compared to conventional non-AI methods. A comprehensive analysis of different deep learning models for enhancing QoS in IoT applications is presented in (Kimbugwe et al., 2021) [30]. It can be observed that the QoS in IoT systems are affected with the quality and efficiency of the IoT resources. In addition, the security and privacy of IoT networks also play an important role in deciding the QoS parameters (Asharf et al., 2020) [31] (Hussain et al., 2020) [32] (Al-amri et al., 2021) [33]. It can be inferred from the existing works that the ML algorithms have a potential to transform IoT processes. However, there is a need to analyze their performance in terms of authenticating the information and improving the communication. This aspect motivated this research to investigate the performance of different ML models for enhancing the QoS parameters in IoT.

3. Proposed Research Methodology

3.1 Proposed IoT-based Smart Network

This research proposes the deployment of an IoT-based smart system using Blockchain. The proposed smart system is composed of different IoT components, sensors, and connecting devices. A huge volume of data is collected from sensors and IoT devices. This increases the heterogeneity and complexity of the IoT networks. As discussed previously, the heterogeneous nature of IoT makes it difficult to maintain the QoS in all layers of the IoT architecture. In addition, the sharing of data across different networks raises security concerns. A Blockchain-based authentication scheme is proposed for securing the IoT data against unknown IoT nodes and to establish trust in the network which is infinitely more secure than the comuter systems that we have today.

In real time work, the hardware implementation must be designed with sensor devices and the IoT MQTT communication protocol has to be added in to the hardware module .

Different operations of IoT such as data collection, monitoring network activity, data transmission, and storage requires high QoS. These operations are performed by delegated agents in the IoT network to satisfy desired QoS requirements. The agents are responsible for collecting the real-time and maintaining the previously stored data in the system. A set of agents in the network together form an overlay zone or subnetworks and these subnetworks form the IoT network. The parameters used for determining the QoS are defined by their scalability and the agents in the IoT system also act as Blockchain nodes. Blockchain authenticates the nodes in each subnetwork and thereby ensures the security of data stored and transmitted by these nodes. A Blockchain-based secure hash code authentication scheme is deployed along with different sensors for creating a smart path in the IoT network. This scheme solves the problems of network congestion, delay time, throughput, packet delivery ratio, and packet loss ratio in the proposed IoT-based smart network.

In this work, a ML dataset is used for evaluating the QoS parameters which is simulated in a NS2 environment. In general, ML algorithms, support certain files and hence the data is converted into comma separated vector format (csv). The dataset contains 6 attributes and 413 records extracted from ns2 trace file analysis. This dataset attribute is containing simulation time,Port number, interface_type,routing_type,Packet_size, bcc_status and for classification dataset class label contains two things (a) Qos(Packet transferred successfully) and (b) 0-Qos(packet transfer failed). Finally, the IOT dataset was deployed and the ML algorithm was implemented in python using different machine learning algorithm with validation results.

3.2 Analysis of QoS Parameters

Different QoS parameters evaluated in this research are discussed as follows:

- ***Delay:*** Delay is defined as the time required to transfer a data packet or group of data packets from the source to the sink. Delay in IoT networks can either be a delay in the transmission or a delay in the processing of IoT data. During data transmission, there can be several barriers that affect the communication process and can cause delays in the data transfer process. Extreme delay can also result in the loss of important data while communicating across different networks. With the increase in the adoption of IoT systems, a huge volume of data is collected by the sensors and IoT devices which in turn increases the network traffic. The increasing network traffic causes delays in the network.

- ***Packet Loss Ratio (PLR):*** In general, data transmission is carried out by sending and receiving data packets from one end to another. While transferring data packets over the internet or any other wireless network, there are chances that certain small data packets fail to reach the destination. In this process, some valuable information might be lost and this loss is defined as packet loss. The packet loss ratio is defined as the ratio between the number of data packets lost to the total number of transferred packets. Packet loss can be caused due to high network congestion, problems due to network hardware, software bugs, and security threats. Packet loss hurts the performance of IoT systems since it disrupts the network operation and reduces the service quality. Therefore, it is extremely important to minimize packet loss to maintain a better QoS value.
- ***Throughput:*** During data transmission in IoT networks, a large volume of data is transmitted across the networks. It is important to ensure that the network possesses a high throughput value. Throughput is defined as the amount of data (or a number of data packets) delivered successfully from the source to the destination in a given period. Throughput is measured in bits per second (bps), as in megabits per second (Mbps) or gigabits per second (Gbps). A throughput value is set to control the workload on the IoT systems, and it must be ensured that the system should meet the throughput target irrespective of high workloads. Hence, throughput is considered an important QoS parameter in IoT networks.
- ***Jitter:*** Jitter is caused when the data packets transferred using a particular link reach the destination at different times. In other words, jitter is defined as the variation in the time delay during the process of data transmission. It is mainly caused due to network congestion, poor prioritization of data packets, and disruption in the performance of network hardware components.
- ***Packet Delivery Ratio (PDR):*** The PDR is obtained by determining the ratio of the total number of data packets received by the sink nodes to the total number of data packets transferred by the source node. In simple terms, PDR is the ratio of the number of data packets arriving at the destination to the number of data packets transmitted from the source. The value of PDR should be maintained such that a maximum number of data packets should reach the destination without losing any important information.
- ***Packet Drop:*** Packet drop occurs when the router in the network system intentionally drops the data packets. It is mainly caused due to faulty network wires, issues in the

software, network attacks, congestion in the network bandwidth, and insufficient hardware.

The flowchart of the QoS analysis process is illustrated in figure 3.1.

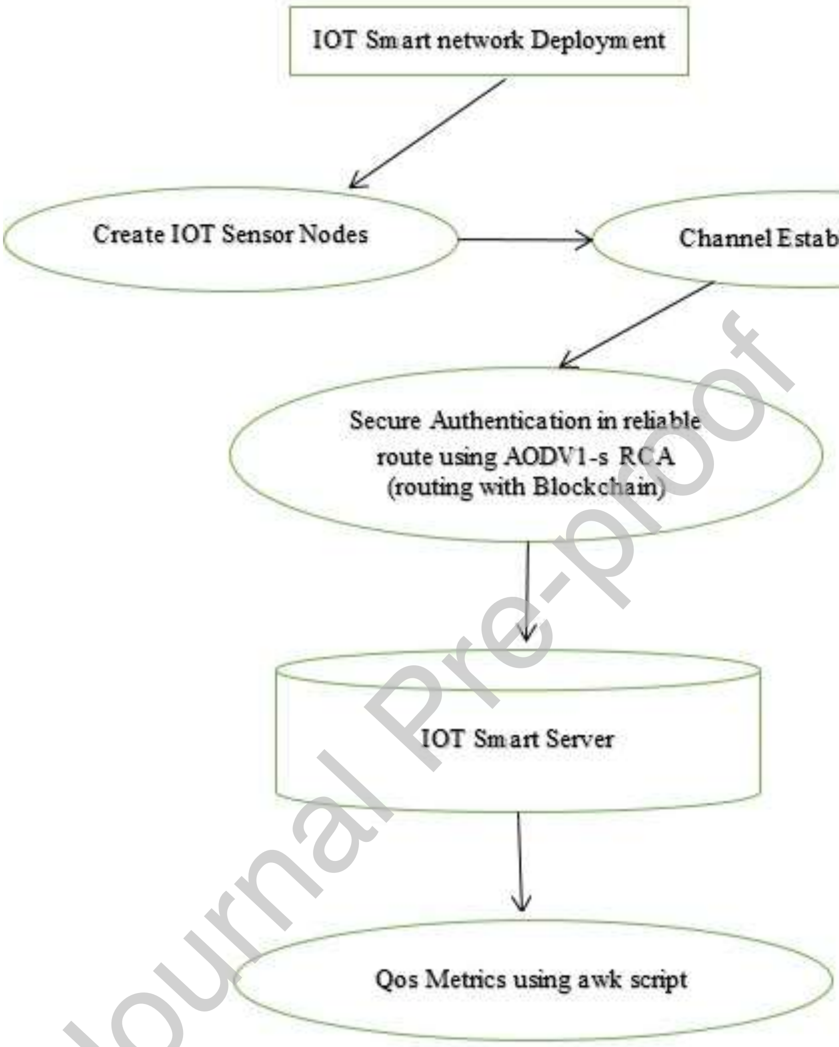
IOT Smart network Deployment
Create IOT Sensor Nodes
Channel Establishment
Secure Authentication in reliable route using AODV1-s RCA (routing with Blockchain)
IOT Smart Server
Qos Metrics using awk script

Figure 3.1 Flowchart of the QoS analysis process

The QoS parameters discussed in the above points are simulated in the NS2 simulator along with Blockchain based secure hash authentication scheme for securing IoT data. The enhanced QoS parameters are given as input to the machine learning algorithms for classification.

4. Machine Learning for Classification of QoS

The deployment of ML in smart IoT applications is increasing significantly in recent times. This is mainly due to the ability of ML algorithms to transform conventional tasks into automated tasks which are adaptable to dynamic environments that change continuously. Besides, ML assists IoT devices to perform complex tasks without any manual intervention. ML also can provide solutions to different IoT problems such as computational complexity, processing difficulty, and performance efficiency. ML helps machines and IoT devices to learn from previous instances and perform tasks accordingly (Ali et al., 2020) [34] (Haidian et al., 2021) [35]. As a result, the combined application of IoT and ML is implemented successfully in several integrated applications. The ML-IoT combination enhances different aspects of data transmissions including machine-to-machine communication without human interventions. In a complex and application-specific environment, connectivity is important and communication technology must meet QoS specifications. Achieving reliable communication between different IoT devices is a complex issue and enhancing the QoS in IoT systems is important for the successful deployment of IoT networks (Nurelmadina et al., 2021) [36].

This research emphasizes ML classifiers such as Naive Bayes, Decision Tree, and Ensemble Learning classifiers.

4.1 Naive Bayes Classifier

Naive Bayes (NB) classifiers belong to the family of fundamental probabilistic classifiers which employ a Bayes' theorem with robust naive assumptions between the features. The Naive Bayes classifiers are one of the simplest network models whose assumptions are independent. This property of the classifier allows easy training of the model without using any previous data. These classifiers are incorporated with kernel density estimation which improves the classification accuracy. The Naive Bayes classifiers possess high scalability and require fewer parameters for learning a problem. This will reduce the complexity of the network and minimize the computational burden on the network layers. Maximum likelihood training can be performed using the Naive Bayes classifiers by employing closed-form expression which requires a linear time, unlike conventional classifiers which use a computationally expensive iterative approximation technique.

For a given sample data to be classified, the data is represented in the form of a vector x = (x_1, x_2, …. xn) which represents the 'n' number of features or independent variables.

For all these variables, the classifier assigns a probability function 'p' for each possible output which is given as:

$$p\ (C_k \mid x_1, x_2, \ldots, x_n) \ \ldots.\ (1)$$

Where k is the number of possible outputs for classes C_k.

The issue with the problem formulation technique is that if the number of features is more or if the value of 'n' in a feature set is large, then it is difficult for the model to use it for classification purposes.

The pseudocode for the Naive Bayes Classifier is given as follows:

Pseudocode for Naive Bayes Algorithm

```
Step 1: Declare the input parameter and output parameter
            Input dataset [] = {"simulation time, Port number, interface type, routing type,
            Packet size, bcc status"}
            Output [] = {"status of bcc transfer:0 and 1"}
Step 2: Split the dataset as training and testing
         X Train, X Test, Y Train, Y Test = train test split (x, y, test size = 0.3, random state
         = 0)
Step 3: Import naive Bayes library
           def Naïve Bayes ()
           classifier = GaussianNB()
Step 4: Analysis of classification report
            Classification report (Y Test, Y Pred))
Step 5: Calculate the confusion matrix and accuracy
            cm = confusion matrix (Y Test, prediction)
            accuracy = 100.0 * accuracy score (Y Test, prediction)
            End Naïve Bayes ()
```

4.2 Decision Tree (DT) Classifier

The decision tree algorithm is one of the advanced supervised machine learning algorithms used for performing both classification and regression. Decision trees are aimed to construct a mode that can predict the value of a target variable by learning decision-based rules. These rules are derived from the extracted features. For a given sample, initially, an individual decision tree will perform a random selection process through the bootstrap resampling mechanism and the obtained samples will be employed for constructing a decision tree. A tree can be considered as a constant approximation for each sample data. The data obtained from the dataset is arranged and the possible outcome of the data is structured in the form of a decision tree. A decision tree is a hierarchical structure composed of multiple nodes and directed edges. A decision tree consists of three types of nodes.

- ***Root Node:*** This node has no incoming edges and has zero or more outgoing edges.
- ***Internal Nodes:*** Internal nodes have exactly one incoming node and have two or more outgoing edges.
- ***Leaf or terminal nodes***: In leaf or terminal nodes, each node has only one incoming edge and zero outgoing edges.

In a decision tree, each leaf node is provided with a class label and the non-terminal nodes, which consist of internal and root nodes, are incorporated with certain feature test constraints to distinguish different features with different characteristics. The decision tree mechanism is adopted to estimate the target functions of the discrete systems and the learned function is determined using decision tree algorithms. The mechanism involved in the decision tree algorithm is highly inductive and is used widely in image classification applications. A decision tree is a data mining technique and is most significant in IoT systems for classifying different QoS parameters with high classification accuracy.

The pseudocode for the Decision Tree Classifier is given as follows:

Pseudocode for Decision Tree Algorithm

```
Step 1: Declare the input parameter and output parameter
    Input dataset [] = {"simulation time, Port number}
    Output [] = {"status of bcc transfer:0 and 1"}
    Interface type, routing type, Packet_size, bcc_status"}
Step 2: Split the dataset as training and testing
    X_Train, X_Test, Y_Train, Y_Test = train_test_split(x, y, test_size = 0.3,
    random_state = 0)
Step 3: Import the decision tree library
    def Decision_Tree()
    clf = tree.DecisionTreeClassifier()
Step 4: Analysis of classification report
    classification_report(Y_Test, Y_Pred))
Step 5: Calculate the confusion matrix and accuracy
    cm = confusion_matrix(Y_Test, prediction)
    accuracy = 100.0 * accuracy_score(Y_Test, prediction)
    End Decision Tree ()
```

4.3 Ensemble Learning Classifier

Ensemble learning is the combination of two or more ML classifiers. Ensemble learning helps ML models to achieve better performance by combining several individual ML models. Compared to single ML modes, ensemble models achieve better predictive and classification performance. In IoT, ensemble learning overcomes different problems such as statistical problems, computational problems, and representational problems.

The pseudocode for the Ensemble learning classifier is given as follows:

Pseudocode for Ensemble Learning Algorithm

```
Step 1: Declare the input parameter and output parameter
        Input dataset [] = {"simulation time, Port number,
        Interface_type, routing_type, Packet_size, bcc_status"}
        Output [] = {"status of bcc transfer:0 and 1"}
Step 2: Split the dataset as training and testing
        X_Train, X_Test, Y_Train, Y_Test = train_test_split(x, y, test_size = 0.3,
        random_state = 0)
 Step 3: Import preprocessing library
          def preprocessing ():
          begin
          for {set i=0 i<=data_limit; incr i}
          begin
          Resize the data using StandardScaler()
          end for
          end Preprocessing ();
 Step 4: Import ensemble library
          def ensemble ()
        RandomForestClassifier(n_estimators = 1000, criterion = 'entropy', random_state = 0)
Step 5: Analysis of classification report
        classification report (Y_Test, Y_Pred))
Step 6: Calculate the confusion matrix and accuracy
        cm = confusion matrix (Y_Test, prediction)
        accuracy = 100.0 * accuracy score (Y_Test, prediction)
        End ensemble ()
```

5. Results and Discussion

This section discusses the results of the simulation analysis. The proposed IoT-based smart network is simulated in the NS2 simulator. The input data for IoT is collected from the NS2 trace file which was already simulated in NS2. The network configuration and simulation parameters used for the experimental analysis are tabulated in table 1.

Table 1. Network Configuration and Simulation Parameters

S.No	Parameter	Value	Explanation
1	Network Area (m²)	1821 * 500	Defines the physical space for node deployment.
2	Number of Sensor Nodes	50	Determines the network's complexity and data generation.
3	Packet Size (bytes)	1024	Affects network efficiency and latency.
4	Propagation Model	Two-Ray Ground	Simulates signal propagation.
5	Number of Channels	16	Provides bandwidth for communication.
6	MAC Standard	IEEE 802.11	Defines wireless communication protocol.
7	Antenna	Uni-directional	Directs signal transmission and reception.
8	Scheduler	MAC Scheduler	Manages access to the shared wireless medium.
9	Mobility Model	Random Waypoint	Simulates node movement patterns.
10	Internet Protocol	UDP	Connectionless datagram protocol.

11	IoT Protocol	MQTT	Lightweight messaging protocol.
12	Traffic Agent	CBR (Constant Bit Rate)	Generates steady data traffic.

The results of the simulation analysis in terms of different QoS parameters are illustrated in the figures below. The simulated environment consists of the IoT-based smart system and Blockchain-based secure hash authentication technique. The network scenario is constructed with 50 sensor nodes and 16 channels in the IoT-based smart networks and the data stored in the nodes are secured against various attacks using a secure hash authentication scheme. In this case, each user must be authenticated and verified using a secure hash authentication scheme. The data can only be accessed by the user after validating the user's requests.

In addition, all blocks are verified using hash code. Simulation shows that the QoS is enhanced.

The simulation analysis of QoS parameters is presented in the figures below:

- **Delay Vs Buffer Size**

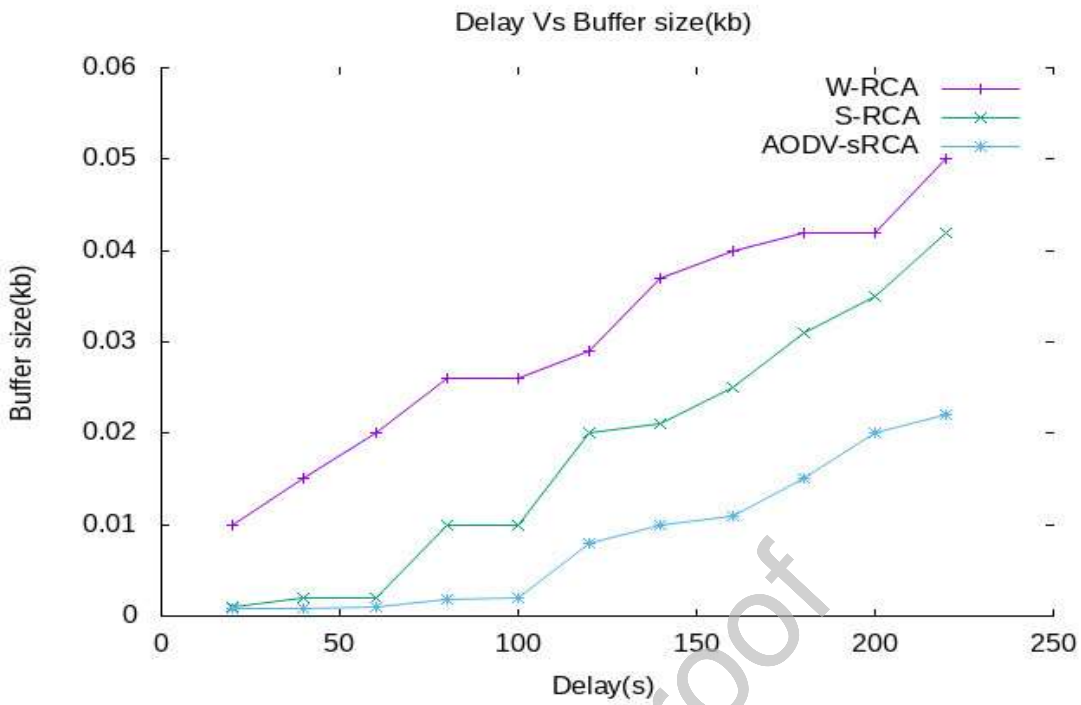

Figure 5.1 Simulation of Delay Vs Buffer Size

- **Number of nodes Vs Packet Loss Ratio**

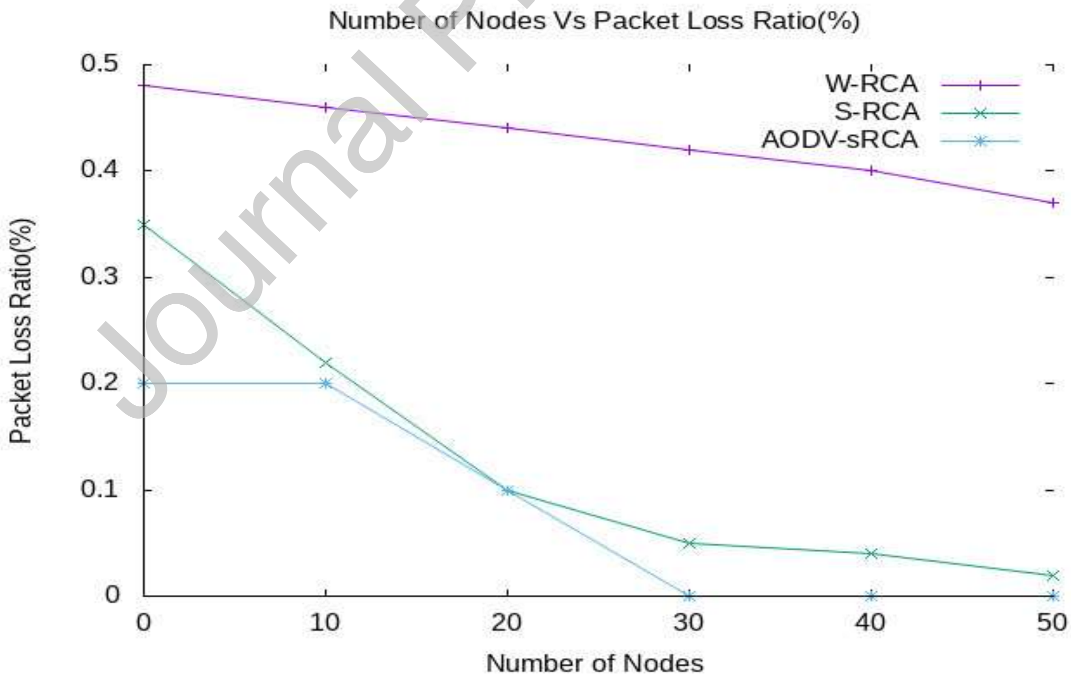

Figure 5.2 Simulation of Number of nodes Vs Packet Loss Ratio

- **Throughput**

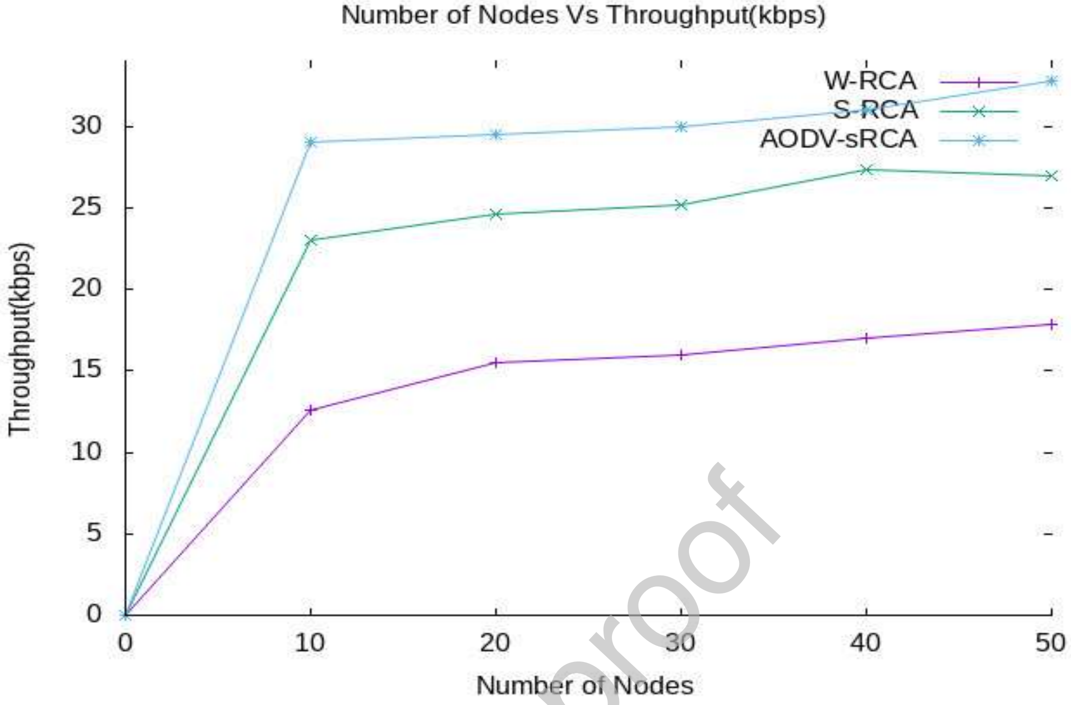

Figure 5.3 Simulation of Throughput analysis

- **Number of Nodes Vs Jitter**

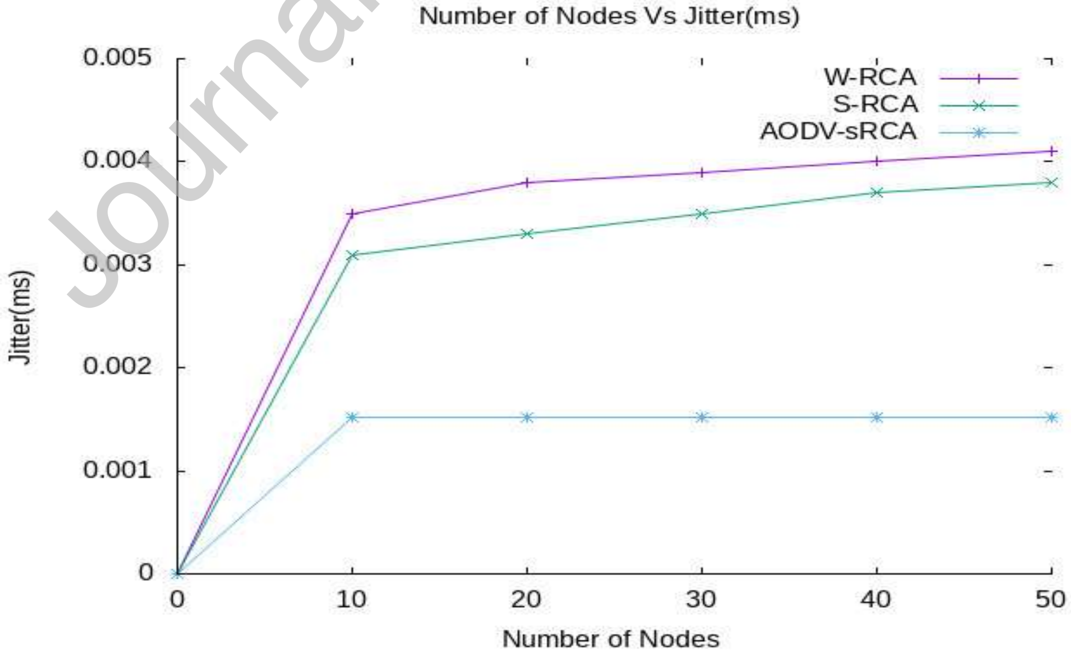

Figure 5.4 Simulation of Number of Nodes Vs Jitter

- **Number of nodes Vs Packet Delivery ratio**

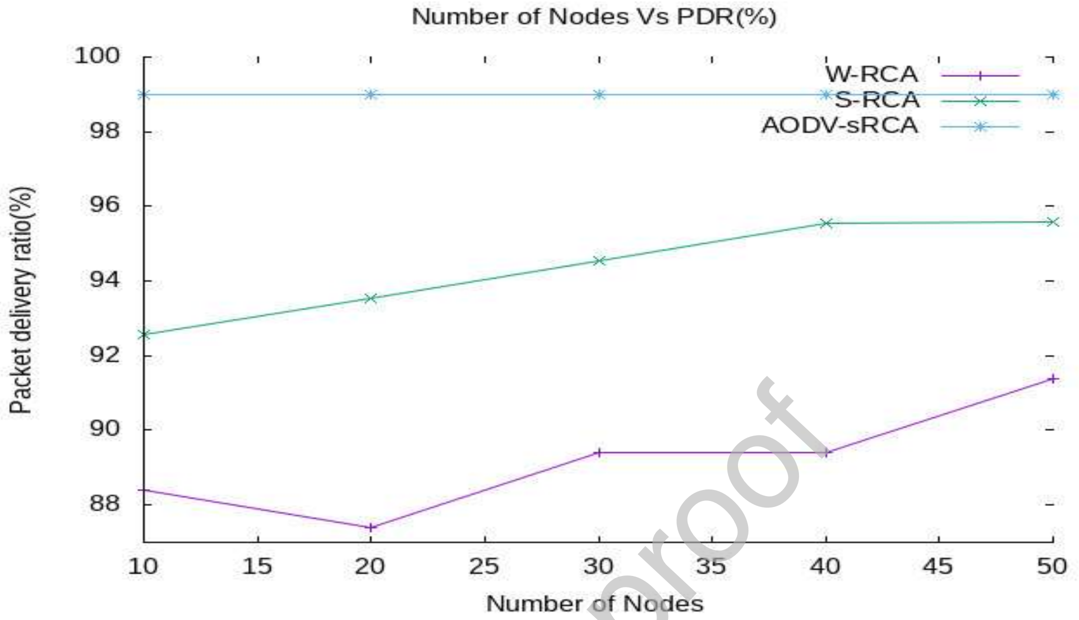

Figure 5.5 Simulation of Number of Nodes Vs Packet Delivery ratio

- **Number of Nodes Vs Packet Drop**

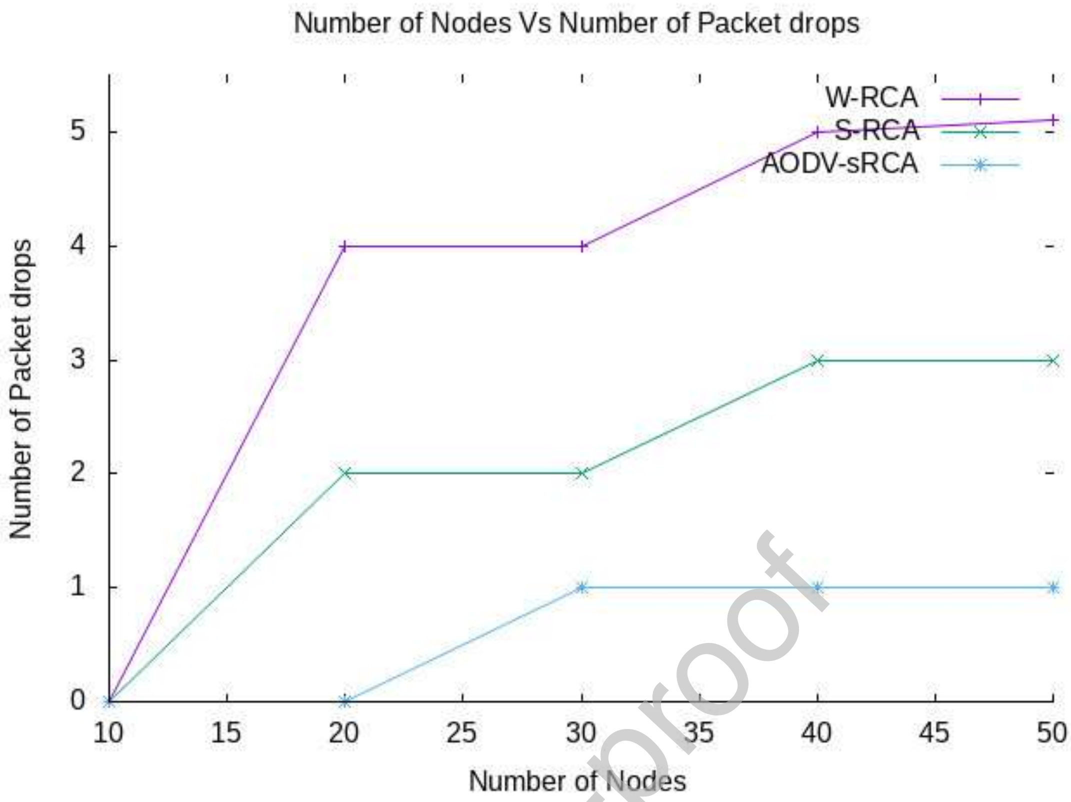

Figure 5.6 Simulation of Number of Nodes Vs Packet Drop

The simulation of delay analysis concerning buffer size is analyzed and validated using different approaches such as wRCA, sRCA, and AODV-sRCA using blockchain technology. It can be observed from the simulation that the AODV_sRCA approach achieves better performance in terms of enhancing different QoS parameters compared to the other two methods.

After the execution of the NS2 simulation, two textual output files were obtained namely the Nam file and the Trace file(.tr). ML supports only structured data formats; hence, the data was converted into the corresponding structured data format in the. CSV format.

The dataset contains 6 attributes and 413 records extracted from NS2 trace file analysis. This dataset attribute is containing simulation time, Port number, interface type, routing type, Packet size, and bcc status (behavior change communication). The QoS is classified into two class labels i.e., 1 and 0 where 1 states that the packets are transferred successfully and 0 states that the packet transfer failed, as shown below.

Classification result; 1: QoS(Packet transferred successfully)

0: QoS (packet transfer failed)

The obtained IoT dataset was deployed and implemented in python using different ML algorithms with validation results. Classification is performed using an ensemble, decision tree, and naive Bayes algorithm. The performance of the classifiers is evaluated in terms of different performance metrics namely accuracy, precision, and recall. These metrics are measured using four different classification elements namely: True positives (TP), True negatives (TN), False positives (FP), and False negatives (FN). These terms are used for constructing a confusion matrix. The confusion matrix is mainly used for solving problems related to classification accuracy where the output can be two or more classes.

The mathematical expressions for calculating the performance metrics are discussed in below equations:

$$\textit{\textbf{Accuracy}} = \frac{TP+TN}{TP+TN+FP+FN} \ldots. (2)$$

$$\textit{\textbf{Recall}} = \frac{TP}{TP+FN} \ldots.. (3)$$

$$\textit{\textbf{Precision}} = \frac{TP}{TP+FP} \ldots. (4)$$

The results of the ML classifiers are illustrated below:

- **Ensemble Learning Classifier**

The confusion matrix of the ensemble learning classifier and the results are presented as follows:

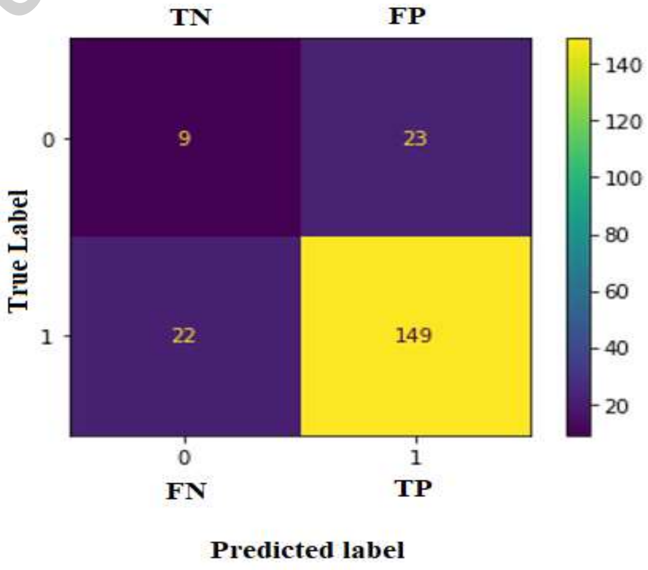

Figure 5.7 Confusion matrix for ensemble learning classifier

Here, the values of TN, FP, FN, and TP are 9, 23, 22, and 149 respectively. Based on this, the values of accuracy, recall, and precision is calculated using equation 2, 3, and 4. The values of the same are tabulated in table 2. Similar to the ensemble learning classifier, the confusion matrix of the decision tree and naive Bayes algorithm is illustrated in figures 5.8 and 5.9 respectively.

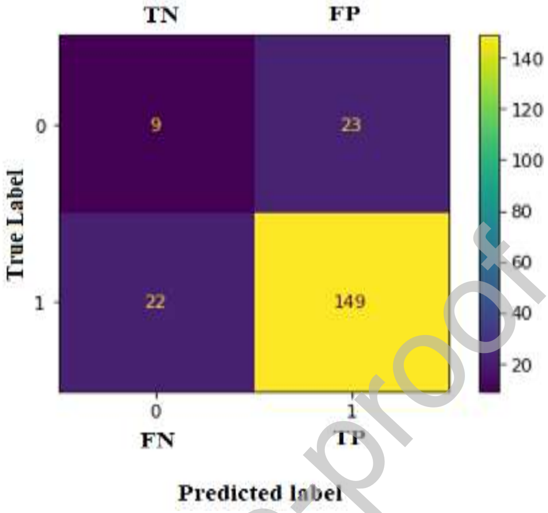

Figure 5.8 Confusion matrix for decision tree classifier

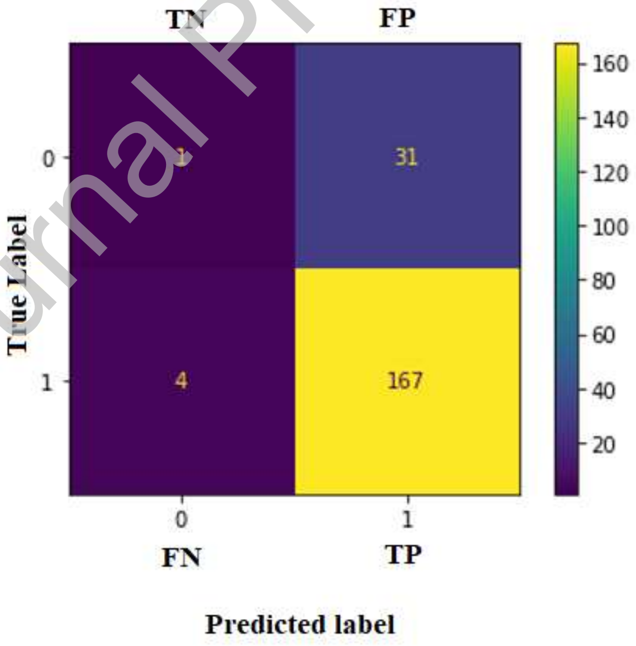

Figure 5.9 Confusion matrix for Naive Bayes classifier

Table 2. Performance metrics of the ML classifiers

	Ensemble Learning	**Decision Tree**	**Naive Bayes**

Accuracy	83.74%	77.83%	82.75%
Precision	86%	87%	84.34%
Recall	97%	87.13%	98%

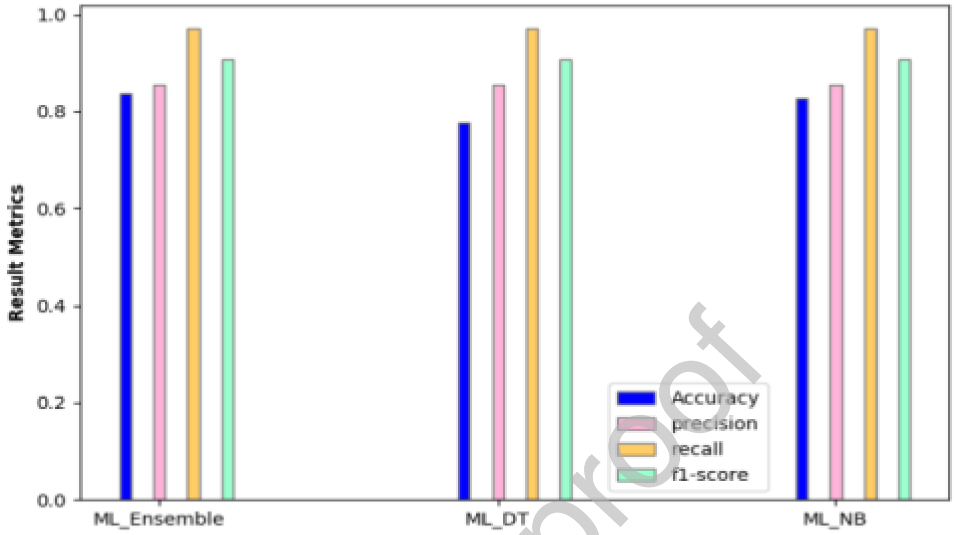

Figure 5.10 Comparison of ML classifiers

It can be inferred from the simulation results that the ensemble learning algorithm achieves better classification accuracy compared to the other two algorithms.

5. Conclusion

In this work, a ML dataset is used for evaluating the QoS parameters which are simulated in a NS2 environment. In general, ML algorithms, support certain files and hence the data is converted into comma separated vector format (csv). The dataset contains 6 attributes and 413 records extracted from ns2 trace file analysis.

The QoS parameters was analyzed by the proposed IoT and Blockchain technology simulated using an NS2 simulator. The parameters are analyzed in terms of swift data communication, better service, robust network performance, and security of IoT data. We used the classifier to access the success or failures of the packet as this could be another way of analyzing the quality of services. A service is successful if a packet is delivered (correct 1) and vise versa. This is the result of the quality of service. A Blockchain-based authentication approach was implemented for securing the data against unknown IoT nodes present in the network thereby providing trust. Different parameters such as delay, packet loss, and throughput are improved in this research. The enhanced parameters are classified using different machine learning classifiers. A classifier therefore is used here as a parameter that can be used to access Qos. The performance of ML classifiers such as NB, DT, and Ensemble classifiers were evaluated in python concerning different performance metrics namely precision, recall, and accuracy. The design of the novel AODV – sRCA algorithm enhances the performance of the classifier. It was observed from the results that the accuracy of the ensemble classifier, NB classifier, and decision tree was found to be 83.74%, 82.75%, and 77.83% respectively. Results validate that the ensemble classifier achieves better classification performance compared to other algorithms and the values of QoS parameters were enhanced. For future research, the study can be extended to the energy consumption analysis of QoS parameters and verify the classification performance using neural networks.

References

1. Rajab, H., & Cinkelr, T. (2018, June). IoT based smart cities. In *2018 international symposium on networks, computers and communications (ISNCC)* (pp. 1-4). IEEE.

2. Shah, S. H., & Yaqoob, I. (2016). A survey: Internet of Things (IOT) technologies, applications and challenges. *2016 IEEE Smart Energy Grid Engineering (SEGE)*, 381-385.

3. Agarwal, P., & Alam, M. (2020). Investigating IoT middleware platforms for smart application development. In *Smart Cities—Opportunities and Challenges* (pp. 231-244). Springer, Singapore.

4. Nauman, A., Qadri, Y. A., Amjad, M., Zikria, Y. B., Afzal, M. K., & Kim, S. W. (2020). Multimedia Internet of Things: A comprehensive survey. *IEEE Access*, *8*, 8202-8250.

5. Alhasan, A., Audah, L., Alhadithi, O. S., & Alwan, M. H. (2019). Quality of service mechanisms in internet of things: A comprehensive survey. *Journal of Advanced Research in Dynamical and Control Systems*, *11*(2), 858-875.

6. Duan, R., Chen, X., & Xing, T. (2011, October). A QoS architecture for IOT. In *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing* (pp. 717-720). IEEE.

7. Singh, M., & Baranwal, G. (2018, February). Quality of service (qos) in internet of things. In *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)* (pp. 1-6). IEEE.

8. Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012, December). Future internet: the internet of things architecture, possible applications and key challenges. In *2012 10th international conference on frontiers of information technology* (pp. 257-260). IEEE.

9. Liang, J. M., Chen, J. J., Cheng, H. H., & Tseng, Y. C. (2013). An energy-efficient sleep scheduling with QoS consideration in 3GPP LTE-advanced networks for internet of things. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, *3*(1), 13-22.

10. Sosa-Reyna, C. M., Tello-Leal, E., & Lara-Alabazares, D. (2018). Methodology for the model-driven development of service oriented IoT applications. *Journal of Systems Architecture*, *90*, 15-22.

11. Varga, P., Kozma, D., & Hegedús, C. (2018, September). Data-driven workflow execution in service oriented iot architectures. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)* (Vol. 1, pp. 203-210). IEEE.

12. Huang, J., Xing, C. C., Shin, S. Y., Hou, F., & Hsu, C. H. (2017). Optimizing M2M communications and quality of services in the IoT for sustainable smart cities. *IEEE Transactions on Sustainable Computing*, *3*(1), 4-15.

13. Patan, R., Ghantasala, G. P., Sekaran, R., Gupta, D., & Ramachandran, M. (2020). Smart healthcare and quality of service in IoT using grey filter convolutional based cyber physical system. *Sustainable Cities and Society*, *59*, 102141.

14. Dineshreddy, V., & Gangadharan, G. R. (2016, March). Towards an "Internet of Things" framework for financial services sector. In *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)* (pp. 177-181). IEEE.

15. Lau, C. H., Alan, K. H. Y., & Yan, F. (2018, December). Blockchain-based authentication in IoT networks. In *2018 IEEE Conference on Dependable and Secure Computing (DSC)* (pp. 1-8). IEEE.

16. Khalid, U., Asim, M., Baker, T., Hung, P. C., Tariq, M. A., & Rafferty, L. (2020). A decentralized lightweight blockchain-based authentication mechanism for IoT systems. *Cluster Computing*, *23*(3), 2067-2087.

17. Zafar, S., Jangsher, S., Bouachir, O., Aloqaily, M., & Othman, J. B. (2019). QoS enhancement with deep learning-based interference prediction in mobile IoT. *Computer Communications, 148,* 86-97.

18. Jaiswal, K., & Anand, V. (2020). EOMR: An energy-efficient optimal multi-path routing protocol to improve QoS in wireless sensor network for IoT applications. *Wireless Personal Communications*, *111*(4), 2493-2515.

19. Shankhpal, S. V., & Savadatti Hanumantha, B. (2022). KMFA2 based QoS improvement for multi-channel IoT networks. *Concurrency and Computation: Practice and Experience*, e6949.

20. Simiscuka, A. A., & Muntean, G. M. (2018, May). A relay and mobility scheme for QoS improvement in IoT communications. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)* (pp. 1-6). IEEE.

21. Sheikh, A., Ambhaikar, A., & Kumar, S. (2019). Quality of services improvement for secure iot networks. *International Journal of Engineering and Advanced Technology (IJEAT) ISSN*, 2249-8958.

22. Jain, J. K. (2019). Secure and energy-efficient route adjustment model for internet of things. *Wireless Personal Communications*, *108*(1), 633-657.

23. Farahani, M., & Ghaffarpour Rahbar, A. (2019). Double leveled unequal clustering with considering energy efficiency and load balancing in dense iot networks. *Wireless Personal Communications*, *106*(3), 1183-1207.

24. Manjula, R., & Datta, R. (2018). A novel source location privacy preservation technique to achieve enhanced privacy and network lifetime in WSNs. *Pervasive and Mobile Computing*, *44*, 58-73.

25. He, Y., Han, G., Wang, H., Ansere, J. A., & Zhang, W. (2019). A sector-based random routing scheme for protecting the source location privacy in WSNs for the Internet of Things. *Future Generation Computer Systems*, *96*, 438-448.

26. Wang, H., Han, G., Zhou, L., Ansere, J. A., & Zhang, W. (2019). A source location privacy protection scheme based on ring-loop routing for the IoT. *Computer Networks*, *148*, 142-150.

27. Sood, K., Karmakar, K. K., Yu, S., Varadharajan, V., Pokhrel, S. R., & Xiang, Y. (2019). Alleviating heterogeneity in SDN-IoT networks to maintain QoS and enhance security. *IEEE Internet of Things Journal*, *7*(7), 5964-5975.

28. Alsamhi, S. H., Almalki, F. A., Al-Dois, H., Othman, S. B., Hassan, J., Hawbani, A., ... & Saleh, H. (2021). Machine learning for smart environments in B5G networks: Connectivity and QoS. *Computational Intelligence and Neuroscience*, *2021*.

29. Sheikh, A., Kumar, S., & Ambhaikar, A. (2022). Improvement of QoS Parameters of IoT Networks Using Artificial Intelligence. In *Ubiquitous Intelligent Systems* (pp. 1-13). Springer, Singapore.

30. Kimbugwe, N., Pei, T., & Kyebambe, M. N. (2021). Application of deep learning for quality of service enhancement in internet of things: A review. *Energies*, *14*(19), 6384.

31. Asharf, J., Moustafa, N., Khurshid, H., Debie, E., Haider, W., & Wahab, A. (2020). A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions. *Electronics*, *9*(7), 1177.

32.Hussain, F., Hassan, S. A., Hussain, R., & Hossain, E. (2020). Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges. *IEEE communications surveys & tutorials*, *22*(2), 1251-1275.

33. Al-amri, R., Murugesan, R. K., Man, M., Abdulateef, A. F., Al-Sharafi, M. A., & Alkahtani, A. A. (2021). A review of machine learning and deep learning techniques for anomaly detection in IoT data. *Applied Sciences*, *11*(12), 5320.

34. Ali, S., Saad, W., Rajatheva, N., Chang, K., Steinbach, D., Sliwa, B., ... & Malik, H. (2020). 6G white paper on machine learning in wireless communication networks. *arXiv preprint arXiv:2004.13875*.

35. Haidine, A., Salmam, F. Z., Aqqal, A., & Dahbi, A. (2021). Artificial intelligence and machine learning in 5G and beyond: a survey and perspectives. *Moving Broadband Mobile Communications Forward: Intelligent Technologies for 5G and Beyond*, 47.

36. Nurelmadina, N., Hasan, M. K., Memon, I., Saeed, R. A., Zainol Ariffin, K. A., Ali, E. S., ... & Hassan, M. A. (2021). A systematic review on cognitive radio in low power wide area network for industrial IoT applications. *Sustainability*, *13*(1), 338.

Author's Declaration Statement

I, the undersigned Lawrence Nforh CheSuh hereby declare that I co author of this article with the following:
Ramón Ángel Fernández Díaz,Jose Manuel Alija Perez, Cármen Benavides Cuellar and
Héctor Alaiz Moretón.

To the best of my knowledge, this article contains no material previously published by any other person except where due acknowledgment has been made. This article contains no material accepted as part of the requirements of any other academic degree or non-degree program, in English or any other language.
This is a true copy of the article, including final revisions.

Date: 11/02/2024

Name: Lawrence Nforh CheSuh

Signature: NCSH