AN INTELLIGENT IDS FRAMEWORK FOR IOT SYSTEMS USING MACHINE LEARNING, FEATURE SELECTION AND FUSION LEARNING

Hiren Shukla[1], Dr. Bhavesh Jaiswal[2], Mayank Gandhi[3]
[1]Ph.D Scholar, Monark University, Ahmedabad, Gujarat, India
Email: hirenshukla09@gmail.com
[2]Associat Professor, Department of Electronics and Communication,
Monark University, Ahmedabad, Gujarat, India
Email: bhavesh2g@gmail.com
[3]Lecturer, EC Electronics and Communication Engineering,
Government Polytechnic, Gandhinagar, Gujarat, India
mayank.life@gmail.com

Abstract - *Implementation of large amount of IoT devices has impacted greatly on the contemporary digital infrastructure since it provides a smooth connection between smart systems and individual sensor devices. But this massive interconnection has come with its many security glitches as well, because of limited computing capabilities and do not have powerful security mechanisms therefore they are susceptible to different cyber-attack. Traditional IDS have their own limitation. In order to overcome such obstacles, this paper suggests a smart intrusion detection model, which combines machine learning and feature selection methods as well as fusion learning techniques. The framework analyses three common ML algorithms, which are Decision-Tree (DT), Logistic-Regression (LR), and Gaussian Naïve-Bayes (GNB). In order to improve model efficiency and reduce data dimensionality, feature selection techniques including Fisher Score, RF feature importance and RFE are applied. The proposed framework is evaluated and compared with widely recognized benchmark datasets like NSL-KDD, UNSW-NB15, and CIC-IoT-2023. Experimental results shows that the Decision Tree classifier achieves a detection accuracy of 95% amongst various machine learning algorithms, while the proposed fusion learning model further improves the performance and achieves an accuracy of 97.22%. These findings indicate that the suggested framework offers a scalable and effective solution in relation to protecting the IoT networks against cyber threat.*

Keywords: *IoT, IoT Security, IDS, Network Security, Machine Learning, Feature Selection, Cybersecurity, Fusion Learning, Anomaly Detection, Network Traffic Analysis, Intelligent Security Systems.*
How to cite this article: Shukla H, Jaiswal B, Gandhi M. An intelligent IDS framework for IoT systems using machine learning, feature selection and fusion learning. Int J Drug Deliv Technol. 2026;16(74s): 869-880. DOI: 10.25258/ijddt.16.74s.105

1. INTRODUCTION

IoT has become the most powerful technologies in the current computing world. There is an ever-increasing number of touching devices that interact with each other over networks to allow applications like smart homes, healthcare monitoring, intelligent transportation and industrial automation [1]. Although it has several advantages, IoT networks are much susceptible to cyberattacks because of the scarcity of device capabilities, the heterogeneity of protocols, and the absence of standard security features [2].

The IoT has rapidly evolved into the most advance influential technologies in the modern digital era. IoT network refers as the multiple internally connection of physical devices, sensors and smart systems that connect with each other through the internet to collect, process and interchange data [1]. Due to the development of communication technologies and embedded systems, billions of IoT devices are currently implemented in the world. These devices have a broad line of applications such as healthcare monitoring systems, smart homes, intelligent transportation systems, industrial automation, environmental monitoring as well as smart city infrastructure.

This is due to the high-end capacity of the IoT devices are collecting the raw data in real-time information and offer intelligent services has greatly enhanced efficiency, automation and decision making in most sectors. Although there are great advantages associated with the IoT technology, the massive increase in the number of interconnected devices has presented major security issues as well [2]. A massive number of IoT components use a small amount of processing power, memory, and energy resources limiting them to the use of sophisticated security solutions. Also, IoT systems are frequently used in distributed and heterogeneous large-scale environments in which devices can use various protocols and networks. All these factors render IoT ecosystems very susceptible to cyber-attacks and security violations. IoT devices are

prone to hacking by cyber attackers, who frequently use such vulnerability to execute all kinds of attacks. Threats in IoT environments are common like DDoS, Botnet, malware, unauthorized access, and data manipulation [6].

As an example, IoT devices than can be compromised may be incorporated in large-scale botnets to create a huge amount of malicious traffic and interfere with network services. These attacks are not only limited to individual devices but also to the stability of whole network infrastructures. With the recent increase the vast use of various devices, the question rais, how to ensure the security of IoT networks has become an urgent research problem [6]. The Intrusion Detection Systems (IDS) is an effective security system that is extensively utilized to track network traffic and identify malicious activities. IDS solutions work by examining the behavior of the network and detecting patterns that denote possible cyber-attacks [8].

Majority traditional IDS methods are broadly divided based on the behavior or the nature of the response of the attack or simply we can say the behavior of the data. The signature-based systems respond to the attack by performing a comparison against known attack signatures that are recorded in a database. Though this is a good strategy with regards to identifying the old threats, it fails to detect newer or newer attack patterns [7]. Conversely, behavior-based attack detection, detect uncharacteristic behavior in a network, which is likely to be an attack. Nevertheless, network traffic can be complicated and unpredictable and thus designing proper behavior detection models can be difficult to design. In past few years, ML has changed the entire system. As a potential solution to increase the IDS, based on the data of historical network traffic, ML models can be automatically learn based on patterns so that anomalies can be detected which signify the possibility of cyber-attacks [7]. With machine learning, it is possible to change according to the changing attack patterns, whereas in traditional rule-based systems, such changes are not possible. Other ML algorithms such as DT, SVM, neural networks and probabilistic classifiers been effectively used in the network intrusion detection studies [11][12]. The high dimensionality of network traffic datasets is also another challenge in intrusion detection. The current net-work datasets usually have dozens or even hundreds of features, many are redundant or irrelevant to classification. Datasets with high dimensions can make the computational processes more complicated and decrease the precision of ML models [11].

The techniques of feature selection can be used to solve this issue because they can be used to extract the most informative features and eliminate redundant attributes [10]. The feature selection method reduces the dimensionality of the data sets thereby increasing performance in the model, decreases the time of training and increases detection of data. Besides picking features, the fusion of several machine learning models has also been demonstrated to enhance the results of classification. The fusion learning or ensemble learning methods combine the output of the predictions of various classifiers and yield a more accurate final decision. Fusion models are able to surpass an individual classifier as a perfection and capability by combining the capabilities of various algorithms [13].

This study is motivated by these issues and offers an intelligent intrusion detection model that is specially created to fit the IoT networks. The suggested framework incorporates both machine learning classification, feature selection methods and fusion learning methods for better efficiency in detection of intrusion [14].

Here we compare various algos as mentioned earlier, some of the feature selection methods that are used to determine the most relevant features in the network traffic datasets are: Fisher Score, RF feature importance and RFE [9][10]. The suitability of the suggested strategy is assessed with the help of the globally popular benchmark datasets such as NSL-KDD, UNSW-NB15, and CIC-IoT-2023 [3][4][5]. The experimental findings reveal that hybrid ML models coupled with feature selection yield a considerable amount of percentage of identification and minimize the wrong attack detection [13][15]. The results of the study are useful in creating more dependable and scalable IDS for IoT systems.

This article is structured with, like Section-2 is about structured overview or the content of this article, Section-3 contains the review of work done so far as per the various reference paper, on the IoT intrusion detection systems. Section 4 is the proposed techniques for higher achievement in accuracy. Section 5 covers various datasets that were used in the experiments. Section V gives the results and performance analysis of the experiment. Lastly, it is the conclusion of the paper where Section VI presents some possible directions of future research.

2. STRUCTURED OVERVIEW OF THE WORK

This paper contains various aspect which are identified from the research gap from a reference papers. Those are as follows:

- Creation of an intrusion detection system whereby machine learning is deployed in an IoT environment.
- Combination of various feature selection algorithms such as FS, RF importance as well as Recursive Feature Elimination.
- Incorporation of the fusion method of learning that involves Decision Tree, Logistic Regression, and Naive Bayes.

- Assessment of some basic IDS datasets: NSL-KDD, UNSW-NB15, and CIC-IoT.
- In depth analysis in terms of percentage achievement, Precision, Recall, F1-score, AUC and FPR.
- Novelty of such article is fusion algo with feature selection. Fusion can be made with any of available ML algo. along with various weightage of percentage in coding.

3. RELATED WORK

In the last few years data shows that major development of IoT technologies attracting the interest of numerous researchers interested in discussing the matter of cybersecurity. With the ever-growing number of IoT devices installed in different fields of application, the question of secure communication and network protection against cyber threats has become one of the most important issues. As a result, numerous researchers have aimed at coming up with effective intrusion detection systems (IDS) that can detect malicious activities within the IoT set-ups.

The UNSW-NB15 dataset that was introduced by Moustafa and Slay [3] is one of the most popular datasets to test the use of intrusion detection models. The dataset was created to overcome the drawbacks of previous datasets like KDD Cup 1999 and NSL-KDD that did not provide realistic current day attack patterns. UNSW-NB15 contain vast variety of attacks and standard traffic patterns, which is why it is appropriate to test machine learning based systems of intrusion detection. Their work emphasized the need to use more recent and actual datasets to train and test IDS models. A number of researchers have also analyzed the problems that are associated with securing IoT networks. Chaabouni et al. [6] were able to find a thorough survey of methods of intrusion detection in an environment that is specifically built on the IoT. Their paper talked about different architectures of IDS and the weaknesses of the conventional security solutions as they are implemented in resource-limited IoT devices. The authors stressed that lightweight machine learning algorithms are necessary in determining effective intrusion detection without burdening unreasonable computational load on the IoT systems.

Many IDS uses basic level of ML because it can be used to analyze big amounts of network traffic data and identify patterns of sophisticated attacks. Sommer and Paxson [7] talked about machine learning use in network intrusion detection and stressed upon the fact that data-driven methods can enhance the adaptability and precision of IDS models. Their work illuminated the possibility of the gains and practice issues that come with the implementation of ML models in the actual network security systems.

The other direction of research that is crucial is feature selection methods. The datasets used in network intrusion frequently have high number of attributes with many of them being redundant or useless in classification [11]. The data is high-dimensional and might lead to an increase in the complexity of computations and have an adverse effect on the performance of models. The solution to this problem is provided by feature selection methods whereby the most meaningful attributes which lead to sound classification are established. Fisher Score, Information Gain, and Recursive Feature Elimination are the frequently used methods of dimensionality reduction of datasets, and the enhancement of classification [9][10]. Besides the selection of features, a number of studies have also been conducted to examine ensemble and hybrid learning methods to enhance intrusion detection. The ensemble learning involves the combination of the forecast of two or more classifiers to make a more accurate one [13].

Some of these techniques which have been used with success in intrusion detection studies. Such practices tend to be more effective than single classifiers since such practices take advantage of the capabilities of the various learning algorithms.

Also, even though the findings of earlier research are promising, a lot of challenge lies with a lot of intrusion detection systems like the high level of false positive, low scalability, and the inability to identify advanced or novel attacks. Moreover, the heterogeneity of the IoT environments adds an extra complication in the process of developing effective security solutions. Thus, the necessity to design more sophisticated intrusion detection systems is on the rise and is associated with the efficient feature selection techniques and the effective machine learning models. Being driven by these issues, the current paper introduces a smart intrusion detection system, which integrates the feature selection methods with various machine learning classifiers and fusion learning algorithms. The framework proposed by combining these strategies is expected to offer high accuracy in terms of detection and lower computational complexity to offer a better security solution to the current IoT networks.

4. PROPOSED TECHNIQUE

This segment presents intelligent intrusion detection framework for the IoT device system. The framework will combine machine learning and feature selection and fusion learning to ensure that the accuracy and reliability of attack detection increase. It focuses on the technique to detect malicious behavior of data in a network, occurrences of the most efficient manner with the least number of false alarms and computational costs.

IoT networks produce extensive network traffic data which are heterogeneous. The analysis of this kind of data demands effective preprocessing, reduction of features and powerful classification methods. The given framework will deal with all of these difficulties by following a systematic pipeline

comprising with various steps: data preprocessing, feature selection, ML selection and fusion learning. The overall procedure is as shown in below Figure 1.

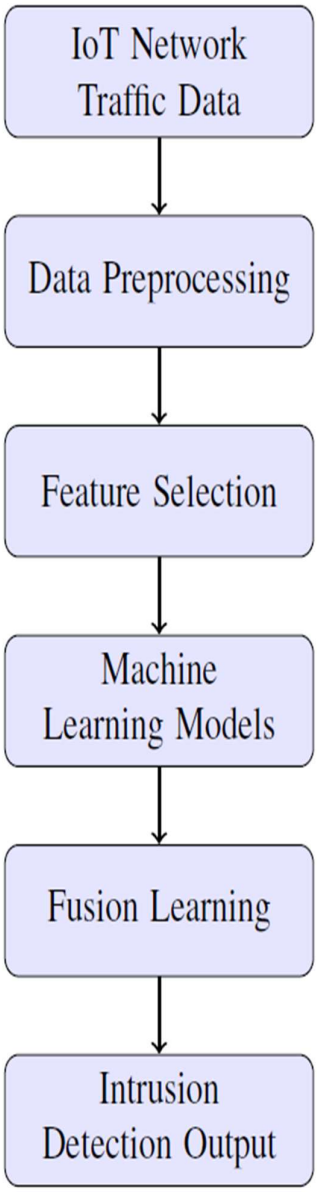


Figure 1: Architecture of the Proposed IoT Intrusion Detection Framework

As figure shows, network traffic data generated from IoT devices is first collected and processed before applying machine learning techniques for classification. Each stage of the proposed technique is described in detail in the coming subsections.

4.1 Data Preprocessing

Network traffic datasets typically contain large numbers of records with different types of features. These datasets may include missing values, redundant information, or categorical attributes can't be equally proposal processed by ML algos. Therefore, a data preprocessing step is necessarily important for better efficiency and consistency of the dataset.

In this stage, irrelevant and duplicate records are removed to reduce noise in the data. Categorical attributes such as protocol type, service, and connection flags are converted in to numerical figures using various techniques like, label encoding or one-hot encoding. Additionally, normalization techniques are applied to scale the features within a uniform range. This makes learning not to be dominated by features which have bigger numerical value.

The dataset is processed after which it is further divided into training and testing. Machine got constructed using the training dataset. The models of studying and the testing data to measure their performance is done with the help of the testing data.

4.2 Feature Selection

The datasets that are generated by the IoT networks tend to have numerous attributes that explain various features of network traffic. Nevertheless, not every feature will be of much importance to intrusion detection. Redundant and irrelevant features can be observed that would add complexity to computational operations and this may affect the performance of the model in a negative way [10].

In an attempt to solve this problem, we use feature selection methods to assert the most pertinent attributes. The three feature selection methods are used in proposed framework: Fisher Score, RF Feature Importance, and RFE. These methods help rank the features based on their contribution to classification accuracy.

Benefits of this technique include reducing the scope of dataset, improving model training speed, and enhancing classification performance. Table 1 presents some of the important features selected for intrusion detection.

Table 1: Important Network Traffic Feature

Feature Name	**Description**
Duration	Time span of network session
src_bytes	Quantity of data transmitted form source device to destination device.
dst_bytes	Volume of data received by source from destination.
Protocol Type	Communication prototype like TCP, UPD.
Service	Type of Service Provider Network's Communication.
Flag	Indication of linkage.
Count	Count of connections made by the host
svr_count	Connection's count to same service

4.3 Machine Learning Classification

After selecting the most relevant features, machine learning algorithms are applied to existing device or the system, as normal or malicious. Below are such ML algo which are used.

Gaussian Naive Bayes works on probability of the data that applies Bayes' theorem under the assumption that features follow a Gaussian distribution. It is computationally efficient and performs well in high dimensional datasets [11][14].

$$\square(\square_{\square}|\square) = \frac{1}{\sqrt{2\square\square_{\square}^{2}}}\ \square^{\left(-\frac{(\square_{\square}-\square_{\square})^{2}}{2\square\square_{\square}^{2}}\right)} \quad (1)$$

Logistic Regression (LR) uses statistical method for binary classification. It models the probability that a given input belongs to a particular class and is particularly effective when the relationship between variables is approximately linear [14].

$$\square_{(\square=1|\square)} = \square(\square^{\square}\square + \square) = \frac{1}{1+\square^{-(\square^{\square}\square+\square)}} \quad (2)$$

Decision Tree is a hierarchical model that divide dataset into sub parts based on feature values. It generates a tree like structure where each internal node represents a decision rule. Decision trees are capable of capturing nonlinear relationships and are highly interpretable [11][14].

Each model is trained with 30% of total instance using the selected feature subset, and their predictions are evaluated using the testing dataset of the sample only to show comparison with graph in better understanding way.

4.4 Fusion Learning Strategy

Although individual machine learning models can achieve good classification performance, combining multiple models often leads to improved results. Therefore, the proposed frame- work incorporates a fusion learning strategy to integrate the predictions of different classifiers.

$$\square\square\square\square\ \square\square\square\square\square\square\square\square\square\square = \square\square\square\square(\square_{\square\square}, \square_{\square\square}, \square_{\square\square}) \quad (3)$$

The fusion model uses a majority voting mechanism to combine predictions from the three classifiers. Each classifier independently predicts whether a network instance represents normal traffic or an attack. The final classification decision is determined based on the majority vote of the classifiers.

This is one of the ways to improves the attack detection system by reducing the impact of errors made by individual classifiers. As a result, the fusion model achieves higher detection accuracy and lower false positive rates.

4.5 Workflow of proposed system

The complete workflow of my experiment is as shown in below Figure 2.

The workflow begins with collecting network traffic data from IoT devices, followed by preprocessing and feature selection. Machine learning models are then trained using the selected features. Finally, fusion learning model integrates the predictions of individual classifiers to generate the final intrusion detection output.

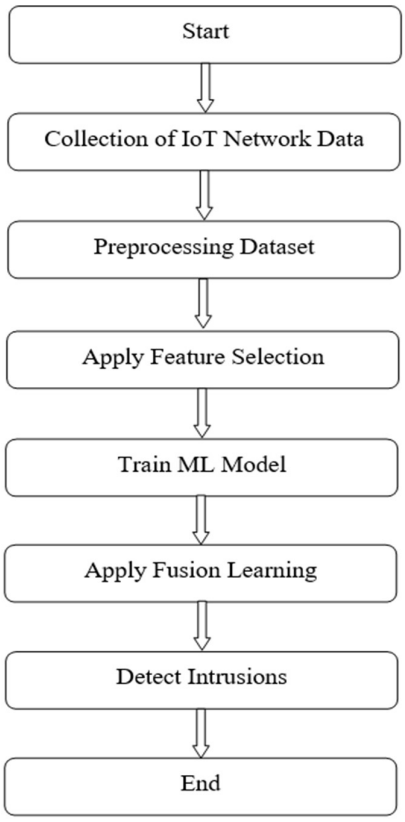


Figure 2: Workflow of the Proposed Intrusion Detection System

The combination of feature selection and fusion learning allows the proposed framework to achieve improved detection accuracy while maintaining efficiency which makes it suitable for actual world IoT security.

5. EXPERIMENTAL SETUP AND DATASET DESCRIPTION

To evaluate the efficiency of the proposed intrusion detection framework, a comprehensive set of test bench was conducted using well known benchmark datasets and multiple ML models. The ultimate goal of the experimental study was to investigate how effectively the proposed feature selection and fusion learning strategy can detect malicious activities within IoT network environments.

IoT networks generate massive volumes of heterogeneous traffic data originating from various smart devices, sensors, and network services. Detecting intrusions in such environments requires robust data processing, efficient feature selection, and reliable classification models. Therefore, the experimental setup was carefully designed to simulate realistic network intrusion detection scenarios and to ensure that the obtained results are reliable and reproducible.

The experimental setup mainly focuses on three important components: the datasets used for

evaluation, the computing environment used for model implementation, and the configuration of the machine learning models. All these elements give an integrated platform to assess the performance, scalability as well as reliability of the proposed intrusion detection framework.

5.1 Dataset Used

In order to get the results that would be reliable and impartial, three major intrusion detection datasets were referred to in this research: NSL-KDD, UNSW-NB15, and CIC-IoT [3][4][5]. Researchers often use these datasets in their cybersecurity research and machine learning research activities since a large range of network traffic patterns and various cyber-attacks is presented there.

With the help of various datasets, the proposed framework can be tested with varying network conditions and attack scenarios. It is also useful in checking whether the proposed system can generalize properly in various kinds of intrusion detection problem.

Here we have mention in earlier regarding data preprocessing. In that case we have experimented the performance based on the few sample dataset.

In NSL-KDD datasets, preprocessing will not decrease the instances that are included because that dataset is already narrowed and has few redundancies. Thus, the records are not significantly different. Nevertheless, nominal characteristics like type of protocol and service are coded into corresponding numbers and normalization is done so that the scales of variables are similar.

In reference to the UNSW-NB15 dataset, preprocessing is connected with the manipulation of different types of features and the preservation of the uniformity of data representation. Their use is also like NSL-KDD, no significant reduction in the number of instances is done because the removal of the records can cause the loss of key attack patterns. The main effects of preprocessing include higher consistency of data and transformation of features but not downsizing of data.

In the case of CIC-IoT-2023 dataset, where the number of instances is very large, preprocessing is concerned with effective processing of high-volume data. Although the overall number of instances does not change, normalization and encoding both make the dataset much more usable to machine learning models. In other instances, light filtering of corrupt or incomplete records can be done but this will not have much impact on the total size of the dataset.

Below summary shows the various dataset used the various amount of instance out of total for testing and few for training, further all graphs starting from figure 3 to 11 are from the samples value.

Table 2 : Summary for reduction in Datasets

Dataset	**Total Instances (As per Table 2)**	**Test Set Size (30%)**	**Instances Used in Confusion Matrix**	**Reason for Reduction**
NSL-KDD	125973	~37792	~1975 (sampled)	Having a basic behavioural attack sample
UNSW-NB15	257673	~77302	~1955 (sampled)	Large test data reduced for redundancy
CIC-IoT-2023	8200000	~2460000	~2010 (sampled)	Extremely large dataset; sampled need to be tested for quick result

5.1.1 NSL-KDD Dataset

One of the most common data sets used in the evaluation of the network IDS is the NSL-KDD dataset [3][4][5]. It is a better variant of other KDD-Cup-1999 dataset which was design to come over some of the weaknesses of the previous data set. NSL-KDD has one of the greatest advancements whereby redundant records which had been used to produce biased learning results have been eliminated.

The data includes the records of network connections that are categorized by the network type as normal traffic or the type of attacks. In every record, there are 41 features describing various attributes of network connection, type of protocol, type of service, count of transferred bytes and time taken to maintain the connection.

The attacks in the NSL-KDD data set can be grouped into various vast categories: DoS, Probe Attacks, R2L Attacks, U2R Attacks.

These attack categories represent different types of network threats and provide a useful benchmark for evaluating intrusion detection systems.

5.1.2 UNSW-NB15 Dataset

The UNSW-NB15 [3][4][5] dataset has been designed in such a way that it reflects the contemporary network traffic and attack patterns more closely. Older datasets have fewer recent realistic attack scenarios compared to UNSW-NB15, which has more recent and realistic attack scenarios, which capture the current threat of cybersecurity.

The dataset was created using the IXIA Perfect Storm tool in a controlled cyber range environment. It includes both normal network traffic and multiple attack types. Each network record contains 49

features that describe flow characteristics, content features, and time-based attributes.
The UNSW-NB15 dataset includes nine different attack categories, including: Exploits, Fuzzers, Backdoors, Worms, Shellcode, Analysis attacks
Because of its modern attack representation and detailed feature set, UNSW-NB15 is widely used for evaluating advanced intrusion detection systems.
5.1.3 CIC-IoT-2023 Dataset
The CIC-IoT [3][4][5] dataset is specifically designed for evaluating security solutions in Internet of Things environments. It contains network traffic generated from multiple IoT devices operating under both normal and attack conditions.
The system captures traffic from IoT devices like smart cameras, smart sensor, and other embedded components that communicate over the network. It includes a variety of IoT specific attacks, making it highly relevant for modern intrusion detection research.
Different type of attack incorporated in the CIC-IoT-23 dataset like: Distributed Denial-of-Service (DDoS), Botnet attacks, Brute force login attempts, Mirai malware traffic.
The CIC-IoT dataset holding a vast number of traffic records and more than 80 traffic-related features. This large dataset provides a realistic representation of traffic patterns observed in IoT networks.
Table 3 summarizes the key characteristics of the datasets used in this study.

Table 3 : Summary of Datasets Used for Evaluation

Dataset	Instances	Features	Attack Types
NSL-KDD	148517	41	DoS, Probe, R2L, U2R
UNSW-NB15	257673	49	Exploits, Fuzzers, Worms
CIC-IoT-23	46.7 million	80+	DDoS, Botnet, Brute Force

Using multiple datasets helps ensure that the proposed intrusion detection framework is tested across a variety of network environments, thereby improving the reliability and generalizability of the experimental results.

5.2 Experimental Environment

The entire experimentations were done on a common computing environment that would support machine learning and data analytics activities. The suggested intrusion detection model was realized on the Python programming language that offers a broad array of libraries in machine learning and data processing.
A number of popular Python packages were used in the experiments. Machine learning algorithms were implemented with the help of the Scikit learn library, data preprocessing and manipulation were conducted with the help of Pandas, and numerical computations were conducted with the help of NumPy. Moreover, the experimental results were visualized using Matplot lib and Seaborn.
The experiments have been conducted on a system with Intel Core processor, 16 GB RAM and windows operating system. This setup was adequate to train the machine learning models and process the datasets.
The entire preprocessing, feature selection, training and testing were conducted within the same environment to have consistency and reproducibility of the experimental findings.

5.3 Model Training and Configuration

Once the data preprocessing and feature selection phases are finished, three ML types: DT, LR and GNB were trained on the selected features.
These algorithms have been chosen as they represent various kinds of classification methods and are very popular in intrusion detection studies.
Decision Tree is a rule-based classifier, which builds a tree structure that is hierarchical in the sense that it attempts to classify data instances. LR can be consider as a easier model, which approximates the likelihood of belonging to a class with the help of a logistic function. GNB is a probabilistic classifier, which assumes that features are independent and independency of features is modeled as Gaussian distributions. In order to test the performance of the model, the dataset was split into the two subsets of training and testing in the standard of 70:30. The machine learning models were trained on the training set and tested on data that was not seen before on the testing set [11][14].
Table 4 presents the major configuration parameters applied in each machine learning model.

Table 4 : Machine Learning Model Parameters

Model	Parameter	Value
Decision Tree	Max Depth	10
Decision Tree	Criterion	Gini Index
Logistic Regression	Solver	lbfgs
Logistic Regression	Max Iterations	200
Gaussian Naive Bayes	Distribution	Gaussian

These parameters have been chosen depending upon the typically used parameters in the machine learning studies and initial experimental observations to provide the performance of the model being stable and reliable.

5.4 Evaluation Metrics

In an attempt to determine how effective the proposed intrusion detection framework is, a number of standard evaluation metrics were employed. These measures are useful in determining the ability of the models to recognize the difference

between normal traffic activities in the network and those that are malicious.

The primary measures of evaluation employed in this project are:

Accuracy: This measure is the general percentage of the correctly classified data in the dataset [11][12][13].

$$\square\square\square\square\square\square\square\square = \frac{\square\square + \square\square}{\square\square + \square\square + \square\square + \square\square} \quad (4)$$

Precision: This measures the rate of the correct isolation of attack cases, out of all the cases that have been correctly predicted as attacks [11][12][13].

$$\square\square\square\square\square\square\square\square\square = \frac{\square\square}{\square\square + \square\square} \quad (5)$$

Recall (Detection Rate): Recall is used to determine the extent to which the model is able to identify actual attack cases within the dataset [11][12][13].

$$\square\square\square\square\square\square = \frac{\square\square}{\square\square + \square\square} \quad (6)$$

F1-Score: The F1-Score having a mean of precision and recall and gives a balanced measure of the performance of the various class of bit stream [11][12][13].

$$\square 1 - \square\square\square\square\square = 2 \times \frac{\square\square\square\square\square\square\square\square\square \times \square\square\square\square\square\square}{\square\square\square\square\square\square\square\square\square + \square\square\square\square\square\square} \quad (7)$$

False Positive rate (FPR): This is a measure that involves the percentage of normal traffic that is wrongly configured as malicious [8][13].

$$\square\square\square = \frac{\square\square}{\square\square + \square\square} \quad (8)$$

These measures present a holistic measurement of the performance of the model. Although the accuracy is measure of the correctness, the precision and the recall measure allow a better understanding of how good the IDS is to detect unknown traffic and reduce the false alarms.

The following section shows the output of experimental and the detailed analysis of the proposed intrusion detection framework in all the three datasets.

5.5 Detailed Performance Analysis

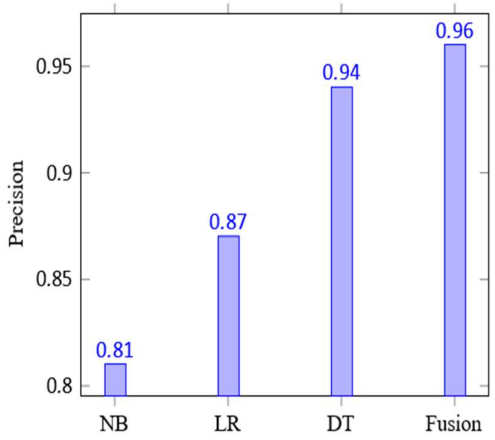
Various visualization methods were also employed in the further analysis of the proposed intrusion detection framework to gain a closer insight into the behavior of each machine learning model. The visual analysis will be significant in determining the trends, performance of models, and effectiveness of proposed fusion learning approach.

In the following subsections, we have shown various graphical displays, which outline the classification accuracy, detection capability, error distribution and feature importance of various machine learning models and datasets.

The dataset is too large so its difficult to show the results for such large dataset in a graph, that's why we have displayed the sample results in a graphical form.

5.6 Precision Comparison

Precision measured in percentage for actual predicted attack instances from all the predicted attack. A larger value of precision means that the classifier has less false alarms.



Above result shows that the fusion technique is effective at reducing false attacks prediction as the model has the highest precision.

5.7 Recall Comparison

It is also known for the detection rate, measures the efficiency for rightly detection of intrusion which is consider to be an actual attack instance.

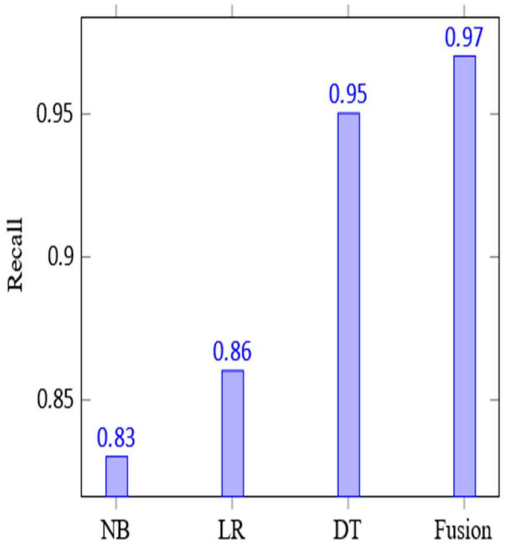


Figure 4: Recall Comparison

The Decision Tree algorithm and Fusion model's algorithm gets significantly higher recall values compared to Naive Bayes and Logistic Regression machine learning algorithms.

5.8 F1 Score Comparison

The F1-score provides a balanced measure between precision and recall parameters.

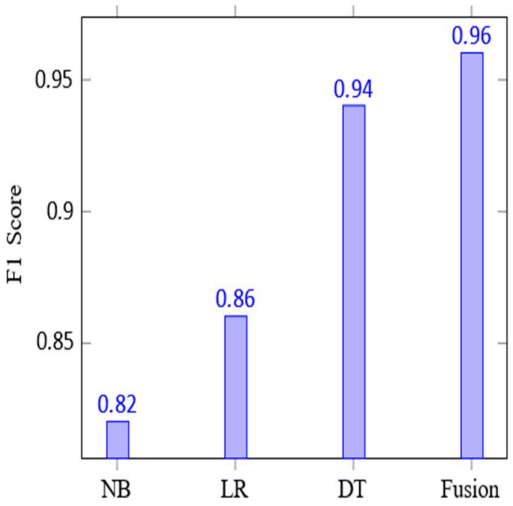


Figure 5 : F1 Score Comparison

The fusion model always attains the optimum balance between detection and prediction reliability.

5.9 False Positive Rate Comparison

False positive rate is an essential parameter within an IDS since it shows the frequency of treating normal traffic as malicious.

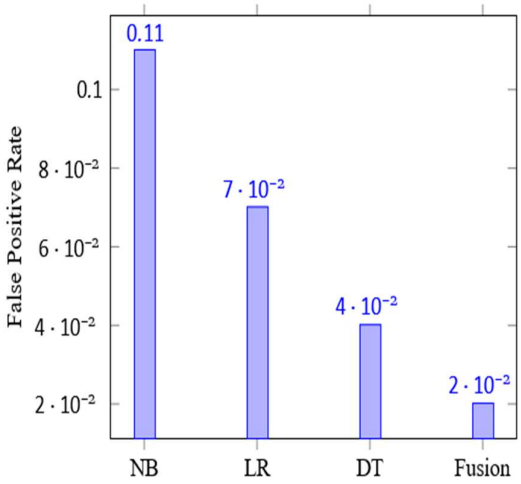


Figure 6 : False Positive Rate Comparison

Fusion learning model also shows a huge decrease in false alarm as compared to single classifiers.

5.10 Dataset Accuracy Comparison

This experiment compares the model accuracy of the three datasets which were utilized in this study.

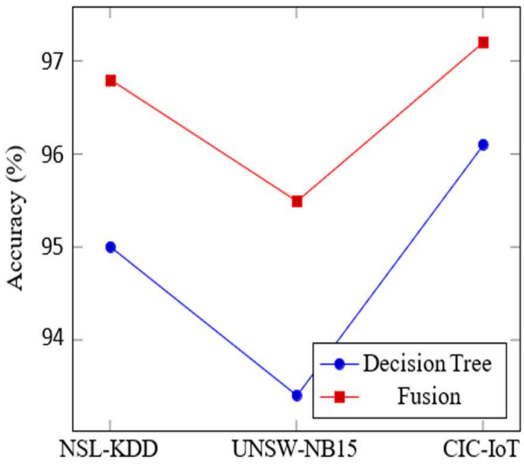


Figure 7 : Accuracy Comparison Across Datasets

Fusion model performs better than individual classifiers in all the datasets.

5.11 AUC Comparison

It shows the differentiate between normal and attack traffic, we use Area Under Curve (AUC). The fusion model has the largest AUC value and this shows that it is a better model in classification.

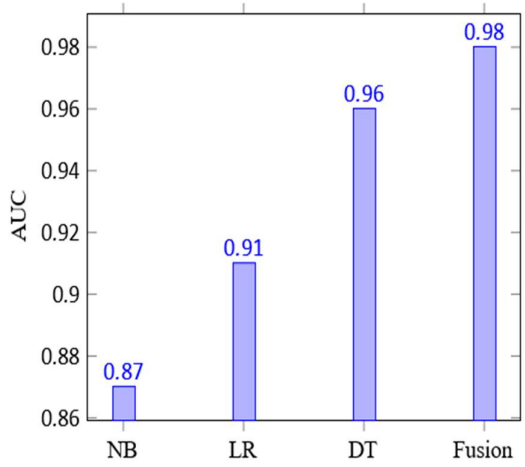


Figure 8 : AUC Comparison

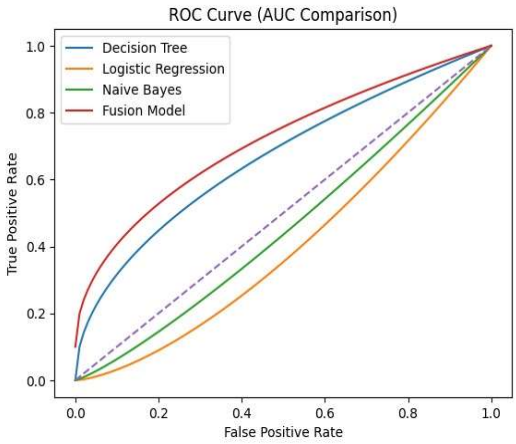


Figure 9 : ROC Curve

5.12 Training Time Comparison

Another aspect of consideration during the evaluation of ML models to a real-world intrusion detection system is training time.

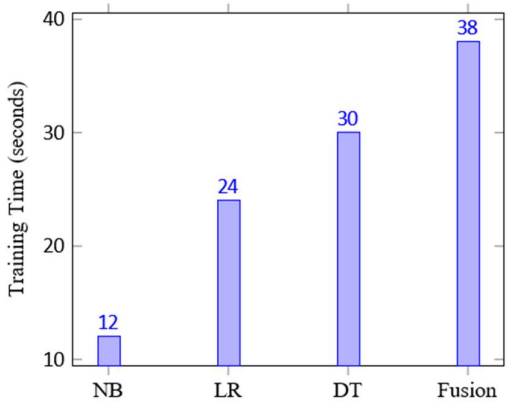


Figure 10: Model Training Time

The fusion model has a disadvantage that it has a little longer training time but this does not justify the extra cost of computation since the performance will be better.

5.13 Detection Rate Across Attack Types

In this experiment, the detection performance of various types of attacks is analyzed. The fusion model shows enhancement in the capability of detection in all categories of attacks.

5.14 Confusion Matrix Visualization

Confusion matrix visualization was also analysed to learn more about classification behavior.

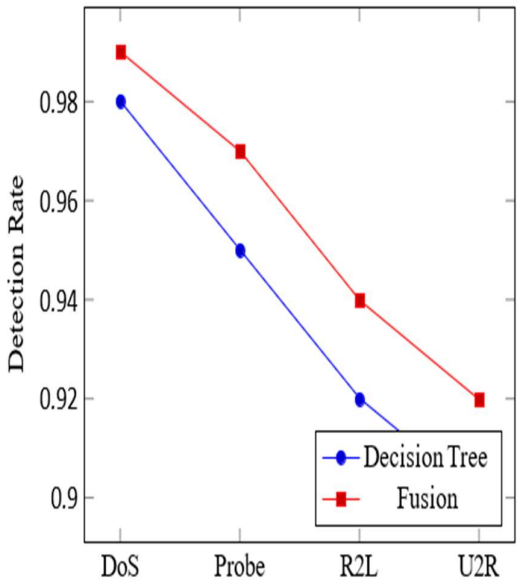


Figure 11: Detection Rate Across Attack Categories

5.15 Confusion Matrix Analysis

Going further with classification capability of the shown intrusion detection framework, the confusion matrices were created the best efficient model, which is fusion learning. Here confusion matrix can give a more detailed description of the results of prediction by comparing the actual with the predicted attacks of various classes.

Every confusion matrix includes four important elements which are True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) [11][12]. TP are the number of attack cases that have been correctly identified by the model and true negatives are the number of normal traffic that has been correctly identified. The FP happen when a normal traffic is mistaken as an attack and false negatives happen when an attack traffic is mistaken as normal traffic.

As mentioned earlier in table-2 that each dataset used here have a large number of instances, physically in a normal computer its not possible to apply all the algo on whole dataset, So, a sample dataset has been taken for consideration for purely quick result. Here below confusion matrices are generated on a representative sampled sub-set of the test data to enhance readability, minimize the complexity of the computation.

5.15.1 Confusion Matrix for NSL-KDD Dataset

Table 5 shows the confusion matrix that was achieved with the proposed fusion model on the NSL-KDD dataset.

Table 5: Confusion Matrix for NSL-KDD Dataset (Fusion Model)

Actual/ Predicted	Normal	Attack	Accuracy
Normal	940	45	95.43%
Attack	35	955	96.46%

The findings indicate that this model has detected majority of attack cases and has a fairly low false positive rate. There were few cases of normal cases which were mistaken to be malicious traffic.

5.15.2 Confusion Matrix for UNSW-NB15 Dataset

Table 6 reveals the confusion matrix of the UNSW-NB15 dataset.

Table 6: Confusion Matrix for UNSW-NB15 Dataset (Fusion Model)

Actual/ Predicted	Normal	Attack	Accuracy
Normal	915	60	93.84%
Attack	42	938	95.71%

The UNSW-NB15 dataset contains more complicated patterns of attacks compared to the NSL-KDD one. Although such complexity increases, the fusion model is also exhibiting a high detection rate with a large percentage of the correctly classified instances.

5.15.3 Confusion Matrix for CIC-IoT Dataset

Table 7 presents the confusion matrix of the CIC-IoT dataset.

Table 7: Confusion Matrix for CIC-IoT Dataset (Fusion Model)

Actual/ Predicted	Normal	Attack	Accuracy
Normal	970	30	97%
Attack	28	982	97.22%

Above results show that mentioned fusion technique is specifically effective in IoT setups. This model appropriately detects most attack cases and gives

small number of false detections. This goes to show that the intrusion detection framework proposed can be useful to differentiate between normal and malevolent traffic in the IoT networks.

The analysis indicated in the confusion matrix proves the usefulness of the suggested intrusion detection framework in all three datasets. In both instances, correct count of classified instances is much more than the incorrectly classified instances. The CIC-IoT dataset exhibited the best detection accuracy in the three datasets, meaning that the feature selection and fusion learning method proposed is especially suitable in the environment of the IoT networks. It is also shown in the results that using combination of multiple classifiers can reduce false positives and false negatives and enhance reliability of intrusion detection system.

5.16. Experimental outcome

These experimental outcomes explain clearly the benefits of combining the feature selection methods with a fusion learning method. The Decision Tree model is one of the individual classifiers that could consistently give good performance because it is able to describe complex nonlinear relationships in network traffic data.

Nevertheless, the discussed fusion learning model also enhances the performance of the detector by integrating the results of several classifiers. This hybrid solution minimizes the error of classification and makes the IDS stronger.

In all the datasets, the fusion model will have the greatest values of accuracy, precision, recall, and AUC with a low false positive rate. These findings affirm that the proposed IDS can be successfully used to detect cyber-attack in both the conventional network settings and contemporary IoT systems.

6. CONCLUSION

In this study, Implementation of ML for IDS in the IoT networks with the help of feature selection and fusion learning algorithms was proposed, which achieve 97.22% of accuracy in detection of attack. The results of the experiment proved that Decision Tree classifier was the most efficient in terms of the single machine learning models but the fusion model giver higher accuracy and can consider as the most successful in terms of overall detection amongst this experiment setup.

The methods of selection of features showed considerable enhancement on the performance of models through dimensionality reduction and removal of redundant features in the dataset. The framework suggested is good in identifying various forms of network attack and lowering false positives.

Adopted technique gives better results compared to other models that are referred from literature. An example is the ensemble-based intrusion detection system which has demonstrated good performance but at low accuracy levels [13]. Equally, RF based model have been said to be 93% accurate in intrusion detection activities [15], whilst ML based IoT IDS have reached accuracy levels of almost 95% per cent [14]. Conversely, the suggested framework has a better accuracy of 97.22%, which can be regarded as a sign of its efficiency IDS.

7. Future Scope

Future studies will aim at using Deep Learning (DL) models could be used for the detection of intrusions to enhance the accuracy of intrusion detection even more. Additionally, real-time implementation of the proposed system in large scale IoT environments will be explored.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] A. Alrawais, A. Alhothaily, C. Hu, and X. Cheng, "Fog Computing for the Internet of Things: Security and Privacy Issues," IEEE Internet Computing, vol. 21, no. 2, pp. 34–42, 2017.

[3] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in Proc. Military Communica- tions and Information Systems Conference (MilCIS), 2015.

[4] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," in Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009.

[5] I. Sharafaldin, A. Lashkari, and A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in Proc. International Conference on Information Systems Security and Privacy, 2018.

[6] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network Intrusion Detection for IoT Security Based on Learning Techniques," IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2671–2701, 2019.

[7] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in Proc. IEEE Symposium on Security and Privacy, 2010.

[8] H. J. Liao, C. H. Lin, Y. C. Lin, and K. Y. Tung, "Intrusion Detection System: A Comprehensive Review," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 16–24, 2013.

[9] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[10] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene Selection for Cancer Classification Using Support Vector Machines," Machine Learning, vol. 46, pp. 389–422, 2002.

[11] C. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.

[12] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[13] J. Ashraf, M. K. Afzal, M. Sharif, and A. Rehman, “An Ensemble Learning Approach for Intrusion Detection Systems,” Future Generation Computer Systems, vol. 86, pp. 741–751, 2018.
[14] A. Verma and V. Ranga, “Machine Learning Based Intrusion Detection Systems for IoT Applications,” Wireless Personal Communications, vol. 111, no. 4, pp. 2287–2310, 2020.
[15] N. Farnaaz and M. A. Jabbar, “Random Forest Modeling for Network Intrusion Detection System,” Procedia Computer Science, vol. 89, pp. 213–217, 2016.