

ASSESSMENT OF HYBRID MACHINE LEARNING ALGORITHMS FOR INTRUSION ANOMALY DETECTION IN CYBERSECURITY USING MATHEMATICAL MODELLING AND SIMULATION

Naseem Afzal Qureshi¹, Sana Batool², Muhammad Ali Khan³, Muhammad Zohaib Khan⁴,
Muqaddas Salahuddin⁵, Hussain Bux Marri⁶, Faisal Hassan^{*7}

¹Department of Computer Science, University of Karachi, Karachi-75270, Sindh, Pakistan.

²Faculty of Computer Science, Riphah International University, Faisalabad-55000, Punjab, Pakistan.

³Department of Industrial Engineering and Management, Mehran University of Engineering & Technology, Jamshoro-76062, Sindh, Pakistan.

⁴Department of Information Technology, Shaheed Mohtarma Benazir Bhutto Institute of Trauma (SMBBIT), Karachi-74200, Sindh, Pakistan.

⁵Faculty of Computer Science and Information, Technology, Superior University, Lahore-54000, Punjab, Pakistan

⁶Department of Mechanical Engineering Technology, The Benazir Bhutto Shaheed University of Technology and Skill Development, Khairpur Mir's-66020, Sindh, Pakistan

⁷Department of Mathematics, Faculty of Science, University of Karachi, Karachi-75270, Sindh, Pakistan

¹qureshinaseemafzal@gmail.com, ²sanabatoolofficial7@gmail.com, ³zohaib_khan2017@yahoo.com,

⁴muhammad.nagar@faculty.muett.edu.pk, ⁵muqaddassalahuddin60@gmail.com,

⁶hussain.marri@bbsutsd.edu.pk, ⁷hasssanfaisal@gmail.com

DOI: <https://doi.org/10.5281/zenodo.21294407>

Keywords

Cyber Security, Machine Learning, Mathematical Modelling, Simulation, Intrusion Anomaly Detection Systems (IADS), Fuzzy C-Means (FCM), Support Vector Machines Linear (SVML), Multi-layer Perceptron (MPL), Stochastic Gradient Descent (SGD) Algorithms.

Article History

Received: 30 January 2026

Accepted: 06 March 2026

Published: 31 March 2026

Copyright @Author

Corresponding Author: *

Faisal Hassan

Abstract

The primary objective of an Intrusion Anomaly Detection System (IADS) is to identify and notify users of suspicious network or system activities that deviate from normal behaviour and may indicate cybersecurity attacks. Cyber threats have become increasingly sophisticated with the rapid growth of IoT and network technologies, highlighting the need for effective Machine Learning (ML)-based intrusion detection techniques. This study evaluates and compares the performance of classification algorithms for anomaly detection in IADS. A Kaggle dataset containing 42 features and 25,192 instances of normal and attack traffic was used, with 75% of the data allocated for training and 25% for testing. Feature selection and Fuzzy C-Means (FCM) clustering were applied before classification using Support Vector Machine Linear (SVML), Multi-Layer Perceptron (MLP), and Stochastic Gradient Descent (SGD) algorithms. The proposed hybrid models achieved classification accuracies of 99.74% with SVML, 99.53% with MLP, and 98.87% with SGD, outperforming existing approaches. The findings demonstrate that integrating FCM with supervised ML algorithms significantly enhances intrusion detection performance and provides an effective framework for developing accurate and reliable cybersecurity applications capable of adapting to evolving cyber threats.

1 Introduction

Machine learning and artificial intelligence, particularly deep learning, have become powerful approaches for solving complex problems by learning patterns from large datasets without explicit programming. In software development, these techniques play an important role in predicting defects before deployment, helping organizations improve project planning, optimize resources, estimate maintenance efforts, and maintain software quality [1] [2] [3]. Machine Learning & AI approaches, including ANN, SVM, and K-Nearest Neighbor Algorithms, are widely used for risk management, prediction, and classification in improving accuracy while optimizing time, cost, and resources across multiple domains [4]–[7]. The growing role of Machine Learning and Deep Learning is being observed in solving complex prediction and classification problems across various domains. The techniques such as K-Means Clustering, Multi-Layer Perceptrons (MLPs), Decision Trees (DT), and Naïve Bayes (NB) have been widely applied in areas like COVID-19 detection, software defect prediction, sentiment analysis, and student performance evaluation [1]–[3], [8]–[11]. Machine Learning and Deep Learning techniques, including PCA, data mining, Fuzzy Logic, Logistic Regression (LR), Decision Trees (DT), and Naïve Bayes (NB) are also evaluated to compare their effectiveness in detecting banking fraud anomalies and improving fraud detection accuracy [12]. The internet has become an essential part of modern life, providing global connectivity, economic growth, and easy access to information. However, the rapid growth of internet usage and emerging technologies has also increased cybercrime risks, making cybersecurity a major concern for protecting personal and organizational data [13]. The growing reliance on digital services such as mobile banking, e-commerce, and social networks has made cybersecurity essential for protecting private and sensitive data. As cyber threats continue to rise, strengthening data networks, enforcing cyber laws, and increasing public awareness are crucial for ensuring economic stability, national security, and safe online

communication [14]. Machine Learning (ML) has become an important approach for identifying cyber-attacks on infrastructure and devices, using datasets like KDDCUP99 and algorithms such as Decision Tree (DT), Gradient Boost (GB), Naïve Bayes (NB), and Random Forest (RF) to build effective intrusion detection systems. Cybersecurity and network security have become essential for protecting sensitive data and ensuring secure communication with the increasing importance of connectivity in modern life [15]–[19].

The increasing adoption of online banking, e-commerce, and other digital services has heightened the importance of cybersecurity, as users require secure and reliable platforms to protect sensitive financial and personal information [20]–[22]. Customer satisfaction in commercial banking is strongly influenced by service quality, staff commitment, and a secure banking environment, highlighting the importance of reliable digital banking systems [23], [24]. The adoption of e-commerce increased significantly during the COVID-19 pandemic, with users' purchase intentions being influenced by perceived control, external pressure, and social norms, highlighting the growing reliance on secure digital platforms [25]. Intrusion detection system was initially created by J. P. Anderson in 1980 which sparked intrusion detection interests [26]. Current intrusion detection systems (IDSs) still face challenges in improving detection accuracy, reducing false alarms, and identifying unknown attacks, leading researchers to explore advanced Machine Learning (ML) and Deep Learning (DL) approaches for enhancing anomaly-based intrusion detection. [27]–[28]. An Intrusion Anomaly Detection System (IDAS) plays a critical role in network security by monitoring and analyzing activities to identify potential threats. However, existing systems often suffer from high false alarm rates and limitations in detecting new attack patterns, which has encouraged the development of more accurate hybrid models combining Fuzzy C-Means (FCM) with machine learning algorithms such as SVM, SGD, and MLP for improved intrusion

detection [29]–[33]. The rapid growth of technology and IoT networks has increased security challenges, motivating researchers to develop advanced methods for detecting and preventing cyberattacks [34]. Cybersecurity and Intrusion Anomaly Detection Systems (IADS) are critical challenges in IoT environments, where inadequate security measures can expose interconnected devices to various cyberattacks. As IoT adoption continues to grow, users often prioritize its benefits over privacy concerns. Machine Learning (ML) and Deep Learning (DL) techniques, particularly classification and clustering methods, have significantly improved anomaly detection in IoT systems. Since the introduction of IADS as a generic intrusion detection framework by Denning, researchers have continuously developed automated intrusion detection techniques to address the increasing number of Internet-connected devices, which exceeded 26 billion worldwide, an upsurge in both the diversity and volume of cybersecurity threats is expected [35].

The main objectives of this research are a) To analyze and compare the performance of different machine learning algorithms for IoT intrusion anomaly detection and develop a hybrid clustering-classification framework; b) To integrate Fuzzy C-Means (FCM) clustering with supervised machine learning algorithms (SVML, MLP, and SGD) to improve intrusion detection performance and c) To evaluate the

proposed hybrid framework using standard performance metrics, including accuracy, precision, recall, and F1-score, and compare its performance with existing approaches. The investigation of unsupervised and supervised learning approaches is done to discover IADS, with the implementation of FCM and prediction techniques using SVML, MPL and Stochastic Gradient Descent (SGD) algorithms. This research implements unsupervised and supervised ML algorithms combination. Utilizing these approaches in classification techniques, clustering, and linear regression. The mentioned algorithms have achieved a justified accuracy.

2 LITERATURE REVIEW

Machine learning is a scientific discipline that explores the study and development of mathematical algorithms that can construct functional relationships between quantities on the basis of known information and rules [36]. Machine learning tasks are mainly divided into classification, regression, and clustering which are depicted in Figure 1. Classification predicts categorical outputs by assigning data into predefined classes, regression predicts continuous numerical values, and clustering groups similar data points into clusters without predefined labels. These techniques are widely used for prediction, pattern recognition, and decision-making across various domains [36], [37]

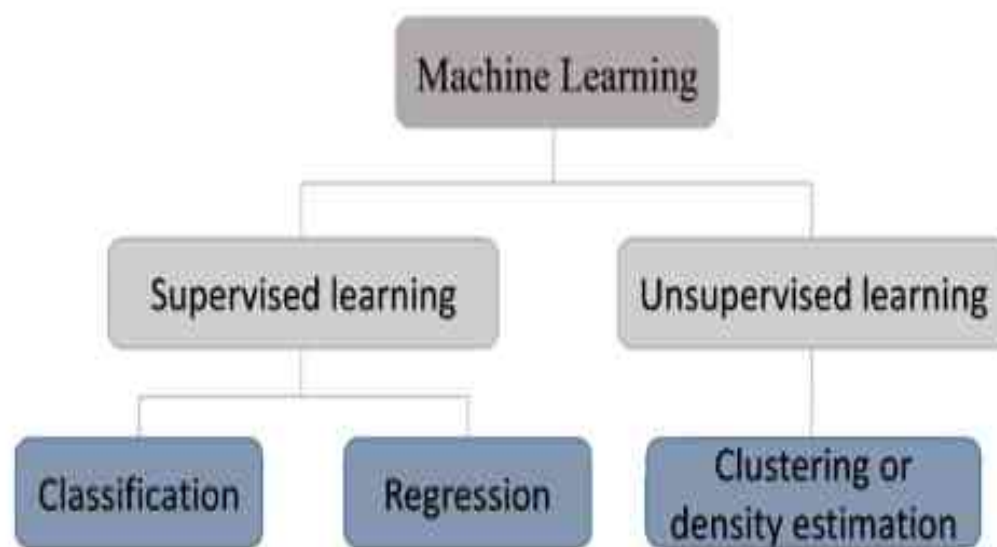


Figure 1: Machine Learning source: [38]

Classification is a supervised machine learning technique that assigns structured and labeled data to predefined categories based on shared characteristics. Common classification algorithms include Naïve Bayes (NB), Linear Classifier, and Logistic Regression. Figure 2 illustrates the classification techniques used in various applications. Linear and non-linear regression algorithms used in unsupervised as well as supervised methods, accordingly. due to the different types of relationships between the

dependent and independent variables in different regression models. It carries out regression tasks. As shown in Figure 3, regression models utilize both structured and unstructured data to predict continuous outcomes through linear and non-linear techniques. Clustering, one of the most widely used unsupervised learning methods, groups similar data based on inherent patterns to uncover hidden structures within datasets. The clustering process is illustrated in Figure 4

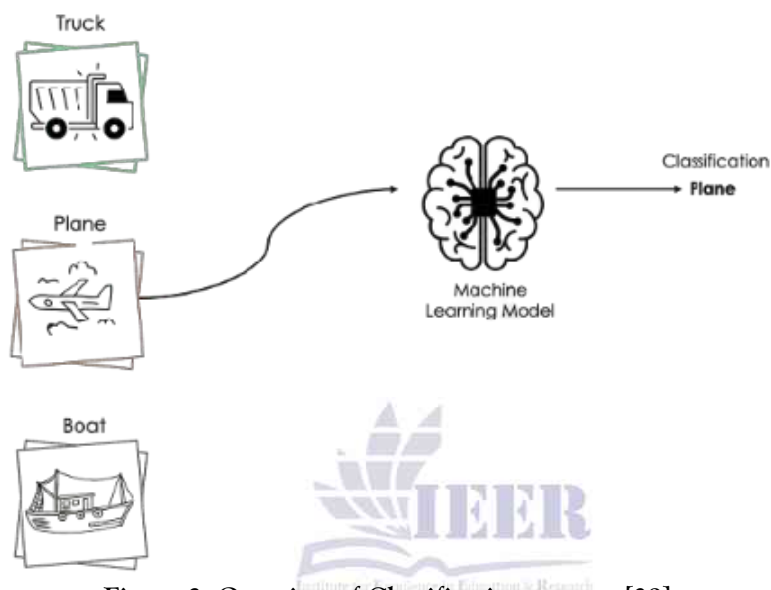


Figure 2: Overview of Classification source: [39]

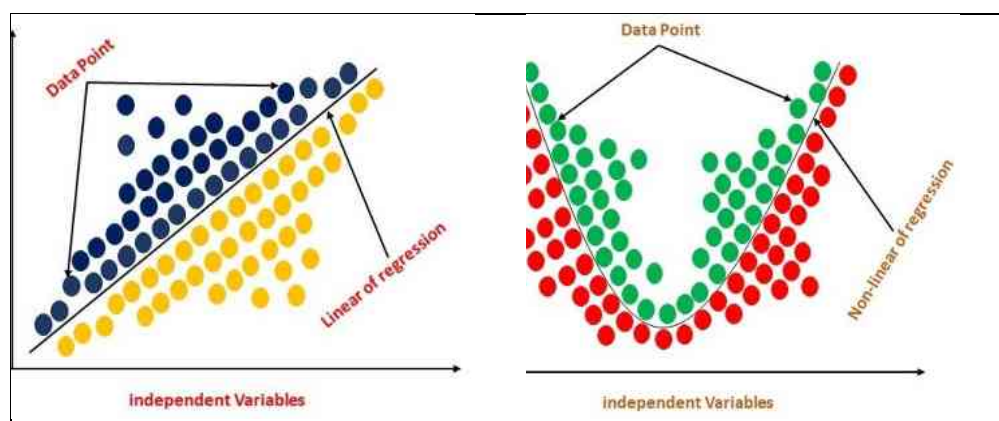


Figure 3: Overview of Linear regression & non-linear regression Models source: [40], [41]

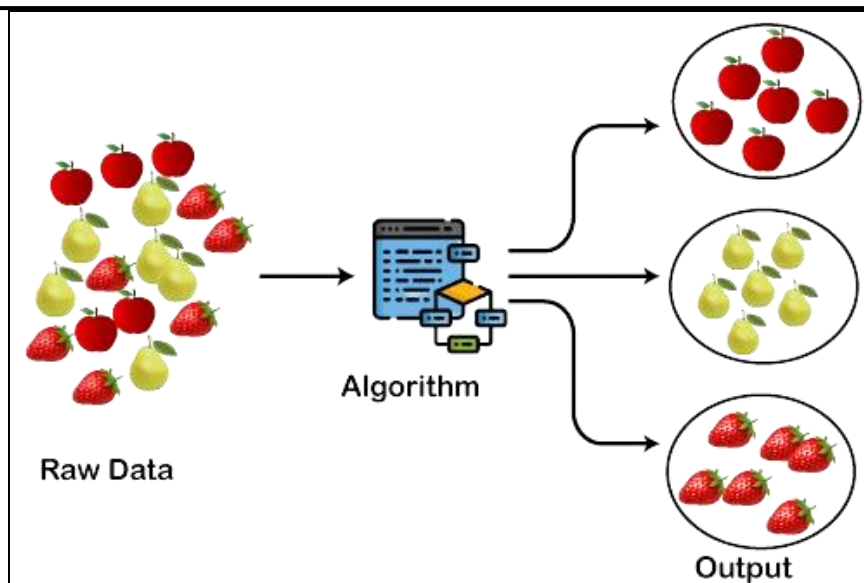


Figure 4: Clustering source: [42] [43]

Kalhor et al. (2019) found that customer satisfaction in commercial banking depends on secure, reliable, and high-quality banking services, emphasizing the need for robust cybersecurity mechanisms to maintain customer trust in digital financial transactions [23], [24]. Ali et al. (2021) reviewed the challenges of online banking from the customers' perspective, highlighting the importance of secure electronic transactions and user awareness [20]. Gul et al. (2021) further investigated the infrastructural challenges of e-banking, emphasizing that secure and reliable digital infrastructures are essential for improving customer trust and ensuring safe online financial services [21], [22]. Junejo et al. (2023) demonstrated that the rapid growth of e-commerce has increased users' dependence on digital platforms, emphasizing the need for secure online transactions and effective cybersecurity mechanisms to maintain user confidence [25]. The widespread adoption of online banking and e-commerce has increased the demand for secure and reliable digital services. Ensuring transaction security, infrastructure reliability, and customer trust has become essential for maintaining confidence in digital platforms and protecting sensitive financial information [20]–[25].

Several types of research conducted have used different ML approaches for enhancing the accuracy and efficiency of detection systems helping in securing the network security. Now artificial intelligence techniques are employed

for finding the IOT assaults more reliably. The difference in assault routes is also recognized by artificial intelligence. A problem for security solutions in dealing with IoT attacks was that attackers made minute changes to prior breaches that safety measures failed to detect. By investigating network data, researchers and developers use the power of artificial intelligence (AI) to prevent any threats to the IoT ecosystem [44], [45]. Machine and deep learning are increasingly being used in security solutions to sense these types of attacks powerfully. DL is such AI advancements that is used to manage complex nonlinear data in a variety of real-world applications. To identify botnet assaults via IoT devices, DRNN (Deep Recurrent Neural Network) is employed [46], [47]. Zaman et al. (2021) conducted a comprehensive comparative study on Relational and Non-Relational DBMS, focusing on SQL and NoSQL databases. Their research evaluated key performance metrics such as speed, efficiency, loading, response, and retrieval times by applying AI and ML techniques, including K-Means, KNN, and SVM, to assess the overall effectiveness of both database systems [11]. Khan et al. (2022) emphasized the important role of simulations in providing a controlled environment to evaluate the performance of selected algorithms under different conditions. Their study aimed to develop more accurate models and highlight the strengths of AI and ML techniques in comparing SQL and NoSQL

databases in terms of speed, efficiency, and overall performance, offering practical insights for optimized database selection [48], [9]. Khan et al. (2023) conducted a comprehensive performance analysis of various ML and DL techniques, including PCA, data mining, Fuzzy Logic, LR, DT, and NB, for detecting anomalies in credit card fraud. Their study aimed to compare these techniques to improve fraud detection accuracy and evaluate their effectiveness using performance metrics such as accuracy, precision, recall, and F1-score [12]. Nonetheless, while the dataset in this paper is extensively used in the security related cyber sector, it is not a viable option for the IoT situations [49]. In this research paper, the authors proposed improved cyber-attacks prediction and execution method used Neural network (NN), RF, NB, DT, GB and Multilayer perceptron algorithm and compared another ML algorithm [50]. The goal of this study conducted by the researchers was to assess the effectiveness of several ML algorithms in order to precisely forecast cyberattacks and abnormalities on Internet of Things (IoT) devices [51]. The results are that RF has recorded the best results and its techniques are ANN, DT, LR, RF, and SVM. In this research paper, authors reported about the Detection of Anomaly in Internet of Things devices [52]. The results of the merger between RF and DT provided best results and the techniques used in it are DT, SVM, NN, and RF. The authors suggested functionality difficulties with machine learning for identifying abnormalities in Internet of Things devices in this research [53]. The Random Forest algorithm attained the optimal results and the methods used are SVM, RF, NB, and DT. The stated identification of unusual behaviours that may indicate the presence of cyberattacks is made in study [54]. The result is that the RF achieves the optimum outcome. The techniques LR, SVM, RF, KNN, and XGBoost (XGB). Li et al. (2020) mentioned that despite the application of ML techniques, consistent and reliable results have not yet been achieved, indicating the need for further refinement of these approaches [55]. In this research paper, [56] the recommended anomaly detections and attack uses UNSW-NB 15 dataset in IoT devices. The result is that AD-IoT

is successful in reaching well results and the algorithm used is RF. In this research paper, authors proposed the IADS for IoT Botnet through the use of their selected dataset [57]. The results are these three ML classifiers performed well and the methods used are LR, SVM and RF. In this research paper, the authors proposed for the improvement of through IoT security by using ML with BoT-IoT dataset. RF and CNN have demonstrated the highest levels of accuracy, recording superior results in comparison to other techniques & models [58]. Hassan et al. (2023) examined the performance of SC using ML by integrating K-Means with LR, RF, and DT algorithms. The findings showed that K-Means + LR achieved the highest accuracy of 99.98%, followed by K-Means + RF at 99.906% and K-Means + DT at 99.83%, highlighting the effectiveness of hybrid models in enhancing classification accuracy [3]. Shaikh et al. (2023) investigated ML techniques for SDP using MMS by integrating K-Means with SVM, RF, and MLP algorithms. The findings revealed that K-Means + MLP achieved the highest accuracy of 99.67%, followed by K-Means + SVM at 99.19% and K-Means + RF at 97.76%, demonstrating the effectiveness of hybrid models in improving software defect prediction [1]. Baig et al. (2023) investigated the effectiveness of ML and DL techniques in predicting student success in HEIs. These algorithms were evaluated individually and in combination, the study demonstrated their potential to enhance academic performance prediction and support effective educational decision-making [8]. The enhancement of IoT security by the testing of various ML techniques on the IoT dataset was put forward in this study [59]. When employing KNN algorithms correctly, their findings are accurate. The techniques used are GNB, LR, GRU, Simple RNN, CNN, CNN-LTSM, AdaBoost, RF, KNN, DT, DNN and LSTM. In their research paper, Rani and Kaushal mentioned that the purpose is to Strengthen the Intrusion Detection System (IDS) by enhancing its security and accuracy [60]. The results are a proposed simulation with less time detection and algorithms employed are KNN, NB, DT, Logistics Regression (LR) and RF. In this research paper, [61] proposed anomaly

detections and attack is using the IoT-23 dataset in IoT devices. The RF algorithm has obtained favourable results and the techniques used were AdaBoost, NB, RF, SVM, and MLP. Siddiqui et al. (2023) investigated both supervised and unsupervised ML approaches by integrating K-Means clustering with MLP, DT, and NB to improve COVID-19 case detection. Their findings demonstrated high accuracy, highlighting the effectiveness of hybrid learning models. Similarly, the integration of FCM with SVML, MLP, and SGD further emphasizes the potential of combining clustering and classification techniques for accurate IADS detection [2]. Al-Janabi et al. (2021) reviewed

the application of ML classifiers, including NN, NB, DT, and SVM, for Intrusion Detection Systems (IDS). The study highlighted that integrating these classifiers with optimization algorithms, such as GA, PSO, JAYA, and TLBO, significantly improves intrusion detection performance, while emphasizing the need for advanced optimization techniques to detect emerging cyberattacks more effectively [35]. The challenges of IDS are illustrated in Figure 5, encompassing issues such as false alarm rates, low detection rates, imbalanced datasets, and response times, among others, which researchers confront

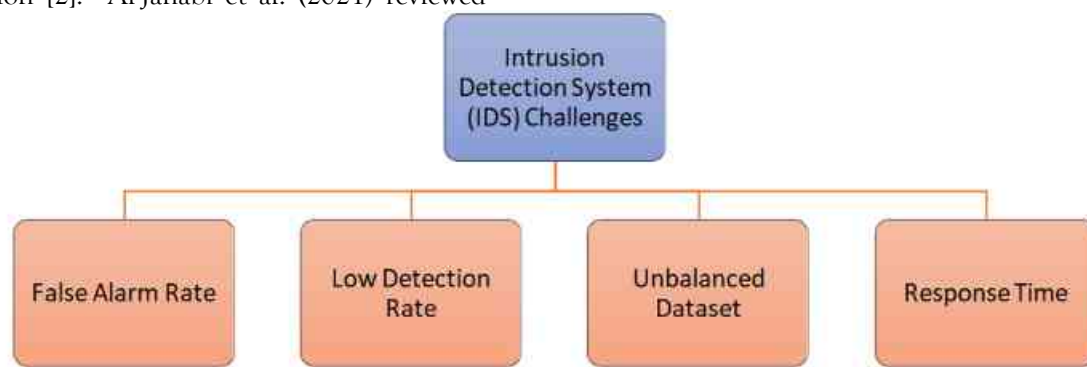


Figure 5: Intrusion detection system (IDS) challenges source: [35]

3 Proposed Hybrid Methodology

The proposed hybrid methodology for the Intrusion Anomaly Detection System (IADS) consists of a sequence of data processing and machine learning steps. Initially, the dataset was obtained from Kaggle and subjected to feature extraction using CountVectorizer and TF-IDF Transformer to convert textual data into numerical representations suitable for machine learning. The extracted features were then pre-processed through MinMax normalization to ensure a consistent feature scale and improve model performance. Subsequently, Fuzzy C-Means (FCM) clustering was applied to group similar data instances before splitting the dataset into 75% training and 25% testing sets. Finally, three supervised machine learning algorithms, namely Support Vector Machine (SVM), Multi-Layer Perceptron (MLP), and Stochastic Gradient Descent (SGD), were trained and evaluated to develop the proposed intrusion anomaly detection model. The overall workflow of the proposed framework is illustrated in Figure 6.

The flowchart shows the steps involved in data preparation, starting with retrieving data from the database. Next, the data is pre-processed using techniques such as normalization and standardization. Cleaning methods are applied to the data before moving on to clustering and classification. The processed data undergoes a division, allocating 75% for training purposes and reserving the remaining 25% for testing, ensuring the validation of the model. The intended technique is categorised into two categories: 1) Classification of data and 2) Pre-processing. This study proposes a hybrid Intrusion Detection System (IDS) that integrates the unsupervised Fuzzy C-Means (FCM) clustering algorithm with supervised machine learning algorithms, including SVML, MLP, and SGD, to improve intrusion detection performance. The proposed framework enhances classification accuracy, precision, recall, and F1-score by combining clustering and classification techniques, demonstrating its effectiveness in developing

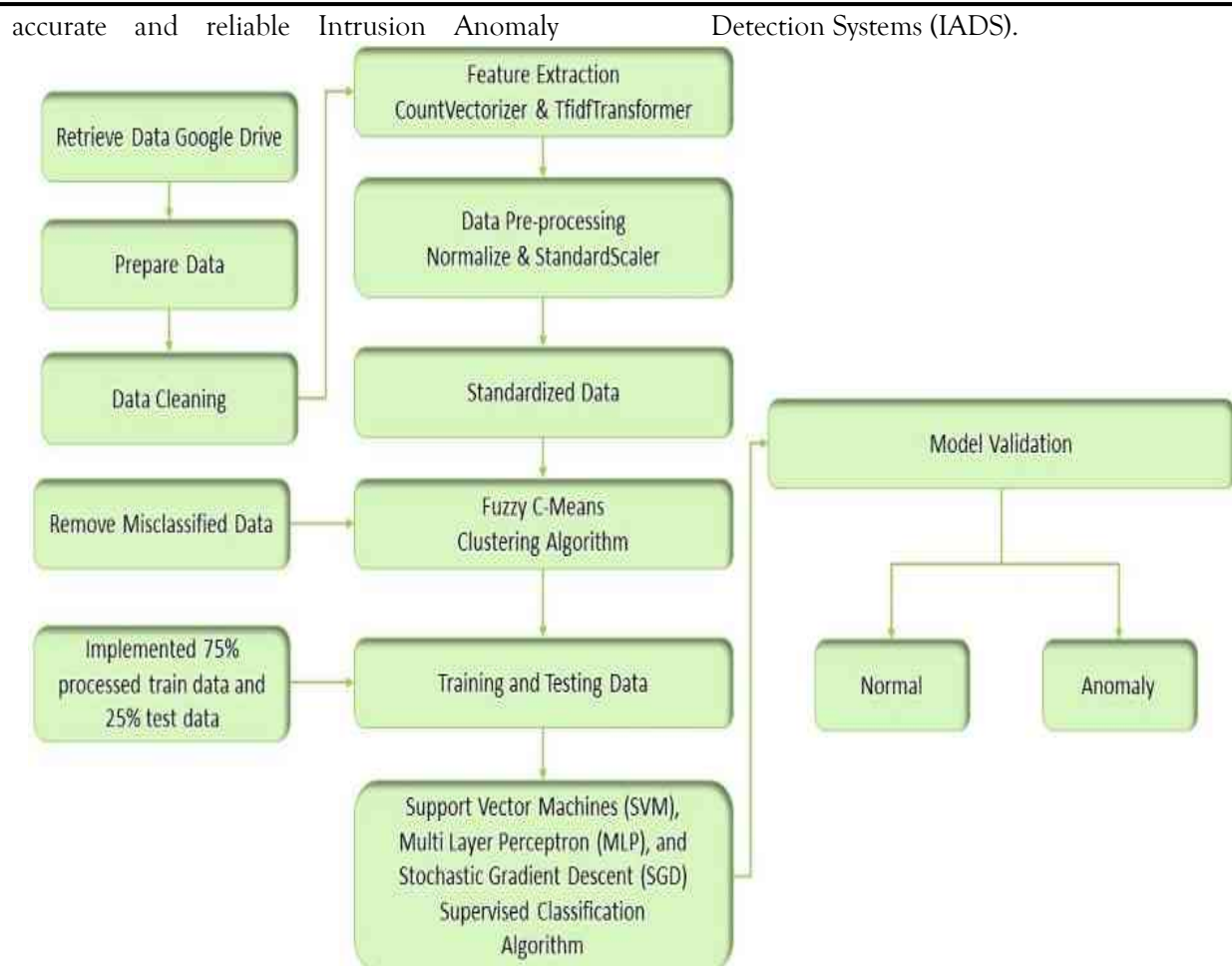


Figure 6: Proposed Hybrid Algorithms Steps in IADS Prediction source: [authors]

4 Data Collection and Analysis

Preprocessing is done at the first stage. Data cleaning and clustering are done during pre-processing to produce meaningful data. To do this, we study two widely used techniques, such as the Fuzzy C-Means clustering mentioned below, before continuing the data processing in the classification stage. Performance analysis is conducted after preprocessing. Python is a language for high-level scripting that is mainly utilised for programming and ML methods, as well as website building and database management. The Anaconda Navigator -> Jupyter Notebook GUI framework was used. We used Python to link datasets and perform Fuzzy C-Means, SVML, MPL and SGD algorithms. The dataset is for IADS and prediction based on anomaly and not-anomaly. This dataset has forty-two columns and 25192 rows. We executed distinct programmes, the initial one of which made use of FCM and

SVML techniques, the second of which used the Multi-layer Perceptron (MPL) and Fuzzy C-Means and Stochastic Gradient Descent (SGD) methods, and they were all based on an identical dataset. All of these programmes are active on the PC. The PC is set up as per the following computer's configuration i.e. Intel (R) Core (TM) 2nd Generation i5-2520M CPU @ 2.50 GHz; four Gigabytes of Random Access Memory; Sixty-Four Bit OS is used; he use of Windows 10 is being done and Five hundred Gigabytes of Hard disk. The dataset used in this study was obtained from Kaggle, a widely used platform that provides diverse machine learning and Intrusion Anomaly Detection System (IADS) datasets for model development and evaluation [32]. According to Mahesh (2019), the dataset consists of network traffic logs collected from an intrusion detection environment, containing records of both normal and malicious activities [62]. Table 1

summarizes the dataset attributes used in this study.

Table 1: Characteristics and Properties of dataset for IADS

Characteristics	Properties
Duration	Duration of the connection's time length
Protocol Type	The protocol utilized for establishing the connection
Service	Utilization of network service in the intended location
Flag	The status of the connection can be either Normal or Error.
Src Bytes	The amount of data bytes transferred from the source to the destination in a single connection.
Dst Bytes	The amount of data bytes transferred from the destination to the source in a single connection.
Land	If the source and destination IP addresses and port numbers are the same, this variable will have a value of 1, otherwise 0.
Wrong Fragment	The total number of incorrect fragments in this connection.
Urgent	The number of urgent packets in this connection, which are packets with the urgent bit activated.
Hot	The number of "hot" indicators in the content, such as entering a system directory, creating programs, and executing programs.
Num Failed Logins	The count of failed login attempts.
Logged In	Login status: 1 if successfully logged in, 0 otherwise.
Num Compromised	The number of compromised conditions.
Root Shell	1 if a root shell is obtained, 0 otherwise.
Su Attempted	1 if the "su root" command is attempted or used, 0 otherwise.
Num Root	The number of root accesses or operations performed as a root in the connection.
Num File Creations	The number of file creation operations in the connection.
Num Shells	The number of shell prompts.
Num Access Files Num	The number of operations on access control files.
Out Bound Cmds	The number of outbound commands in an FTP session.
Is Host Login	1 if the login belongs to the hot list (root or admin), 0 otherwise.
Is Guest Login	1 if the login is a guest login, 0 otherwise.
Count	20. The number of connections to the same destination host as the current connection in the past two seconds.
Srv Count	The number of connections to the same service (port number) as the current connection in the past two seconds.
Error Rate	The percentage of connections that have activated flag (4) s0, s1, s2, or s3, among the connections aggregated in count (23).
SrvError Rate	The percentage of connections that have activated flag (4) s0, s1, s2, or s3, among the connections aggregated in srv_count (24).
Error Rate	The percentage of connections that have activated flag (4) REJ, among the connections aggregated in count (23).
SrvError Rate	The percentage of connections that have activated flag (4) REJ, among the connections aggregated in srv_count (24).
Same Srv Rate	The percentage of connections that were to the same service, among the connections aggregated in count (23).
Diff Srv Rate	The percentage of connections that were to different services, among the connections aggregated in count (23).
Srv Diff Host Rate	The percentage of connections that were to different destination machines, among the connections aggregated in srv_count (24).

Dst Host Count	The number of connections having the same destination host IP address.
Dst Host Srv Count	The number of connections having the same port number.
Dst Host Same Srv Rate	The percentage of connections that were to the same service, among the connections aggregated in dst_host_count (32).
Dst Host Diff Srv Rate	The percentage of connections that were to different services, among the connections aggregated in dst_host_count (32).
Dst Host Same Src Port Rate	The percentage of connections that were to the same source port, among the connections aggregated in dst_host_srv_count (33).
Dst Host Srv Diff Host Rate	The percentage of connections that were to different destination machines, among the connections aggregated in dst_host_srv_count (33).
Dst Host Serror Rate	The percentage of connections that have activated flag (4) s0, s1, s2, or s3, among the connections aggregated in dst_host_count (32).
Dst Host SrvSerror Rate	The percentage of connections that have activated flag (4) s0, s1, s2, or s3, among the connections aggregated in dst_host_srv_count (33).
Dst Host Rerror Rate	The percentage of connections that have activated flag (4) REJ, among the connections aggregated in dst_host_count (32).
Dst Host SrvRerror Rate	The percentage of connections that have activated flag (4) REJ, among the connections aggregated in dst_host_srv_count (33).
Class	There are two categories, Normal and Anomaly

The data were extracted, pre-processed, normalized, and standardized before analysis, and the resulting cleaned dataset is presented in Table 2. Figure 7 illustrates the data distribution before applying the Fuzzy C-Means (FCM) clustering algorithm.

Table 2: Pre-processed Dataset for Intrusion Anomaly Detection System (IADS)

42-Dimension
array ([[0.51039486, 0.12881077, 0.69730558, ..., 0.44025789, 0.16069479, 0.30717637], [0.28271386, 0.34506365, 0.70704749, ..., 0.21096795, 0.59344125, 0.27483909], [0.00308106, 0.19737148, 0.41659779, ..., 0.78916116, 0.42972058, 0.19074031], ..., [0.51224572, 0.62340683, 0.0649827, ..., 0.06133845, 0.5267819, 0.62225094], [0.49793266, 0.37651359, 0.09456641, ..., 0.73468306, 0.38691353, 0.09189262], [0.18193005, 0.65208151, 0.25372843, ..., 0.03408319, 0.43460346, 0.65381383]])

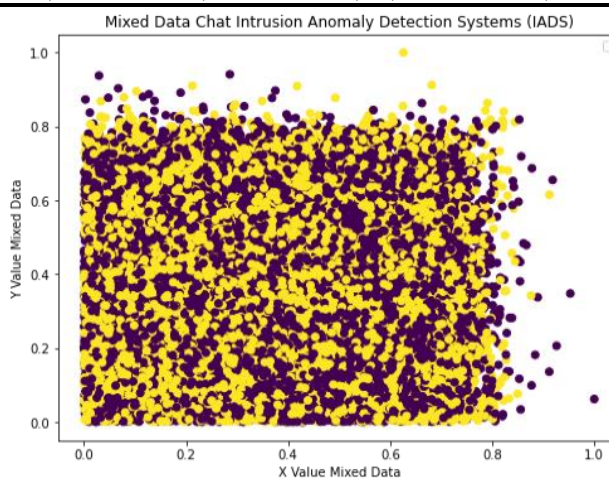


Figure 7: Mixed Data Chat Intrusion Anomaly Detection System (IADS)

4.1 Fuzzy C-Means Clustering (FCM) Algorithm

Clustering is one of the most widely used unsupervised learning techniques for grouping similar data into meaningful clusters across various applications. Fuzzy C-Means (FCM) is a soft clustering algorithm that assigns each data point a membership value indicating its degree of belonging to multiple clusters. The algorithm generates a membership matrix that represents the strength of association between data samples and clusters, enabling effective analysis of complex datasets [63]–[69]. The approach uses a clustering technique based on centroid values, the FCM algorithm, and a 42-dimensional dataset. The participation of each data point with respect to each centroid is determined based on the distance between them. We used the below mentioned formula.

$$\mu_{ij} = 1 / \sum_{k=1}^c (d_{ij} / d_{ik})^{(2/m-1)}$$

$$v_j = \left(\sum_{i=1}^n (\mu_{ij})^m x_i \right) / \left(\sum_{i=1}^n (\mu_{ij})^m \right), \forall j = 1$$

In the provided notation;

- 'n' represents the number of data points.
- 'vj' identifies the jth cluster center.
- The fuzziness parameter 'm' is expressed as $m \in [1, \infty]$.
- 'c' denotes the number of cluster centers.
- ' μ_{ij} ' signifies the membership relationship between the ith data point and the jth cluster center.
- 'dij' represents the Euclidean distance between the ith data point and the jth cluster center.

The fuzzy c-means strategy's main goal is to lower the Euclidean distance shown below:

$$J(U, V) = \sum_{i=1}^n \sum_{j=1}^c (\mu_{ij})^m \|x_i - v_j\|^2$$

The calculation of the distance between the ith data point and jth cluster centre is denoted by ' $\|x_i - v_j\|$ ', which is known as the Euclidean distance. Various mathematical functions, such as Euclidean, Manhattan, and Hamming distances, are utilised in algorithms for classification purposes. Several methods based on these functions include.

$$\begin{aligned} \text{Euclidean} & \quad \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \\ \text{Manhattan} & \quad \sum_{i=1}^k |x_i - y_i| \\ \text{Linkowski} & \quad \left(\sum_{i=1}^k (|x_i - y_i|)^q \right)^{1/q} \end{aligned}$$

Fuzzy C-Means Clustering (FCM) methodology steps:

Let $X = \{x_1, x_2, x_3, \dots, x_n\}$ represent a set of data points and $V = \{v_1, v_2, v_3, \dots, v_c\}$ represent a set of centers.

1. Randomly select 'c' cluster centroids
2. O uses the following formula to calculate the fuzzy participation ' μ_{ij} ':

$$\mu_{ij} = 1 / \sum_{k=1}^c (d_{ij} / d_{ik})^{(2/m-1)}$$

3. Compute the fuzzy centroids 'vj' as follows:

$$v_j = \left(\sum_{i=1}^n (\mu_{ij})^m x_i \right) / \left(\sum_{i=1}^n (\mu_{ij})^m \right), \forall j =$$

4. Continue with step two and three till the least 'J' value or ' $\|U(k+1) - U(k)\|$ ' is obtained.

- 4.1 The process of recurrence is denoted with the letter 'k'.
- 4.2 β is the completion criterion among $[0, 1]$.
- 4.3 ' $U = (ij) n * c$ ' is a clearly stated fuzzy relationship matrix.
- 4.4 The impartiality function is symbolised with the letter 'J'.

The dataset for IoT and Network devices and clusters includes the Intrusion Anomaly Detection System (IADS), which is a 43-dimensional dataset consisting of features related to intrusion anomaly detection and a target property cluster number (refer to Table 3&4). The FCM Clustering centroid value is briefly defined below, while Figure 8 illustrates the structured data obtained after the conversion of the unstructured dataset, displaying two clusters (red and green) with their respective centroid values marked by a yellow star (*). The sum of squared error line chart determined by FCM is presented in Figure 9.

The Fuzzy C-Means Clustering Centroid Values are mentioned in table 3.

Table 3: Fuzzy C-Means Clustering Centroid Values

array ([[0.41111278, 0.4175823, 0.40520431, 0.40963772, 0.39659291, 0.41348441, 0.40628412, 0.40544409, 0.39563059, 0.37751328, 0.41382182, 0.4100243, 0.41324586, 0.41414008, 0.40064532, 0.3909693, 0.3908696, 0.39887343, 0.39959413, 0.39575394, 0.40374983, 0.41479982, 0.41584309, 0.42187425, 0.41248654, 0.38908152, 0.40787647, 0.40884044, 0.39829041, 0.38619826, 0.39859172, 0.41156369, 0.37779919, 0.41189327, 0.39313236, 0.39755094, 0.40555183, 0.39280392, 0.41006084, 0.41166285, 0.3986973, 0.4219502], [0.41111278, 0.4175823, 0.40520431, 0.40963772, 0.39659291, 0.41348441, 0.40628412, 0.40544409, 0.39563059, 0.37751328, 0.41382182, 0.4100243, 0.41324586, 0.41414008, 0.40064532, 0.39096931, 0.3908696, 0.39887343, 0.39959413, 0.39575394, 0.40374983, 0.41479982, 0.4158431, 0.42187425, 0.41248654, 0.38908152, 0.40787647, 0.40884044, 0.39829041, 0.38619826, 0.39859172, 0.41156369, 0.3777992, 0.41189327, 0.39313236, 0.39755094, 0.40555183, 0.39280392, 0.41006084, 0.41166285, 0.3986973, 0.4219502]])

Table 4: Fuzzy C-Means Two Clusters Pre-processed Intrusion Anomaly Detection System (IADS) Dataset

43-Dimension

array ([[0.51039486, 0.12881077, 0.69730558, ..., 0.44025789, 0.16069479, 0.30717637, 1], [0.28271386, 0.34506365, 0.70704749, ..., 0.21096795, 0.59344125, 0.27483909, 1], [0.00308106, 0.19737148, 0.41659779, ..., 0.78916116, 0.42972058, 0.19074031, 0], ..., [0.51224572, 0.62340683, 0.0649827, ..., 0.06133845, 0.5267819, 0.62225094, 1], [0.49793266, 0.37651359, 0.09456641, ..., 0.73468306, 0.38691353, 0.09189262, 1], [0.18193005, 0.65208151, 0.25372843, ..., 0.03408319, 0.43460346, 0.65381383, 1]])

Intrusion Anomaly Detection Systems (IADS) for Prediction Fuzzy C-Means Clusters=2

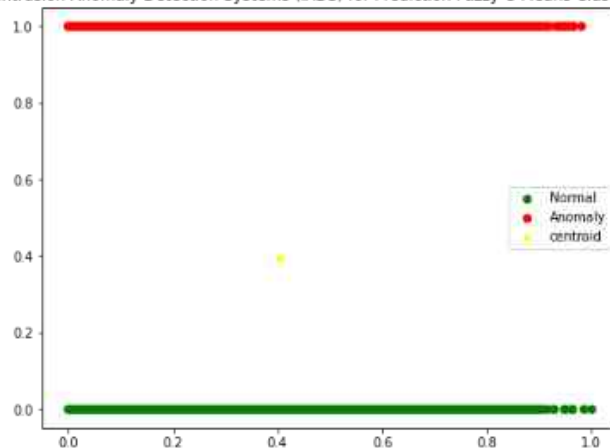


Figure 8: Fuzzy C-Means Two Clusters

Precision, recall, and f-measure are used to compare the performance of CFD employing both clustering algorithms.

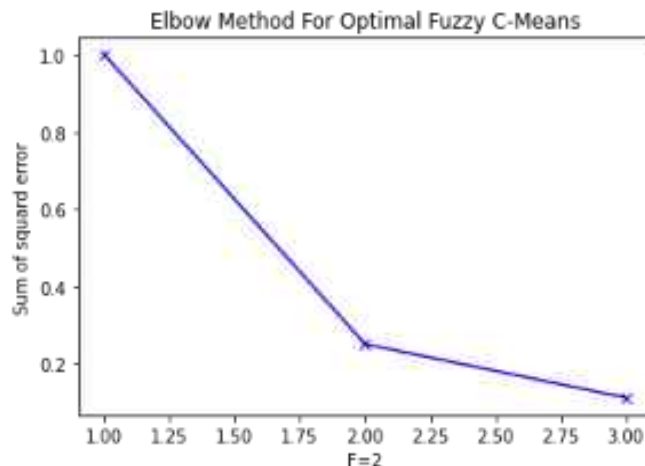


Figure 9: Fuzzy C-Means Sum of Squared Error Line Chart

The amount of divergence from basic systems is used to generate a worry value, and the action is then classified as valid, suspicious, or unlawful.

Classification is a supervised machine learning technique that uses labeled training data to learn patterns and classify new instances into predefined categories. It establishes decision boundaries based on input features to enable accurate prediction and classification. In this study, SVML, MLP, and SGD classifiers were evaluated to identify the best-performing model for intrusion anomaly detection.

4.2 Support Vector Machine Linear (SVML) Algorithm

An approach to supervised learning is the Support Vector Machine Linear (SVML). For regression and classification, utilise the SVML approach. This algorithm is employed in the supervised approach, and it functions as a partition. For example, the X and Y components of a graph are both mixed classes. Now that we have separated 2 groups, specialists would make a distinction between them. Numerous functions, are employed with SVML. The X1, X2 components, as well as the different classes parabolic and circular, are separated by a decision boundary support vector, and the two closest data points to the distance measure are referred to as the support vector. Some mathematical classification and regression functions utilised in the (SVM) technique are briefly described. Linear & Non-linear SVM, and Kernel function are the three types of SVM. These are the functions listed below:

Linear SVM

$$x_i \cdot x_j$$

Non-linear SVM

$$\Phi(x_i) \cdot \Phi(x_j)$$

Kernel function

$$k(x_i \cdot x_j)$$

The key steps of Support Vector Machine Linear (SVML) algorithm works are as follows:

1. Describe the appropriate separating hyper-planes to maximize the margins.
2. Enhancing the over-definition using non-linearly different issues contains a phrase of consequence for misinterpretation.
3. Operation data to high-dimensional storage, in which it is better to understand surface areas with immediate selection, and restructure the certifiable data as per this disk space.
4. Finish.

The Confusion matrix is a widely used metric in classification problem-solving. It can be utilized for both binary and multiclass classifications. The evaluation of the IADS is typically determined by analysing the Confusion matrix.

- The True Positive Rate (TPR) evaluates the correlation between the identified & total attacks. When all intrusions are accurately detected, the TPR is 1, which is highly uncommon for an Intrusion Detection System (IDS). Additionally, the TPR is also referred to as the DR and is defined as:

$$\text{True Positive Rate (TPR)} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad 1$$

- FPR determines how many normal cases are incorrectly identified as attacks in comparison to the total number of normal cases. The calculation of FPR is as follows:

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{(\text{FP} + \text{TN})} \quad 2$$

- FNR indicates that the IDS failed to identify the intrusion and instead labeled it as normal. The calculation for FNR is as follows:

$$\text{False Negative Rate (FNR)} = \frac{\text{FN}}{(\text{FN} + \text{TP})} \quad 3$$

The Support Vector Machine Linear (SVML) model prediction true and predicted label shows the result of that SVML model confusion matrix Figure 10.

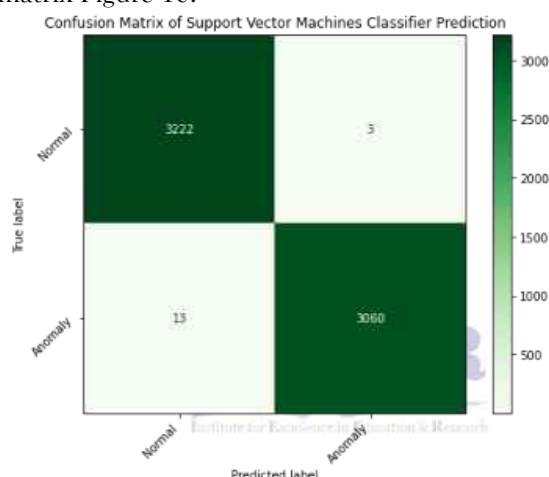


Figure 10: Confusion Matrix Support Vector Machine Linear (SVML) Algorithm

The confusion matrix has an explanation [{"A", "B"}][{"C", "D"}] as of this publication, while

- "A" denotes the number of accurately expected instances that are negative.
- "B" represents number of incorrect predicted positive instance,
- "C" is the number of incorrect estimations that a particular case is negative
- "D" is the total amount of right guesses in a positive occurrence.

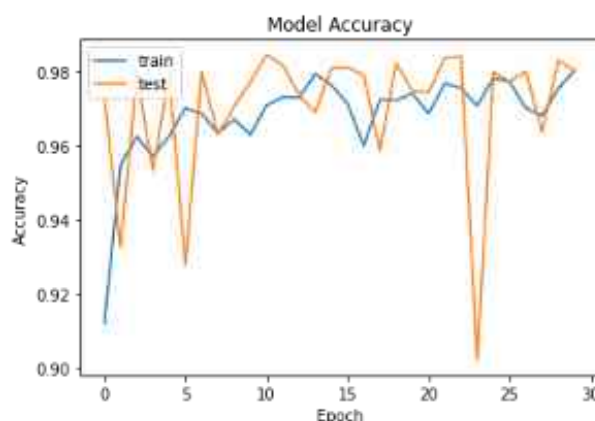


Figure 11: Model Accuracy Support Vector Machine Linear (SVML) Algorithm

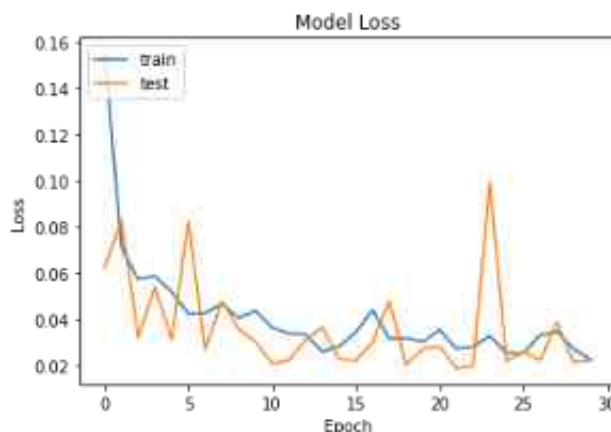


Figure 12: Model Loss Support Vector Machine Linear (SVML) Algorithm

In our study, we evaluated our model using the Support Vector Machine Linear (SVML) Algorithm and considered its accurateness and failure concepts. This approach helps to get better the accuracy of the approximate forecasting method and ensures that the Intrusion Anomaly Detection System (IADS) prediction patterns are frequently met. Figure 11-12 illustrates the importance of model accuracy and loss. Specifically, the Support Vector Machine Linear (SVML) model achieved a train data accuracy of 0.999 and a test data accuracy of 0.998, as shown in Figure 11. Meanwhile, the Logistic Regression Algorithm yielded a train data loss of 0.023 and a test data loss of 0.022, as shown in Figure 12.

4.3 Multi-Layer Perceptrons (MLP) Algorithm

Multi-Layer Perceptrons (MLP) is a supervised neural network algorithm consisting of an input layer, one or more hidden layers, and an output layer. The hidden layers enable MLP to learn complex, non-linear relationships and approximate continuous functions, making it effective for classification and prediction tasks [70], [71]. In this study, the MLP classifier was employed to develop and evaluate the proposed Intrusion Anomaly Detection System (IADS). These represent a few of these methods:

Sigmoid	$S(z) = \frac{1}{(1 + e^{-z})}$
Linear Regression	$y = e^{(b_0 + b_1 \cdot x)} / (1 + e^{(b_0 + b_1 \cdot x)})$
Cost Linear Regression	$(Cost(h\theta(x), y)) = -\log(h\theta(x)), \text{ if } y = 1 \text{ and}$ $(Cost(h\theta(x), y)) = -\log(1 - h\theta(x)), \text{ if } y = 0$
Nonlinear Regression	$Y = f(X, \beta) + \epsilon$

The MLP algorithm main steps exertion as follows:

1. The MLP learns facts by computing partial derivatives of input recordings and variables that emerge among the hidden and input layers. In the buried layer, the sample distribution yields a value. Unlike an activation function, however, we do not increase its value.
2. In each estimated layer, MLP incorporates activation functions. There are several processing methods to consider, including the sigmoid, rectified linear units, and the tanh processes. To transmit the result of measurement to the visible layer, any of these methods is employed.
3. Extraction of the partial derivatives with the needed values should be done once the activation function has produced the desired result at the hidden layer, and these derivatives should then be sent to another layer inside the Multi-Layer Perceptron.
4. Repeat steps 2 and 3 above until you obtain the result.
5. The approximations will be utilized in the final stage to obtain outcomes either for a feed-forward approach that aligns with the activation techniques picked solely for the MLP (in the context of training data), or a choice derived from the findings (in the case of testing data).
6. Finish.

The MLP prediction involves labeling the past data values and forecasting the value of the new data. In the process of training, efforts are made to align the predictions with the labels. The results of this process are illustrated in Figure 13, which displays the confusion matrix of the MLP.

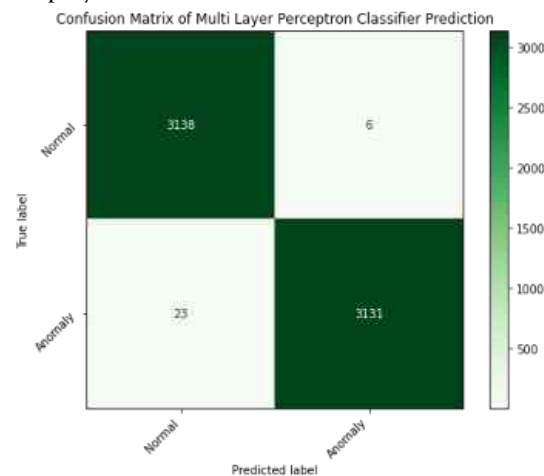


Figure 13: Confusion MLP Algorithm

Presuming that the Multi-Layer Perceptron (MLP) model is suitable for this context, the utilization of a confusion matrix facilitated the computation of the predicted labels for our detection and prediction.

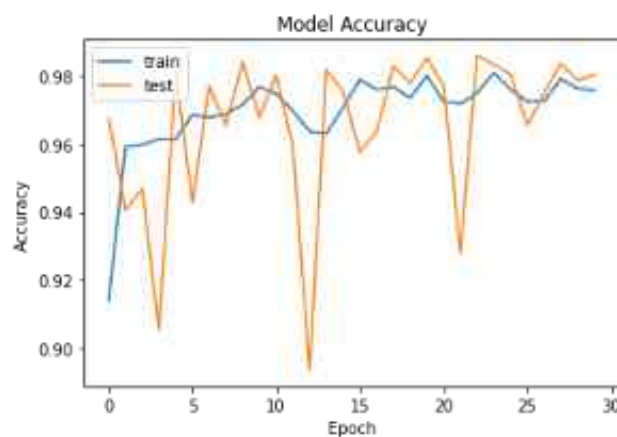


Figure 14: Model Accuracy MLP Algorithm

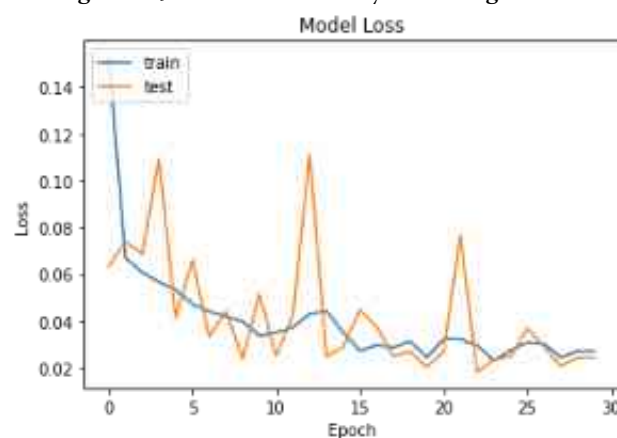


Figure 15: Model Loss Multi-Layer Perceptron's (MLP) Algorithm

In the Multi-Layer Perceptron's (MLPs) Algorithm Figure 14, the train data was 0.998 and the data tested was 0.999 in the Accuracy Model. In the Multi-Layer Perceptron's (MLPs) Algorithm Figure 15, the train data was 0.023 and the data tested was 0.022 in the above figures.

4.4 Stochastic Gradient Descent (SGD) Algorithm

Stochastic Gradient Descent (SGD) is an optimization algorithm that iteratively updates model parameters using individual training samples to minimize an objective function. Its incremental learning approach makes it computationally efficient for large-scale datasets and suitable for training linear models such as Support Vector Machines (SVM) and Logistic Regression. In this study, SGD was employed as one of the classification algorithms for intrusion anomaly detection [72]–[76]. The mathematical expression of the algorithm is given below:

Gradient Descent (SGD) theorem

$$w := w - \eta \nabla Q_i(w)$$

The fundamental phases in how the Stochastic Gradient Descent (SGD) approach performs are as follows:

1. Determine the gradient of the main method about each constraint/ characteristic. In many other respects, estimate the gradients of the method.
2. Choose a random starting value for the parameters. (To explain, inside the parabolic scenario, distinguish "y" about "x". If we had any other characteristics, such as x1, x2, and so on, we would compute the approximate value of "y" about each one of them).
3. Recalculate the gradients method by changing the model parameters.
4. Determine the iterative methods for every characteristic using the formula: sampling rates = gradients * training rates.
5. Establish the following new variables: new parameters = old parameters -step length.
6. Steps were repeated 3 to 5 there until the gradient was almost zero.
7. Finish.

On our above dataset, we perform the SGD method and tag the prior values of the data. When we used the Stochastic Gradient Descent (SGD) approach to attempt to ensure our predictions suit the categories during preparations Figure 16 depicts the outcome of that matrix. The best thing about this confusion matrix is that it doesn't mislead the outcome.

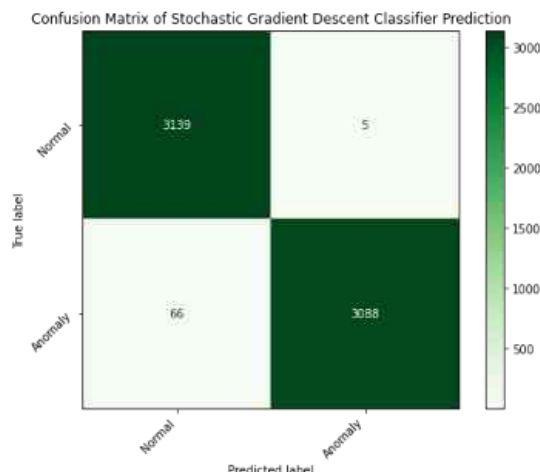


Figure 16 Confusion SGD Algorithm

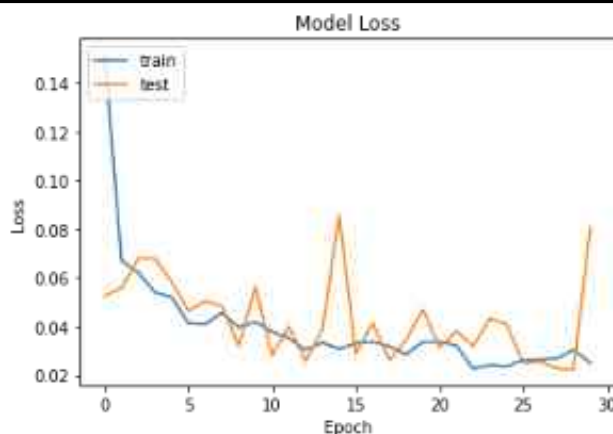


Figure17 Model Accuracy SGD Algorithm

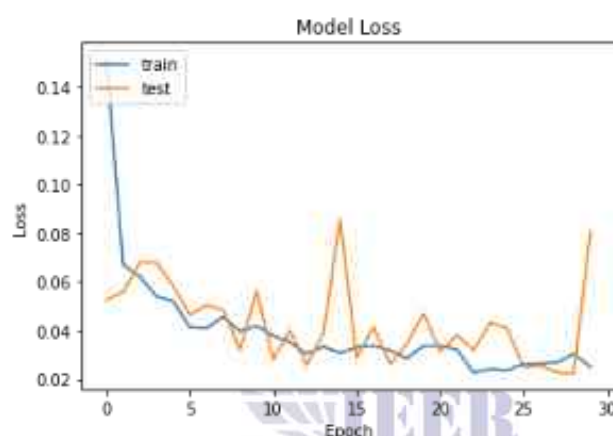


Figure 18 Model Loss SGD Algorithm

In the SGD Algorithm Figure 17, the train data was 0.987 and the data in the trial was 0.985 in the Model Accuracy. In the SGD Algorithm Figure 18, the train data was 0.022 and the data tested was 0.020 in the Model Loss.

- Classification rate (CR) or Accuracy refers to the overall accuracy of the (IDS) in classifying normal activities and intrusion attacks. It is determined by calculating the correct classification rate.

$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})}$	4
---	---

- Precision (P) is calculated as the ratio of the number of true positive (TP) instances to the total number of true positive (TP) and false positive (FP) instances.

$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \times 100 \%$	5
--	---

- Recall (R) denotes the proportion of all appropriate outcomes accurately categorized, which is calculated by dividing the number of true positives (TP) by the total sum of true positives and false negatives (FN) instances.

$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \times 100 \%$	6
---	---

- F-measure, also known as the F1-score, calculates the average of both precision and recall. It is commonly used when there is a need for a single accuracy metric as an evaluation tool.

$\text{F - measure} = \frac{2 * \text{Recall} * \text{Precision}}{(\text{Recall} + \text{Precision})}$	7
--	---

- Sensitivity refers to the likelihood of obtaining a positive test result when an individual is positive.

$\text{Sensitivity} = \frac{TP}{(TP + FN)}$	8
---	---

- Specificity, also known as the true negative rate, refers to the likelihood of obtaining a negative test result when the person is negative.

$\text{Specificity} = TN / (TN + FP)$	9
---------------------------------------	---

5 Experimental Results

This research has proposed an approach for predicting and detecting cyber-attack Intrusion Anomaly Detection Systems (IADS) that makes use of both ML and DL techniques and data from prior cyber-attack incidents. Our study estimates the features of those who've been cyber-attacks assaulted as well as the kinds of attacks to which they'll be applied. This research employs a novel approach that involves combining several algorithms. Fuzzy C-Means, Support Vector Machines (SVM), Multi-layer perception (MLP), and Stochastic Gradient Descent (SGD). In this section, we merge the algorithms to get the most accurate relation for Intrusion Anomaly Detection System (IADS) identification. The algorithms' combinations are as follows: Fuzzy C-Means, Support Vector Machines (SVM), and Fuzzy C-Means, Stochastic Gradient Descent (SGD). We obtained the greatest 99.74% accuracy by combining our methods with Fuzzy C-Means and Support Vector Machines (SVM). After that, we brought the second-ranked combination 99.53% accuracy of FCM, Multi-layer perception (MLP). Subsequently, we carried the third-ranked combination 98.87% accuracy of Fuzzy C-Means and stochastic Gradient Descent (SGD). Tables 5 and 6 show the acquired precision and other performance factors.

Table 5: Combination of Algorithms Model Accuracy for IADS

Combination Algorithm	Accuracy of Algorithms
Fuzzy C-Means, Stochastic Gradient Descent (SGD) Proposed Hybrid Technique	98.87 %
Fuzzy C-Means, Multi-Layer Perceptron (MLP) Proposed Hybrid Technique	99.53 %
Fuzzy C-Means, Support Vector Machines (SVM) Proposed Hybrid Technique	99.74 %

Table 6: Combination of Algorithms Parameter Score for IADS

S/No.	Parameter Score	Fuzzy C-Means, Stochastic Gradient Descent (SGD)	Fuzzy C-Means, Multi-Layer Perceptron (MLP)	Fuzzy C-Means, Support Vector Machine (SVM)
1	Precision	0.99750101	0.99540558	0.98889531
2	Recall	0.99741968	0.99539963	0.98874193
3	F1-Score	0.99745782	0.99539535	0.98872584
4	Sensitivity	0.99906976	0.99809160	0.99840966
5	Specificity	0.99576960	0.99270767	0.97907419

6 Comparative Analysis

The effectiveness of the classifier was evaluated in comparison to several other supervised ML techniques to determine the optimal one. The different ML algorithms were utilized and significant results were achieved through it. The study already available regarding the cyber-attack and IADS using ML algorithms has given limited knowledge. This study proposed, A. Khraisat et al. [32] to improve the classification model performance using various ML techniques SVM, MLP, NB, K-NN, RF, CART, C5 Decision Tree and One Class SVM. The highest accuracy scores 83.24 %, C5 Decision Tree Classifier and One Class Support Vector Machine. Another study suggested A. Khraisat et al. [77] using C4.5 ML classification method and enhanced accuracy, precision, recall, f1-score. The accuracy rate 81.0 %. In this research, these Machine

Learning supervised methods include Fuzzy C-Means clustering, Support Vector Machines (SVM), Multi-layer perception (MLP), and Stochastic Gradient Descent (SGD). These techniques have suggested that the fusion of such algorithms will result in a robust model for the prediction of IADS. A combination of Fuzzy C-Means and Support Vector Machines (SVM) achieved the highest accuracy of 99.74 %. Furthermore, the combination of Fuzzy C-Means and Multi-layer perception (MLP) obtained the second-highest accuracy of 99.74 %. Also, A combination of Fuzzy C-Means and Stochastic Gradient Descent (SGD) gained the third-highest accuracy of 98.87 %. Tables 7 show the ML model performance factors.

Table 7: Combination of Algorithms Model Accuracy for IADS

Combination Algorithm	Accuracy of Algorithms
A. Khraisat et al. [32]	Support Vector Machines (SVM) 69.52 %
	Multi-layer perception (MLP) 77.41 %
	Naïve Bayes (NB) 76.56 %
	K-Nearest Neighbors (KNN) 79.4 %
	Random Forest (RF) 80.67 %
	Classification And Regression Tree (CART) 80.3 %
	C5 Decision Tree Classifier and One Class Support Vector Machine 83.24 %
A. Khraisat et al. [77]	C4.5 81.0 %
Fuzzy C-Means, Stochastic Gradient Descent (SGD) Proposed Hybrid Technique	98.87 %
Fuzzy C-Means, Multi-Layer Perceptron (MLP) Proposed Hybrid Technique	99.53 %
Fuzzy C-Means, Support Vector Machines (SVM) Proposed Hybrid Technique	99.74 %

From the observed results in Figure 19 and Figure 20, it is clear that the combination of Fuzzy C-Means and SVM, in terms of precision, has achieved its peak. Nonetheless, the combination FCM and multi-layer Perceptron (MPL), was placed second, following the integration of FCM, Stochastic Gradient Descent (SGD) as third-ranked. According to the outcomes, the accuracy graph also depicts the combinations' predictions at their highest.

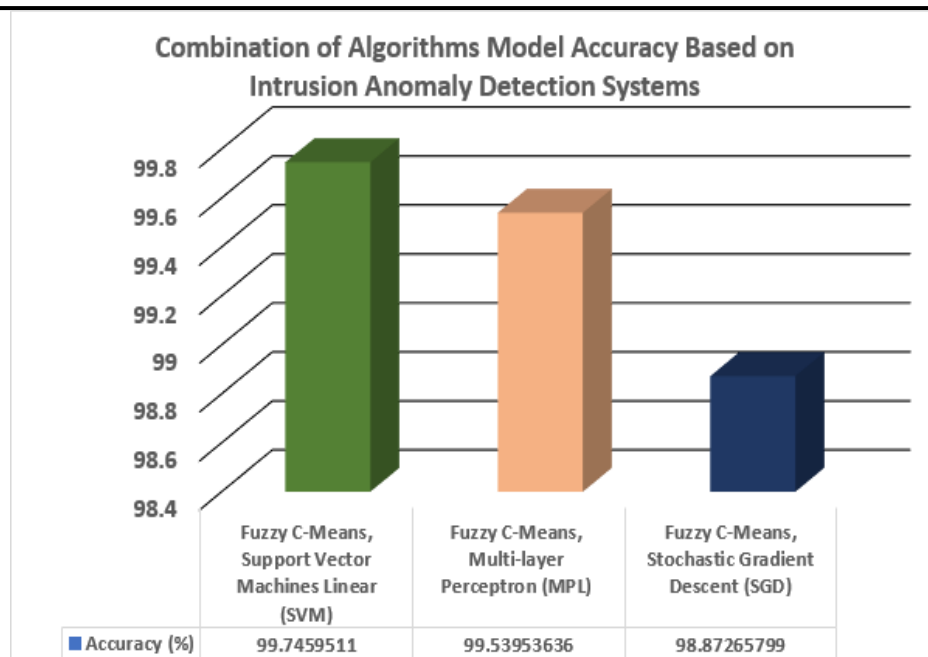


Figure 19: Combinations of Algorithms Model Accuracy IADS of Prediction

We can limit or change the accuracy depending on the situation. For example: Recall, the Parameter Score Precision, Specificity, Sensitivity, and F1-Score are all at their highest levels of accuracy.

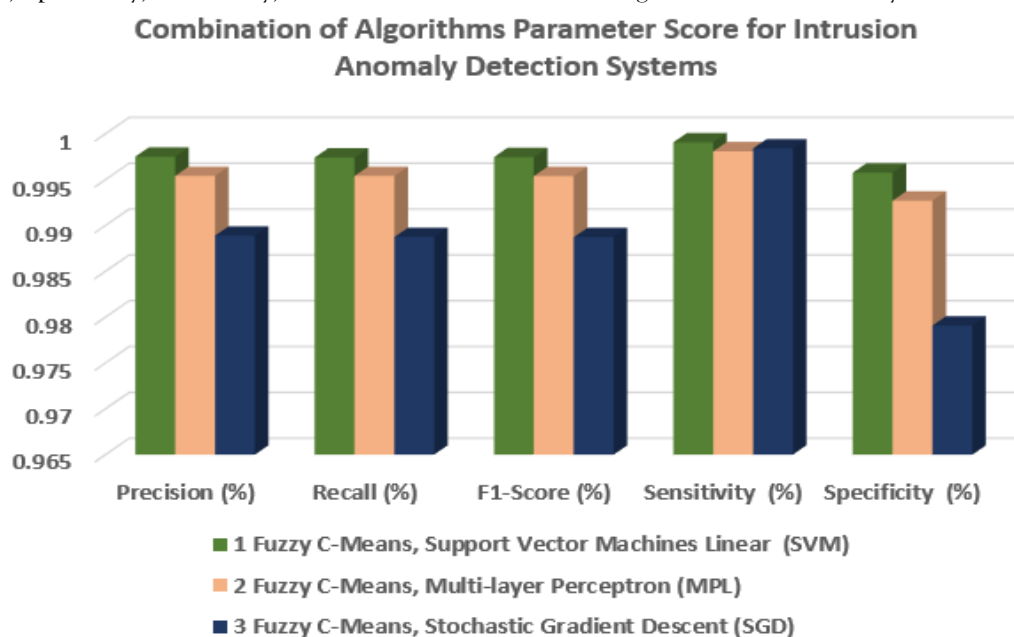


Figure 20: Combinations of Algorithms Parameter Score IADS of Prediction

7 Discussion

The rapid growth of internet technologies and IoT devices has significantly increased cybersecurity challenges, making Intrusion Detection Systems (IDSs) essential for protecting sensitive information and network infrastructures [13], [14], [18]. Although recent advances in Machine Learning (ML) and Deep

Learning (DL) have improved intrusion detection capabilities, existing IDSs still face challenges in detecting unknown attacks while minimizing false alarm rates [27], [28], [29]. Furthermore, the increasing complexity of IoT environments has encouraged the development of intelligent anomaly detection techniques capable of adapting to evolving cyber threats

[34], [15], [19]. Recent studies have demonstrated the effectiveness of ML and AI techniques for intrusion detection and anomaly detection in IoT environments. AI-driven approaches have improved the identification of cyber threats through network traffic analysis [44], [45], while comparative evaluations of NN, RF, NB, DT, GB, and MLP have highlighted their effectiveness for cyberattack prediction [49], [50]. Similarly, several studies have shown that RF consistently provides superior performance for IoT anomaly detection compared with other ML algorithms [51], [52], [53]. These findings support the growing adoption of intelligent ML-based IDS frameworks. The effectiveness of hybrid ML models has also been demonstrated across multiple application domains. Siddiqui et al. (2023) achieved 99.88% accuracy by integrating K-Means with MLP for COVID-19 detection [2], while Shaikh et al. (2023) reported 99.67% accuracy for software defect prediction using K-Means + MLP [1]. Likewise, Hassan et al. (2023) achieved 99.98% accuracy for sentiment classification using K-Means + LR [3], illustrating the advantages of combining clustering and classification techniques. Similar improvements have been reported in fraud detection through ML and DL techniques [12], whereas ML has also shown promising results in software vulnerability assessment and risk prediction [55] and educational prediction systems [8]. Comparative studies on database performance further demonstrate the effectiveness of AI and ML models in evaluating computational efficiency [48], [11]. Consistent with these findings, the proposed FCM-based hybrid framework significantly improved intrusion detection performance compared with conventional standalone ML algorithms. The integration of FCM with SVML achieved the highest classification performance, followed by FCM + MLP and FCM + SGD, demonstrating that clustering before classification improves feature representation and predictive accuracy. These results further support previous recommendations to strengthen IDS accuracy through intelligent hybrid approaches [60]. The performance factor and acquired precision are shown in the table 6 and 7. Regarding the precision support vector machine, and Fuzzy C-

Means, the combination achieved the highest precision, integration of multilayer perceptron and fuzzy c means together was ranked second, and Stochastic Gradient Descent and Fuzzy C-Means together achieved third place and these results are evident from the figure 19, and 20. The previous research results present the accuracy percentage of different machine learning algorithms mentioned in Table 6. Support Vector Machines achieved an accuracy of 69.52%, Multi-layer Perceptron (MLP) attained 77.41%, Naïve Bayes (NB) reached 76.56%, K-Nearest Neighbors (KNN) scored 79.4%, Random Forest (RF) achieved 80.67%, Classification and Regression Tree (CART) obtained 80.3%, and C4.5 demonstrated an accuracy of 81.0%, while C5 Decision Tree Classifier and One-Class Support Vector Machine showed an accuracy rate of 83.24%. Present Research elaborates that the introduction of Fuzzy C-Means in combination with different ML techniques proposed in this research has significantly enhanced the performance as compared to the previous outcomes with individual ML algorithms. Overall, the proposed framework addresses key limitations of existing IDSs by improving classification performance and reducing dependence on individual ML classifiers, thereby contributing to more robust intrusion anomaly detection for modern cybersecurity environments.

8 Conclusion

This study critically evaluated the performance of classification algorithms for an Intrusion Anomaly Detection System (IADS) by integrating Fuzzy C-Means (FCM) with SVML, MLP, and SGD. The proposed hybrid models achieved high classification accuracies of 99.74% with SVML, 99.53% with MLP, and 98.87% with SGD, outperforming the eight comparison models and demonstrating the effectiveness of combining clustering with classification techniques. Despite these promising results, further research is needed to improve the robustness, adaptability, and scalability of the proposed framework for real-world intrusion detection environments. Overall, this study provides an effective hybrid machine learning framework that contributes to

the advancement of accurate and reliable intrusion anomaly detection systems.

9 Limitations

This study has several limitations. First, the proposed framework was evaluated using a single Kaggle dataset, which may not fully represent real-world IoT and network traffic environments. Second, only three classification algorithms (SVML, MLP, and SGD) were investigated, excluding other advanced ML and DL models. Third, the model was validated on a moderate-sized dataset, leaving its scalability in large-scale, real-time environments unverified. Finally, the proposed approach may have limited capability in detecting zero-day and previously unseen cyberattacks.

10 Future Work

Future research should validate the proposed framework using large-scale benchmark datasets, such as UNSW-NB15, CICIDS2017, and BoT-IoT. Integrating FCM with advanced deep

learning models, including CNN, LSTM, and Transformer architectures, may further improve detection performance. Additionally, adaptive learning methods for zero-day attack detection, hybrid feature selection and dimensionality reduction techniques, and deployment in real-time network environments should be explored to enhance scalability and practical applicability.

Acknowledgment

The authors gratefully acknowledge Kaggle for providing the dataset used in this research. We also thank the Higher Education Commission (HEC), Pakistan, for providing access to digital library resources, and express our appreciation to our colleagues, teachers, and the administrative and technical staff of our university for their continuous support and cooperation.

Conflict of Interest

There was no conflict of interest among the authors of the present research paper.

REFERENCES

- S. Y. Shaikh, N. A. Qureshi, M. Z. Khan, M. A. Khan, A. Imroz, and M. A. Kalwar, "Performance Analysis of Classification Algorithms for Software Defects Prediction by Mathematical Modelling & Simulations," *Sindh J. Headways Softw. Eng.*, vol. 2, no. 1, pp. 1-28, 2023, [Online]. Available: <http://sjhse.smiu.edu.pk/sjhse/index.php/SJHSE/article/view/54/11>
- M. Siddiqui *et al.*, "Performance Analysis for the Diagnosis of COVID-19 Prediction by Mathematical Modeling & Simulation," *Int. J. Artif. Intell. Math. Sci.*, vol. 2, no. 1, pp. 1-28, 2023, [Online]. Available: <http://ijaims.smiu.edu.pk/ijaims/index.php/AIMS/article/view/47/18>
- F. Hassan *et al.*, "Performance evolution for sentiment classification using machine learning algorithm," *J. Appl. Res. Technol. Eng.*, vol. 4, no. 2, pp. 97-110, 2023, [Online]. Available: <https://polipapers.upv.es/index.php/JART E/article/view/19306/15859>
- B. P. Ndahi, O. A. Abisoye, H. O. Aliyu, and O. A. Ojerinde, "Development of Pi Sigma Neural Network Model for the Prediction of Software Reliability Using 5 NASA Failure Datasets," 2023.
- G. Secundo, G. Mele, G. Passiante, and A. Ligorio, "How machine learning changes Project Risk Management: a structured literature review and insights for organizational innovation," *Eur. J. Innov. Manag.*, vol. 27, no. 8, pp. 2597-2622, 2024.
- A. Iftikhar, M. Alam, R. Ahmed, S. Musa, and M. M. Su'ud, "Risk prediction by using artificial neural network in global software development," *Comput. Intell. Neurosci.*, vol. 2021, 2021.
- A. Iftikhar, M. Alam, S. Musa, and M. M. Su'ud, "Trust Development in virtual teams to implement global software development (GSD): A structured approach to overcome communication barriers," in *2017 IEEE 3rd International Conference on Engineering Technologies and Social Sciences (ICETSS)*, IEEE, 2017, pp. 1-5.

- M. A. Baig, S. A. Shaikh, K. K. Khatri, M. A. Shaikh, M. Z. Khan, and M. A. Rauf, "Prediction of Students Performance Level Using Integrated Approach of ML Algorithms," *Int. J. Emerg. Technol. Learn.*, vol. 18, no. 1, pp. 216–234, 2023, doi: 10.3991/ijet.v18i01.35339.
- M. Z. KHAN *et al.*, "Comparative case study: an evaluation of performance computation between support vector machine, K-nearest comparative study: Evaluation of performance computation between support vector component analysis," *J. Tianjin Univ. Sci. Technol.*, vol. 55, no. 4, pp. 388–403, 2022.
- M. Z. Khan, F. U. Zaman, M. Adnan, A. Imroz, M. A. Rauf, and Z. Phul, "Comparative case study: An evaluation of performance computation between SQL and NoSQL database," *J. Softw. Eng.*, vol. 1, no. 2, pp. 14–23, 2023.
- F. U. Zaman, M. A. Khuhro, K. Kumar, N. Mirbahar, M. Z. Khan, and A. Kalhor, "Comparative Case Study Difference Between Azure Cloud SQL and Atlas MongoDB NoSQL Database," *Int. J. Emerg. Trends Eng. Res.*, vol. 9, no. 7, 2021, [Online]. Available: doi: 10.30534/ijeter/2021/26972021
- M. Z. Khan *et al.*, "The Performance Analysis of Machine Learning Algorithms for Credit Card Fraud Detection," *Int. J. Online Biomed. Eng.*, vol. 19, no. 03, pp. 82–98, 2023, doi: 10.3991/ijoe.v19i03.35331.
- M. I. Al-Ghamdi, "Effects of knowledge of cyber security on prevention of attacks," *Mater. Today Proc.*, 2021.
- A. Agarwal, P. Sharma, M. Alshehri, A. A. Mohamed, and O. Alfarraj, "Classification model for accuracy and intrusion detection using machine learning approach," *PeerJ Comput. Sci.*, vol. 7, p. e437, 2021.
- M. Komar, V. Dorosh, G. Hladiy, and A. Sachenko, "Deep neural network for detection of cyber attacks," in *2018 IEEE First International Conference on System Analysis & Intelligent Computing (SAIC)*, IEEE, 2018, pp. 1–4.
- N. Ben Amor, S. Benferhat, and Z. Elouedi, "Naive bayes vs decision trees in intrusion detection systems," in *Proceedings of the 2004 ACM symposium on Applied computing*, 2004, pp. 420–424.
- J. Li, Z. Zhao, R. Li, and H. Zhang, "Ai-based two-stage intrusion detection for software defined iot networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2093–2102, 2018.
- B. Mukherjee, L. T. Heberlein, and K. N. Levitt, "Network intrusion detection," *IEEE Netw.*, vol. 8, no. 3, pp. 26–41, 1994.
- O. A. Raheem and E. S. Alomari, "An adaptive intrusion detection system by using decision tree," *J. ALQadisiyah Comput. Sci. Math. Vol.*, vol. 10, no. 2, 2018.
- M. Ali, M. A. Khan, and M. A. Kalwar, "Challenges for Online Banking in Customers' Perspective: A Review," *Int. J. Bus. Educ. Manag. Stud.*, vol. 5, no. 1, pp. 37–57, 2021, [Online]. Available: <https://ijbems.com/doc/IJBEMS-167.pdf>
- M. A. Gul, Z. K. Imran, M. A. Kalwar, and M. A. Khan, "Online Banking's Infrastructural Challenges: Detailed Analysis in Customers' Perspective Across Gender and Availability of Internet," in *Proceedings (Abstract) of the International Conference on Industrial & Mechanical Engineering and Operations Management Dhaka, Bangladesh, December 26-27, 2021.*, 2021, p. 904. [Online]. Available: <https://ieomsociety.org/proceedings/2021dhaka/494.pdf>
- M. A. Gul, Z. K. Imran, M. A. Khan, and M. A. Kalwar, "Analysis of Infrastructural Challenges of E-Banking as Reported by Customers," *Proc. 1st Cent. Am. Caribb. Int. Conf. Ind. Eng. Oper. Manag. (IEOM), Port-au-Prince, Haiti, June 15-16.*, pp. 396–408, 2021, [Online]. Available: <http://ieomsociety.org/proceedings/2021haiti/300.pdf>

- M. Kalhoro *et al.*, "Customer satisfaction in commercial bank of Sindh province a case study of bank Al Falah," in *IEEE International Conference on Innovative Research and Development (ICIRD 2018)*, Asian Institute of Technology Conference Center(AITCC) , Bangkok, Thailand , 11-12, May 2018., 2018. [Online]. Available: <https://icird.etssm.org/wp-content/uploads/2018/05/icird-2018.pdf>
- M. Kalhoro, Q.-A. Abro, M. A. K. Nagar, S. Kalhoro, and L. A. Rahoo, "Customer satisfaction in commercial bank of Sindh province a case study of bank Al Falah," *J. Libr. Philos. Pract.*, vol. 2331, no. May, pp. 1-13, 2019, [Online]. Available: <https://core.ac.uk/download/pdf/215161146.pdf>
- I. Junejo, D. Kumar, M. A. Khan, L. A. Rahoo, and J. Ahmed, "Use Of E-Commerce During Covid-19 And Purchase Intention: The Mediating Role Of Social Norms," *J. Posit. Sch. Psychol.*, vol. 7, no. 2, pp. 959-974, 2023.
- J. P. Anderson, "Computer security threat monitoring and surveillance," *Tech. Report*, James P. Anderson Co., 1980.
- M. Almseidin, M. Alzubi, S. Kovacs, and M. Alkasasbeh, "Evaluation of machine learning algorithms for intrusion detection system," in *2017 IEEE 15th international symposium on intelligent systems and informatics (SISY)*, IEEE, 2017, pp. 277-282.
- H. Kaur, G. Singh, and J. Minhas, "A review of machine learning based anomaly detection techniques," *arXiv Prepr. arXiv1307.7286*, 2013.
- S. K. Biswas, "Intrusion detection using machine learning: A comparison study," *Int. J. pure Appl. Math.*, vol. 118, no. 19, pp. 101-114, 2018.
- N. F. Haq, A. R. Onik, M. A. K. Hridoy, M. Rafni, F. M. Shah, and D. M. Farid, "Application of machine learning approaches in intrusion detection system: a survey," *IJARAI-International J. Adv. Res. Artif. Intell.*, vol. 4, no. 3, pp. 9-18, 2015.
- K. Jayakumar, T. Revathi, and S. Karpagam, "Intrusion Detection using Artificial Neural Networks with Best Set of Features.," *Int. Arab J. Inf. Technol.*, vol. 12, 2015.
- A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "Hybrid intrusion detection system based on the stacking ensemble of c5 decision tree classifier and one class support vector machine," *Electronics*, vol. 9, no. 1, p. 173, 2020.
- H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *Appl. Sci.*, vol. 9, no. 20, p. 4396, 2019.
- I. Vaccari, G. Chiola, M. Aiello, M. Mongelli, and E. Cambiaso, "MQTTset, a new dataset for machine learning techniques on MQTT," *Sensors*, vol. 20, no. 22, p. 6578, 2020.
- M. Al-Janabi, M. A. Ismail, and A. H. Ali, "Intrusion Detection Systems, Issues, Challenges, and Needs.," *Int. J. Comput. Intell. Syst.*, vol. 14, no. 1, pp. 560-571, 2021.
- V. Cherkassky and F. M. Mulier, *Learning from data: concepts, theory, and methods*. John Wiley & Sons, 2007.
- M. M. Taye, "Understanding of machine learning with deep learning: architectures, workflow, applications and future directions," *Computers*, vol. 12, no. 5, p. 91, 2023.
- A. Khan, N. Kim, J. K. Shin, H. S. Kim, and B. D. Youn, "Damage assessment of smart composite structures via machine learning: a review," *JMST Adv.*, vol. 1, no. 1, pp. 107-124, 2019.
- Z. Keita, "Classification in Machine Learning: An Introduction," *Datacamp*, 2013. <https://www.datacamp.com/blog/classification-machine-learning>
- Sonoo Jaiswal, "Linear Regression in Machine Learning," *Javapoint*, 2011. <https://www.javatpoint.com/linear-regression-in-machine-learning>
- Geeksforgeeks, "Understanding Nonlinear Regression with Examples," *Geeksforgeeks*, 2008. <https://www.geeksforgeeks.org/non-linear-regression-examples-ml/>

- L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "A hybrid strategy for krill herd algorithm with harmony search algorithm to improve the data clustering?," *Intell. Decis. Technol.*, vol. 12, no. 1, pp. 3-14, 2018.
- S. Jaiswal, "Clustering in Machine Learning," *Javatpoint*, 2013. <https://www.javatpoint.com/clustering-in-machine-learning>
- Y. Meidan *et al.*, "ProfilIoT: A machine learning approach for IoT device identification based on network traffic analysis," in *Proceedings of the symposium on applied computing*, 2017, pp. 506-509.
- H. Alkahtani, T. H. H. Aldhyani, and M. Al-Yaari, "Adaptive anomaly detection framework model objects in cyberspace," *Appl. Bionics Biomech.*, vol. 2020, pp. 1-14, 2020.
- M. A. Ferrag and L. Maglaras, "DeepCoin: A novel deep learning and blockchain-based energy exchange framework for smart grids," *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1285-1297, 2019.
- O. Alkadi, N. Moustafa, B. Turnbull, and K.-K. R. Choo, "A deep blockchain framework-enabled collaborative intrusion detection for protecting IoT and cloud networks," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9463-9472, 2020.
- M. Z. Khan, F. U. Zaman, M. Adnan, A. Imroz, and M. A. Rauf, "Comparative Case Study: An Evaluation of Performance Computation Between SQL And NoSQL Database," *Sindh J. Headways Softw. Eng.*, vol. 01, no. 02, pp. 14-23, 2022.
- Z. Elkhadir, K. Chougali, and M. Benattou, "An effective cyber attack detection system based on an improved OMPCA," in *2017 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, IEEE, 2017, pp. 1-6.
- D. Yi, F. Binwen, K. Xiaoming, and M. Qianqian, "Design and implementation of mobile health monitoring system based on MQTT protocol," in *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, IEEE, 2016, pp. 1679-1682.
- M. Hasan, M. M. Islam, M. I. I. Zarif, and M. M. A. Hashem, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," *Internet of Things*, vol. 7, p. 100059, 2019.
- M. H. Aysa, A. A. Ibrahim, and A. H. Mohammed, "Iot ddos attack detection using machine learning," in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, IEEE, 2020, pp. 1-7.
- S. H. Haji and S. Y. Ameen, "Attack and anomaly detection in iot networks using machine learning techniques: A review," *Asian J. Res. Comput. Sci.*, vol. 9, no. 2, pp. 30-46, 2021.
- N. Elmrabit, F. Zhou, F. Li, and H. Zhou, "Evaluation of machine learning algorithms for anomaly detection," in *2020 international conference on cyber security and protection of digital services (cyber security)*, IEEE, 2020, pp. 1-8.
- N. Li, M. Shepperd, and Y. Guo, "A systematic review of unsupervised learning techniques for software defect prediction," *Inf. Softw. Technol.*, vol. 122, p. 106287, 2020.
- I. Alrashdi, A. Alqazzaz, E. Aloufi, R. Alharthi, M. Zohdy, and H. Ming, "Ad-iot: Anomaly detection of iot cyberattacks in smart city using machine learning," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, 2019, pp. 305-310.
- S. Bagui, X. Wang, and S. Bagui, "Machine learning based intrusion detection for IoT botnet," *Int. J. Mach. Learn. Comput.*, vol. 11, no. 6, pp. 399-406, 2021.
- T. Wang, M. Z. A. Bhuiyan, G. Wang, L. Qi, J. Wu, and T. Hayajneh, "Preserving balance between privacy and data integrity in edge-assisted Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2679-2689, 2019.

- Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, and S. Khorsandroo, "Anomaly detection on iot network intrusion using machine learning," in *2020 International conference on artificial intelligence, big data, computing and data communication systems (icABCD)*, IEEE, 2020, pp. 1–5.
- D. Rani and N. C. Kaushal, "Supervised machine learning based network intrusion detection system for Internet of Things," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, 2020, pp. 1–7.
- N.-A. Stoian, "Machine Learning for anomaly detection in IoT networks: Malware analysis on the IoT-23 data set." University of Twente, 2020.
- K. Mahesh, "NSL-KDD Dataset for Intrusion Anomaly Detection System (IADS)," *Kaggle*, 2019.
https://www.kaggle.com/datasets/kiranmahesh/nslkdd?select=kdd_train.csv
- T. C. Havens, R. Chitta, A. K. Jain, and R. Jin, "Speedup of fuzzy and possibilistic kernel c-means for large-scale clustering," in *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, IEEE, 2011, pp. 463–470.
- A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.
- W. Wang, Y. Zhang, Y. Li, and X. Zhang, "The global fuzzy c-means clustering algorithm," in *2006 6th World Congress on Intelligent Control and Automation*, IEEE, 2006, pp. 3604–3607.
- S. Askari, "Fuzzy C-Means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development," *Expert Syst. Appl.*, vol. 165, p. 113856, 2021.
- J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2–3, pp. 191–203, 1984.
- Y. Ding and X. Fu, "Kernel-based fuzzy c-means clustering algorithm based on genetic algorithm," *Neurocomputing*, vol. 188, pp. 233–238, 2016.
- N. A. Ali, A. El Abbassi, and O. Bouattane, "Performance evaluation of spatial fuzzy C-means clustering algorithm on GPU for image segmentation," *Multimed. Tools Appl.*, vol. 82, no. 5, pp. 6787–6805, 2023.
- C. V. Nicholson, "A Beginner's Guide to Multilayer Perceptrons (MLP)," *Pathmind*, 2020.
<https://wiki.pathmind.com/multilayer-perceptron>
- DeepAI, "Multilayer Perceptron," *Mach. Learn. Gloss. Terms, Deep.*, 2020, [Online]. Available: <https://deepai.org/machine-learning-glossary-and-terms/multilayer-perceptron>
- A. Sharma, "Guided stochastic gradient descent algorithm for inconsistent datasets," *Appl. Soft Comput.*, vol. 73, pp. 1068–1080, 2018.
- J. Jin, S. Lai, S. Hu, J. Lin, and X. Lin, "GPUSGD: A GPU-accelerated stochastic gradient descent algorithm for matrix factorization," *Concurr. Comput. Pract. Exp.*, vol. 28, no. 14, pp. 3844–3865, 2016.
- A. Rajkumar and S. Agarwal, "A differentially private stochastic gradient descent algorithm for multiparty classification," in *Artificial Intelligence and Statistics*, PMLR, 2012, pp. 933–941.
- H. Zhang, F. Yu, J. Sun, X. Shen, and K. Li, "Deep learning for sea cucumber detection using stochastic gradient descent algorithm," *Eur. J. Remote Sens.*, vol. 53, no. sup1, pp. 53–62, 2020.
- K. Cohen, A. Nedić, and R. Srikant, "On projected stochastic gradient descent algorithm with weighted averaging for least squares regression," *IEEE Trans. Automat. Contr.*, vol. 62, no. 11, pp. 5974–5981, 2017.
- A. Khraisat, I. Gondal, and P. Vamplew, "An anomaly intrusion detection system using C5 decision tree classifier," in *Trends and Applications in Knowledge Discovery and Data Mining: PAKDD 2018 Workshops, BDASC, BDM, ML4Cyber, PAISI, DaMEMO*, Melbourne, VIC, Australia, June 3, 2018, *Revised Selected Papers 22*, Springer, 2018, pp. 149–155.