CLOUD ATTACK DETECTION BASED ON THREE-TIER DEEP-LEARNING BASED MODEL

***AJAY CHANDRA MK**
(Ajay Chandra Manukondakrupa)
***Department of Computer Science, Swami Ramanand Teerth Marathwada University, Vishnu Puri, Nanded, Maharashtra, 431606.**
***Corresponding Author Mail id: ajaymanukonda88@gmail.com**

ABSTRACT

The systematic identification and appropriate response to activities that are not sanctioned or malevolent within cloud computing involve detecting attacks directed toward resources, applications, data, or infrastructure hosted within a cloud-based environment. There exist certain drawbacks and impediments concomitant with the establishment and sustenance of efficacious attack detection systems within a cloud-based environment. To overcome the limitations of the cloud system, we proposed a hybrid deep learning-based attack detection model. The proposed methodology consists of (i) Data collection (ii) Pre-Processing (iii) Feature extraction (iv) Feature selection (v) Deep-learning based attack detection model. For this analysis, the data are collected from KDD Cup and CICIDS2017. In pre-processing, the data are pre-processed via Data cleaning (Missing data removal) and Z-Score Normalization. The pre-processed data are used to extract the features like central tendency (Mean, Median, and Mode), degree of dispersion (SD, Variance and Range) and other features (Packet count, traffic volume, Protocol distribution, and entropy). The extracted features are used to select the relevant features via a hybrid model which is a combination of Battle Royale Algorithm (BRA) and the Chicken Swarm Algorithm (CSA). Then, the selected features are used to detect the attacks via a novel three-tier deep-learning based model, which includes CNN, MLP, RNN and RBM. Initially, the selected features are moved on to the MLP, CNN, and then move on to the RNN. Finally, the outcome derived from the novelty M-RBM. The proposed methodology is executed via the PYTHON platform.

***Keywords**: Cloud Attack Detection, MLP, CNN, RBM, KDD Cup and CICIDS2017*

1. INTRODUCTION

In contemporary times, diverse categories of individual clients and enterprises employ cloud environments. The manifold cloud services are availed by all types of cloud users (CUs), although these CUs are predominantly unacquainted with the implementations (such as the interconnection of datacenters) and technical aspects (for instance, intrusion detection methods) of their Cloud Service Providers (CSPs) [1]. Cloud computing is a modern and efficient technological innovation that executes intricate computations on a broad scale, wherein a vast array of data, services, and storage solutions are readily accessible through the Internet [2]. Cloud computing is inherently fused with the methodologies of both industry and academia, and its expansion is occurring in a particularly robust manner [3]. In 2006, the inception of cloud computing was initiated by Google. The fundamental structure of cloud computing revolves around three primary levels of abstraction, namely system level, platform level, and application level [4]. Cloud computing represents an information technology resource management and operational framework that operates on a usage-based payment model and strives to minimize interactions between users and service providers. The customer's implementation approach determines the three distinct categories of cloud computing, which are public cloud, private cloud, and hybrid cloud [5].

In cloud computing, the system layer, platform layer, and application layer constitute the three different levels. The initial two layers mainly emphasize virtual machines (VM) and operating systems, while the third layer is primarily concerned with cloud-based applications like web-based applications [6]. IDS is a security technology designed to monitor network traffic or system activities for suspicious or unauthorized behavior. In the cloud, common Intrusion Detection Systems (IDS), such as snort and Web Application Firewalls (WAFs), are used to protect against web attacks [7]. There are several attacks in the cloud environment but the most common attack is DDoS. DDoS attacks leverage the pliable characteristics of cloud infrastructures, with the goal of overwhelming the extant resources and impeding the services that are being hosted on cloud platforms [8] [9]. The occurrence of cybersecurity breaches within cloud systems presents a significant threat to the continuous and uninterrupted availability of cloud services. This could result in a deficiency in fulfilling the expectations of the users and ultimately lead to their discontentment. Thus, the identification and prevention of such attacks against cloud systems are crucial [10] [11].

One of the most prevalent and uncomplicated methods for safeguarding a network resource is through the allocation of a distinct appellation and its corresponding confidential code [12]. Several nations have initiated the recognition of advantages derived from leveraging cloud computing in government institutions. Even though the adoption of cloud computing services can offer multifarious gains for government services, there are scarce instances of European nations having formulated governmental cloud strategy plans [13] [14].

The prime contribution of this work is arranged as follows:

- ✓ To propose a hybrid model which is a combination of (BRA and CSA) to select the relevant features from the extracted features.
- ✓ To propose M-RBM to detect the cloud attacks accurately.

The rest of the paper is arranged as follows, Sec (2) of the research investigates the presented papers. Sec (3) shows the proposed methodology. Sec (4) displays the outcome of the research. The paper accomplished in Sec (5).

2. LITERATURE REVIEW

In 2022, Abdullayeva [16] suggested a machine learning technique that accurately clusters network data to find DDoS attacks has been suggested. The methodology makes use of a feature selection strategy to improve the effectiveness of data clustering. The PCA algorithm has been utilized to deliver the feature selection. The DBSCAN, Agglomerative Clustering, and k-means algorithms are used for the dataset created using the chosen features. In the experiment, the clustering outcomes of the approaches employing fewer characteristics outperformed the outcomes of the methods employing all features on all parameters. The PCA + DBSCAN, PCA + Agglomerative, and PCA + k-means algorithms all outperformed the standard methods in terms of the Adjusted Rand Index metric, reaching values of 0.8989, 0.9130, and 0.9094, respectively.

In 2021, Agarwal *et al.* [17] introduced a method to efficiently detect attacks, using a P-estimation detection approach. A number of deep learning LSTM models are trained for this purpose using the web server logs. In order to deploy the proper detection model, an estimate of the attack % is first calculated. This method, which retrains and updates the detection models on a regular basis, accounts for the dynamic nature of websites where the popularity of web pages might alter over time. With a False Negative Rate (FNR) and False Positive Rate (FPR) of 0.0059% and 0.0%, respectively, this strategy exceeds all currently used FRC detection methods to the best of the authors' knowledge.

In 2020, Kushwah*et al.* [18] introduced a novel method of detecting DDoS assaults in a cloud computing setting. Voting Extreme Learning Machine (V-ELM), a kind of artificial neural network, is used to construct the proposed system. Using the NSL-KDD dataset and the ISCX intrusion detection dataset as two benchmark datasets, experiments were conducted to assess the performance of the suggested system. Experiments have demonstrated that the suggested system accurately detects attacks with an NSL-KDD dataset accuracy of 99.18% and an ISCX dataset accuracy of 92.11%.

In 2020, Rani and Geethakumari [19] suggested effective algorithms for secure data transmission and AFA detection. The B-tree Huffman Encoding (BHE) algorithm is used to compress the data packets before the packet marking approach is used to protect the sender's IP address. The Modified Elliptic Curve Cryptography (MECC) algorithm is then suggested for securely transferring the data, encrypting the data packets before transmitting them to a receiver. A Deep Learning Modified Neural Network (DLMNN) classifier is used at the receiver side to evaluate the IP-address of data packets that have been received. The proposed method's experimental findings are compared to those of the standard procedures in terms of performance measures.

In 2023, Pasha *et al.* [20] suggested a mathematical model that could be used to implement a mitigation plan. As a result, we put forth the Hybrid Approach for Low-Rate DDoS Detection (HA-LRDD) as a new algorithm. The algorithm uses a deep auto encoder and convolutional neural networks (CNN) to enable artificial intelligence (AI). Another approach that we developed, termed Dynamic Low-Rate DDoS Mitigation (DLDM), lessens the effects of an assault after they have been identified. Additionally, it guarantees that the attack is stopped and that the infrastructure keeps running. The suggested framework can detect and mitigate low-rate DDoS assaults to maintain an acceptable level of service in cloud computing settings, according to a thorough simulation study.

In 2023, Samunnisa*et al.* [21] proposed effective hybrid clustering and classification models for implementing an anomaly-based IDS for malicious attack type classifications like normal (no intrusion), DoS, Probe, U2R, and R2L. The effectiveness of the proposed models is tested using two different threshold values (e), 0.01 & 0.5. The studies were carried out on the NSL-KDD and KDDcup99 test datasets. The effectiveness of the suggested methodology has been examined using the metrics of detection rate, false alarm ratio, and accuracy. After implementing the suggested method, it was demonstrated that K-means with random forest had better classification accuracy, detection rate, and false alarm rate of 99.85%, 99.78%, and 0.09% on the NSL-KDD dataset and 98.27%, 98.12%, and 2.08% on the KDDcup99 dataset, respectively.

In 2023, Abdullayeva [22] investigated a new cloud computing reference model that consists of the components that make up various cloud computing layers is suggested. In order to offer security for cloud systems, this study investigates the cyber security concerns of cloud computing service models and builds an attack model. It provides an explanation of laws and regulations pertaining to cloud computing's cyber security. Clarification of the cyber security and cyber resilience ideas of cloud systems is offered in accordance with security considerations. The development of intelligent cloud systems' cyber resilience architecture.

In 2021, Kushwah*et al.* [23] presented an enhanced Self-Adaptive Evolutionary Extreme Learning Machine (SaE-ELM)-based DDoS assault detection system. The SaE-ELM model is enhanced by the addition of two more characteristics. It can first adopt the most appropriate crossover operator. Second, it can automatically decide how many hidden layer neurons are necessary. These qualities enhance the model's capacity for learning and classification.Four datasets, namely NSL-KDD, ISCX IDS 2012, UNSW-NB15, and CICIDS 2017, are used to assess the proposed system. With the NSL-KDD, ISCX IDS 2012, UNSW-NB15, and CICIDS 2017 datasets, it achieves detection accuracy of 86.80%, 98.90%, 89.17%, and 99.99%, respectively.

In 2021, Shah *et al.* [24] proposed a techinique, ICMP detection and mitigation model (EDOS-IDM). It can identify and counteract volumetric and typical behavioural ICMP traffic attacks. Since the proposed technique utilizes the fewest resources out of all the mitigation techniques, the results are contrasted with those from the Normal Behavioural ICMP traffic attack. Our research indicates that there is no such strategy that can counteract a typical behavioural ICMP traffic attack. On a test platform for the OpenStack production Cloud Environment, the technique is practically tried out and assessed. The technique is demonstrated to reduce excess resource use and customer costs in a cloud computing environment, according to the findings. Table (1) shows that various authors reviews about research gaps

2.1 Problem Statement

Table 1: Reviews by various authors of research gaps

Author	Aim	Research Gap
Abdullayeva, 2022	To put forth a machine learning technique for detecting Distributed Denial of Service (DDoS) assaults in the network layer of the cloud infrastructure for e-government.	Only focuses on the attack detection in the network layer
Agarwal *et al.* 2021	To suggest a deep learning LSTM model-based P-estimation detection method for FRC (Fraudulent Resource Consumption) attacks on cloud servers.	Didn't include the larger dataset and in a real-world cloud environment.
Kushwah*et al.* 2020	To put forth a new method using a voting extreme learning machine (V-ELM) classifier to detect distributed denial of service (DDoS) assaults in cloud computing environments.	Examining the suggested system's potential to scale and its effectiveness in large-scale cloud networks.
Rani and Geethakumari 2020	To put out a technique using MECC and DLMNN for safe data transfer and the spotting of anti-forensic attacks in a cloud context.	Didn't include the load balancing concept for more efficient system

Pasha *et al.* 2023	To propose a Low-Rate DDoS Attack Detection Framework (LRDADF) that is capable of spotting and reducing low-rate DDoS assaults in cloud computing settings.	Desire to research other deep learning techniques to better effectively combat low-rate DDoS attacks.
Samunnisa*et al.* 2023	To propose a practical hybrid clustering and classification model for the implementation of an anomaly-based intrusion detection system (IDS)	Didn't include the methods of modeling network traffic and attack behavior that best represents the parameters of individual attacks.
Abdullayeva, 2023	To bring up a fresh reference model for cyber security for intelligent cloud computing systems that is capable of effectively identifying threats and supplying the cloud infrastructure with both security and resilience online	Didn't detect the SLA violations in the system based on DL technologies.
Kushwah*et al.* 2021	To provide an improved Self-Adaptive Evolutionary Extreme Learning Machine (SaE-ELM) model for DDoS	-

	attack detection in cloud computing.	
Shah *et al.* 2021	To suggest a novel method for detecting and mitigating volumetric and normal behavioural ICMP traffic threats in cloud computing environments (CCE) that uses software-defined networks (SDN) and is named the ICMP detection and mitigation model (EDOS-IDM).	Didn’t consider the TCP SYN Flooding and UDP Flooding attacks.

3. PROPOSED METHODOLOGY

3.1 Overview

The primary target of this research paper is to detect the attack in cloud systems using a three-tier- deep learning model. The proposed methodology developed as follows: (1) Data collection (2) Pre-Processing (3) Feature extraction (4) Feature selection (5) Three-tier deep learning based Attack detection. (6) Model Evaluations. Figure (1) illustrates the overall architecture of the proposed model.

Step 1- Data collection: The data have been collected from KDD Cup and CICIDS2017

Step 2- Pre-Processing: The collected data has been pre-processed via Data cleaning techniques (missing data removal) z-score normalization.

Step 3- Feature Extraction:

- Central tendency measures: Calculate features such as mean, median and mode.
- Degree of Dispersion Measures: Measures the features like variance, Standard deviation and Range.
- Other Features for Cloud-Based Intrusion Detection: Calculate the other features like Packet Count, Protocol Distribution, Traffic Volume and Entropy.

Step 4- Feature Selection: the features are selected through a Hybrid meta-heuristic optimization model, which is a combination of Battle Royale Algorithm (BRA) and Chicken Swarm Algorithm (CSA).

Step 5- Three-tier deep learning based Attack detection model

Step 6- Model Evaluations: The execution of the hybrid deep learning model will be estimated using performance metrics

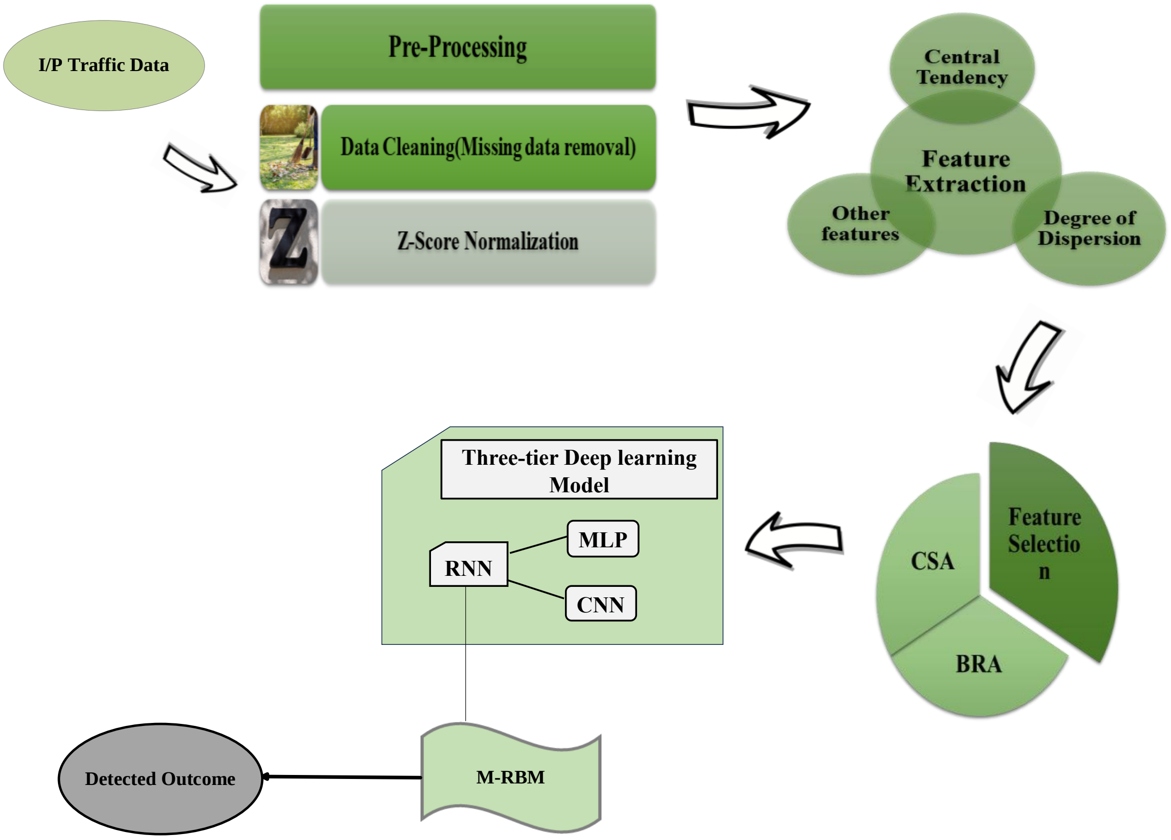


Figure 1: Overall architecture of the proposed model.

3.1.1. Data Collection

The process of data collection entails the systematic acquisition and organization of data from diverse sources, serving as the fundamental basis for subsequent analysis. The level of excellence of the collated data has a direct influence on the reliability and effectiveness of ensuing revelations and achievements. In this work the data are collected from KDD Cup and CICIDS2017 datasets.

3.1.1.1 KDD Cup

The KDD training dataset is made up of 10% of the original dataset, or roughly 494,020 single connection vectors, each containing 41 features and being labelled with only one specific attack type, i.e., either normal or an attack. Each vector has a label designating it as either normal or an attack, with one attack kind in particular. Attacks are defined as deviations from "normal behaviour" or anything that is not "normal". Attacks with the label "normal" are records that behave normally. For memory-constrained machine learning techniques, a 10% training dataset is also given in a reduced version. 19.69% of connections in the training dataset are normal, while 80.31% are attack connections. The most often utilized attack vector in network attacks is KDD CUP 99.

3.1.1.2 CICIDS2017

The CICIDS2017 dataset comprises safe and recent common attacks that closely mirror actual real-world data (PCAPs). Additionally, it contains the output of the CICFlowMeter network traffic analysis, labelled flows based on the time stamp, source and destination IP addresses, source and destination ports, protocols, and attack (CSV files). The definition of extracted characteristics is additionally available. Building this dataset with realistic background traffic generation as our top priority. We profiled the abstract behaviour of human interactions using our suggested B-Profile system (Sharafaldin, et al. 2016), which also creates naturalistic benign background traffic. On the basis of the HTTP, HTTPS, FTP, SSH, and email protocols, we constructed the abstract behaviour of 25 users for this dataset. The data collection period lasted for 5 days, from Monday, July 3, 2017, from 9 a.m. through Friday, July 7, 2017, at 5 p.m. Monday is a typical day with only light traffic. Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet, and DDoS are some of the attacks used. Tuesday, Wednesday, Thursday, and Friday saw their morning and afternoon executions.

3.1.2 Pre-Processing

The collected data are pre-processed via Data cleaning techniques (missing data removal) and z-score normalization. Data preparation refers to the systematic procedure of converting unrefined data into a comprehensible and efficient format. It is commonly acknowledged that data, particularly real-world or recent data, frequently encounters deficiencies such as manual errors and inconsistent formatting. The process of data pre-processing aims to eliminate these inadequacies while simultaneously augmenting the efficacy and comprehensiveness of datasets utilized for data analysis. Figure 2 shows the Pre-processing techniques. Figure 2 shows the data cleaning techniques which is used in this paper

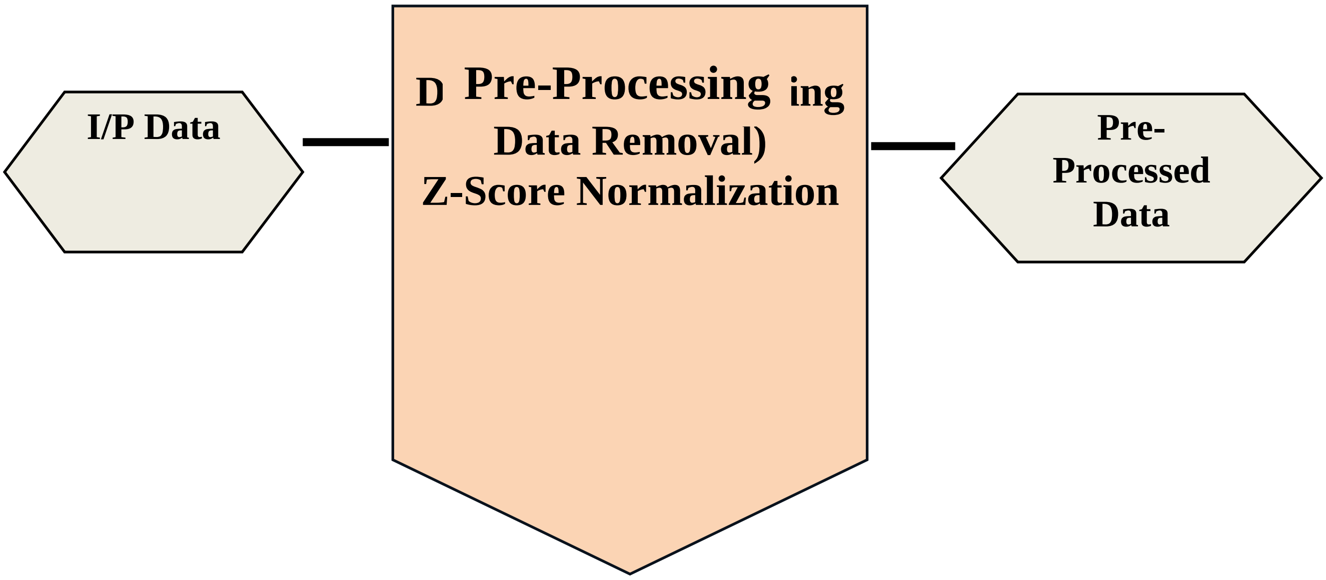


Figure 2: Pre-Processing

3.1.2.1 Data Cleaning (Missing Data Removal)

Data cleaning entails the rectification or elimination of inaccurate, corrupted, improperly structured, replicated, or unfinished data within a given dataset. In the event of combining multiple data sources, there are numerous instances whereby data could be replicated or misidentified.

The process of detecting and addressing absent data points within a given dataset is crucial for guaranteeing the precision, dependability, and significance of ensuing analytical outcomes. The absence of data may transpire due to a range of factors, namely inaccuracies in data inputting, unfinished surveys, technological difficulties, or deliberate omissions.

3.1.2.2 Z-Score Normalization

Z-score normalization pertains to the procedure of standardizing each datum in a given dataset so that the arithmetic mean of all the values is rendered to be 0 and the standard deviation to be 1.

$$z = \frac{x-\mu}{\sigma} \qquad (1)$$

Where x denotes the original value, μ is the mean of the data and σ denotes the standard deviation.

3.1.3 Feature Extraction

The pre-processed data are move on to the feature extraction phase to extract the features like central tendency measures, Degree of dispersion measures and other feature for cloud-based intrusion detection. Feature extraction used to make the process more accurate and it increases the prediction power of the algorithm to select the most relevant features. Figure 3 shows the feature extraction techniques.

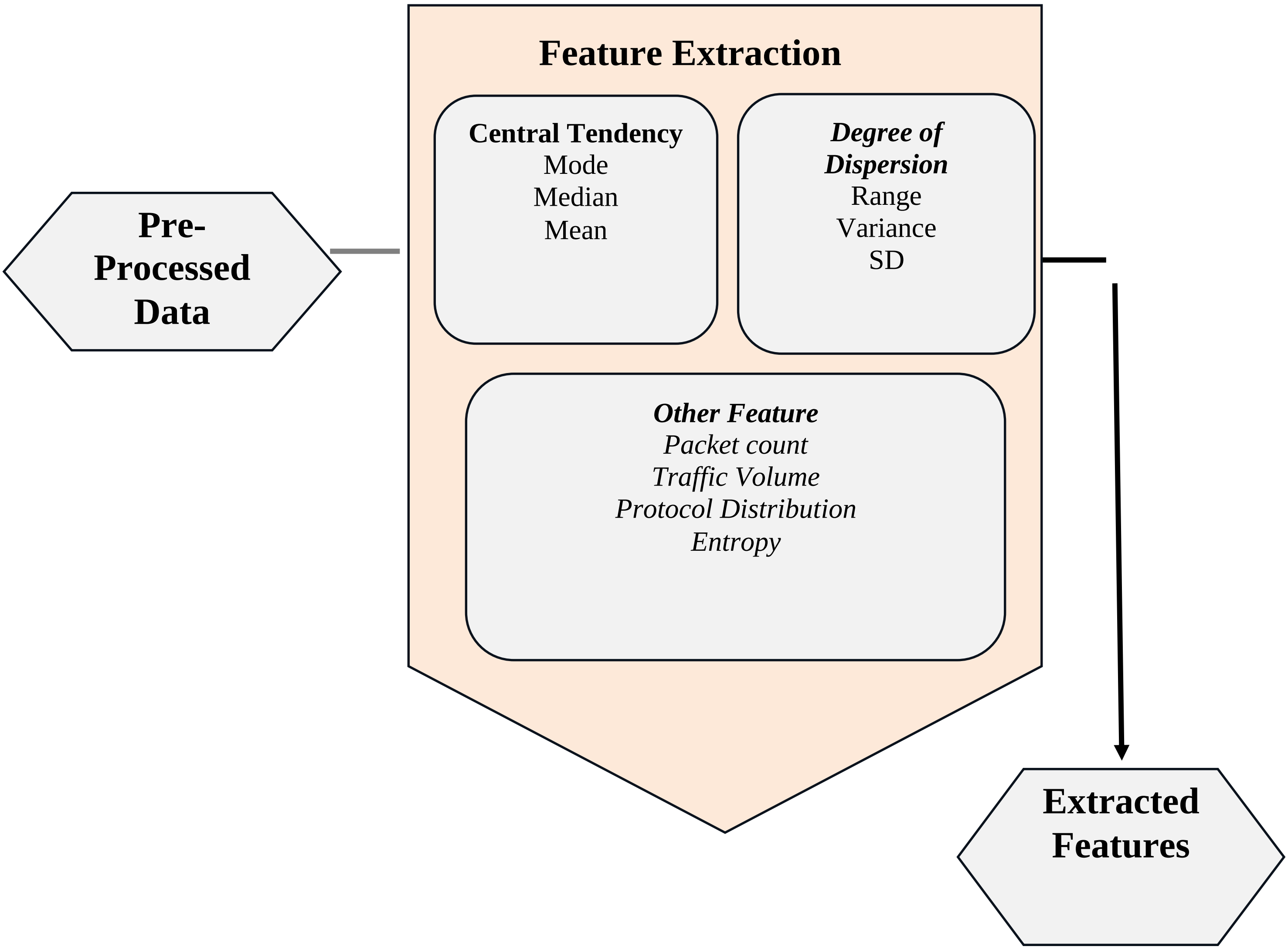


Figure 3: Feature Extraction Process

3.1.3.1 Central Tendency

A measure of central tendency, also known as measures of center or central location, constitutes a summary measure endeavoring to depict an entire dataset by utilizing a solitary value that represents the middle or center of its distribution. Three principal measures of central tendency encompass the mode, median, and mean. Table (2) shows the Definition and formula for Central Tendency.

Table 2: Definition and formula for Central Tendency.

Central Tendency	**Definition**	**Formula**
Mode	The most frequently occurring value in a dataset. In intrusion detection, it might indicate common attack patterns that are being repeatedly attempted.	$Mode = l + h\frac{(f_m - f_1)}{(f_m - f_1) + (f_m - f_2)}$ l is the lower limit of the modal class. h is the size of the class interval. f_m is the frequency of the modal class. f_1 is the frequency of the class that comes just before the modal class. f_2 is the frequency of the class that comes just after the modal class.
Median	The middle value in a sorted dataset. It can be used to find the central value of a feature, helping to avoid skewness caused by outliers.	$Median = l + [\frac{(\frac{N}{2} - c}{f})] \times h$ l = lower limit of the median class, N = Total frequency, c = Cumulative frequency of class before the median class, f = Frequency of the median class, h = Class width (Upper limit - Lower limit)
Mean	The average of a set of values. In intrusion detection, it could be used to analyze the average number of intrusions attempts over a specific time period.	$\bar{x} = \frac{\sum x_i}{N}$ $\bar{x}$= mean value, x_i = initial number of observations, N Total number of observations

3.1.3.2 Degree of Dispersion

The dispersion of data serves as an essential tool for comprehending the distribution of data. It enables one to grasp the varying nature of data and furnishes valuable insights regarding its distribution. To gain an understanding of the distribution data, one may employ methods such as Range, Variance, and Standard Deviation. Table 3 shows the definition and formula for Degree of Dispersion.

Table 3: Definition and formula for Degree of Dispersion.

Degree of Dispersion	**Definition**	**Formula**
Range	The difference between the maximum and minimum values in a dataset. It provides a simple measure of data spread.	$Range = Highest\ Value - Lowest\ Value$
Variance	A measure of how much the values in a dataset vary from the mean. Higher variance can indicate higher diversity in attack attempts.	$\sigma^2 = \frac{\sum(x_i - \mu)^2}{N}$ $i = 1,2,3..$ $\mu = population\ mean$ $N = number\ of\ data\ points$
Standard Deviation	The square root of the variance. It gives an idea of the spread of data points around the mean.	$Standard\ deviation = \sqrt{\sigma^2}$

3.1.3.3 Other Feature for cloud-based intrusion detection

Table 4 shows the definition and formula for other features like packet count, traffic volume, protocol distribution and entropy.

Table 4: Definition and formula for other features (packet count, traffic volume, protocol distribution and entropy)

Other Features	**Definition**	**Formula**
Packet Count	The total number of packets exchanged in a network flow. Unusual spikes in packet count might indicate a potential intrusion attempt.	$Packet\ Count\ =\ Total\ number\ of\ Packets$
Protocol Distribution	The distribution of different network protocols being used. Sudden changes might indicate an attack trying to exploit specific protocol vulnerabilities.	
Traffic Volume	The amount of data transferred in a network flow. Unusual high traffic volumes could suggest a potential Distributed Denial of Service (DDoS) attack.	
Entropy	A measure of randomness or uncertainty in data. High entropy values could indicate anomalous or unexpected behavior.	$E = -\sum_{i=1}^{N} p_i log_2 p_i$ E denotes the entropy, p_i represents the probability of the ith event, N is the number of distinct events.

3.1.4 Feature Selection

The extracted features are move on to the feature selection phase to select the relevant features. Feature selection entails reducing the input variables to a model by utilizing solely pertinent data, while eliminating extraneous noise in the data. The objective of employing such a technique is to enhance the precision of the process and elevate the forecasting capabilities of the algorithms. This is accomplished by selecting the most critical variables, while simultaneously eliminating the redundant and irrelevant ones. In this paper a novel hybrid model is used to select the features which is a combination of BRA and CSA. Figure 4 illustrate the Feature selection process

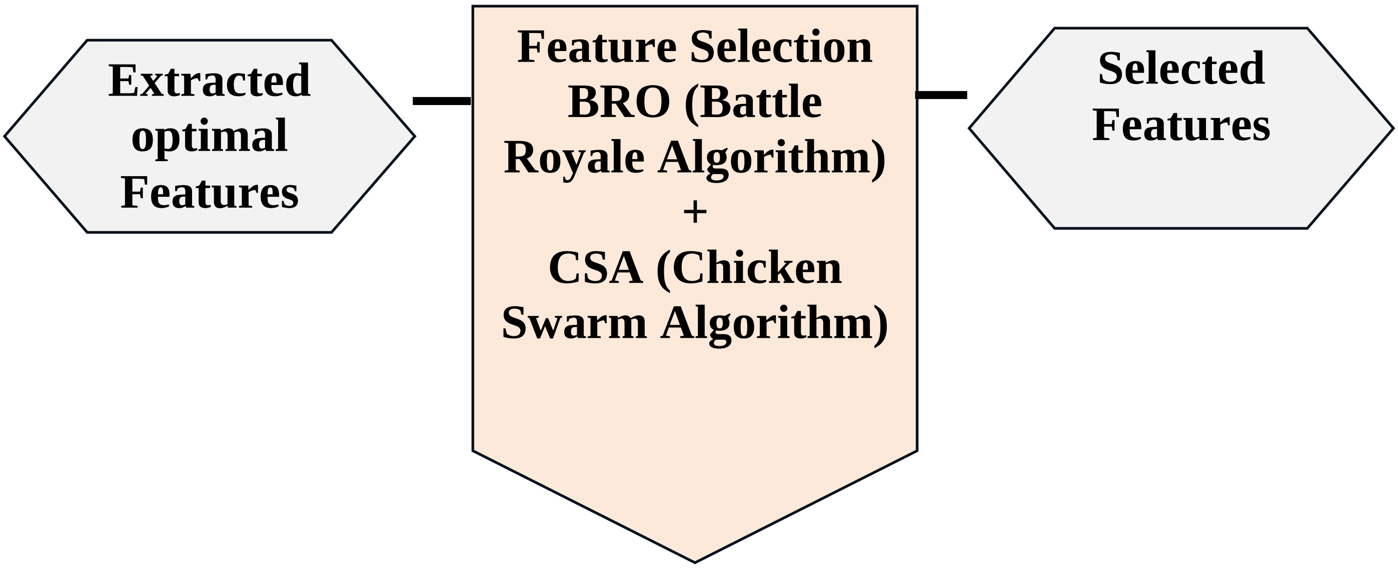

Extracted optimal Features
Feature Selection BRO (Battle Royale Algorithm) + CSA (Chicken Swarm Algorithm)
Selected Features

Figure 4: Feature selection process

- **Battle Royale Algorithm**

Some iterations of the battle royale gaming genre commence with players leaping out of an aircraft and subsequently utilizing a parachute to descend onto the designated game map. Additionally, akin to other swarm-based algorithms, BRO instigates with a stochastic population that is evenly spread throughout the problem space. Subsequently, each individual soldier or player endeavors to inflict damage upon the closest nearby soldier by discharging a weapon. Consequently, soldiers occupying more advantageous positions inflict harm on their closest neighbors. When an instance arises in which a soldier inflicts harm upon another soldier, the extent of the harm incurred by the latter is augmented by a factor of one. These occurrences are subject to mathematical computation through the implementation of the formula $X_i.damage = X_i.damage + 1$, where $X_i.damage$ denotes the degree of injury sustained by the ith soldier within the given population. Furthermore, it is worth noting that soldiers exhibit a proclivity to alter their position subsequent to incurring damage, thereby launching attacks against their opponents from an alternate vantage point. Henceforth, in order to direct attention towards exploitation, the solder that has undergone impairment shifts towards a specific locus that lies somewhere between its prior location and the optimal position achieved thus far, i.e., the elite participant. These interactions are subjected to mathematical computations, as follows:

$$X_{Dam,D} = X_{Dam,D} + R(X_{best,D} - X_{Dam,D}) \qquad (2)$$

whereas the value of R is a number that has been generated on a random basis, with a uniform distribution that ranges between 0 and 1, $X_{Dam,D}$ refers to the location of the soldier who has incurred damage in the Dth dimension. Furthermore, in the event that impaired soldiers have the capability to inflict harm upon their adversary during the following iteration, the variable $X_i.damage$ shall be reset to zero. Additionally, in order to prioritize the aspect of exploration, should the damage magnitude of a soldier exceed the predetermined threshold value, said soldier shall meet their demise and be reborn at a random location within the feasible problem space. Once reborn, the variable $X_i.damage$ shall be reset to zero. After conducting numerous trials and making adjustments, it was discovered that the ideal threshold value for our needs was 3. This strategic decision effectively mitigates the risk of premature convergence and affords us superior opportunities for exploration. The reentry of a previously deceased soldier into the problem space can be articulated as follows:

$$X_{Dam,D} = R(Ub_D - Lb_d) + lb_D \qquad (3)$$

The problem space refers to Lb_d and Ub_d as the lower and upper bounds of dimension D, respectively. Moreover, with each iteration D, the feasible search space of the problem contracts towards the optimal solution. The initial value was $\Delta = \log_{10}(MaxCicle)$ but then $\Delta = \Delta + round\left(\frac{\Delta}{2}\right)$. Here MaxCicle is the maximum number of generations. This particular interaction yields significant contributions towards the dual objectives of exploration and exploitation. As a result, the lower and upper bounds will undergo updates in the following manner.

$$Lb_D = X_{best,D} - sd(\overline{X_D}) \qquad (4)$$

$$Ub_D = X_{best,D} + sd(\overline{X_D}) \qquad (5)$$

Here, $sd(\overline{X_D})$ the represents the standard deviation of the entire population in dimension D. The $X_{best,D}$ signifies the position of the best solution that has been discovered so far. Therefore, in the event that the Lb_D/Ub_D surpasses the initial lower/upper bound, it is then established at the original Lb_D/Ub_D. Moreover, with regards to the notion of elitism, the preeminent player or soldier identified during each iteration is retained and regarded as an elite.

It is vital to take into account the computational complexity of the proposed approach, which relies on two crucial elements - the size of the population and the maximum number of iterations - as well as the various aspects of the problem at hand. It is noteworthy that each solution should be subjected to a comparison process with all other solutions to establish its Euclidean distance. Provided a population magnitude of n, computing all feasible solutions is a strenuous chore with a difficulty of $o(N^2)$. Consequently, the computational complexity of BRO, given the number of iterations m is $o(N^3)$., which further underscores the intricacies of the approach.

- **Chicken Swarm Algorithm**

This algorithm employs a hierarchical order similar to that found within a chicken swarm, and utilizes the collective food-searching mechanism of the swarm. The chickens within the group are separated into different categories, namely dominant roosters, hens, and chicks, based upon their respective fitness values. The designation of roosters is given to chickens exhibiting superior food-searching ability or fitness, while the label of chicks is attributed to those exhibiting the least. Hens, on the other hand, are designated to chickens demonstrating intermediate food-searching ability or fitness. Random assignment is employed in establishing the mother-child relationship. Subsequently, the hierarchical order and mother-child relationship are both subject to updates after every G time steps. The algorithm effectively utilizes the innate behavior observed in hens wherein they tend to follow their group mate rooster and chicks follow their mother while seeking food. Furthermore, it is anticipated that these chickens will engage in food-scratching activity, leading to a competitive environment within the group. This algorithm is segregated into two stages, namely initialization and update.In the initialization phase, CSO defines the population size and associated parameters, including the number of roosters, hens, chicks, and mother hens. The first step involves defining the parameter G. Next, the fitness values of the initial chicken population are evaluated, and a hierarchical order is established based on this assessment. The algorithm operates under the following assumptions:

- The number of hens is the highest in the group

- All the hens are not mother hens
- The mother hens are selected randomly from the set of hens
- The number of chicks is less than hen

There exists variation in the food-seeking aptitude among roosters, hens, and chicks. During the update phase, the initial population's fitness values are revised in accordance with the members' diverse food-seeking abilities. The ability of roosters to locate food is contingent upon their level of physical aptitude, and the algorithm governing their search strategy is articulated as follows:

$$x_{i,j}^{T+1} = x_{i,j}^{T} \times (1 + Randn(0,\sigma^2)) \qquad (6)$$

$If\ f_i \leq f_k$

$$\sigma^2 = 1 \qquad (7)$$

Else

$$\sigma^2 = Exp\left(\frac{(f_k - f_i)}{|f_i| + \gamma}\right) \qquad (8)$$

The Gaussian distribution with a mean of 0 and a standard deviation of σ^2 can be represented by the function randn (0,σ^2). The fitness value of the corresponding x is denoted as f. The index of the randomly selected rooster is indicated by k. Furthermore, a minute constant value, e, is implemented in order to prevent zero division errors. Hens exhibit a proclivity to pursue their fellow roosters in their foraging endeavors. Furthermore, there exists a propensity among the poultry to purloin sustenance obtained by their counterparts. The corresponding mathematical expression for their update formula is illustrated below:

$$x_{i,j}^{T+1} = x_{i,j}^{T} + s1 \times Rand \times \left(x_{i,j}^{T} - x_{i,j}^{T}\right) + s2 \times Rand(x_{r2,j}^{T} - x_{i,j}^{T}) \qquad (9)$$

$$s1 = Exp\left(\frac{f_i - f_{r1}}{abs(f_i) + \gamma}\right) \qquad (10)$$

$$s2 = Exp\ (f_{r2} - f_i) \qquad (11)$$

where $Rand$ denotes a stochastically generated numerical value ranging inclusively from 0 to 1. The variable $r1 \in [1, N]$ designates an index associated with a particular group, to which the subject hen belongs. Meanwhile, N connotes a cohort mate of the aforementioned hen. In addition, $r2$ represents another index pertaining to either a rooster or hen, which is selected in a random fashion, subject to the condition that $r1$ and $r2$ are not equivalent.

The inherent inclination of juvenile birds to trail their maternal figure can be expressed in a mathematical context as demonstrated below:

$$x_{i,j}^{T+1} = x_{i,j}^{t} + fl \times (x_{M,j}^{T} - x_{i,j}^{t}) \quad (12)$$

the variable $x_{M,j}^{T}$ denotes the spatial coordinates of the mother of the ith avian offspring. The parameter fl is indicative of the propensity of the chick to pursue its maternal figure, and is typically selected from the range of 0 to 2.

3.1.5 Three-tier deep learning based Attack detection model

The selected features are move on to the attack detection phase. A three-tier deep learning model used to detect the attacks; it is a combination of CNN, MLP, RNN and M-RBM. Figure 5 illustrate the attack detection model.

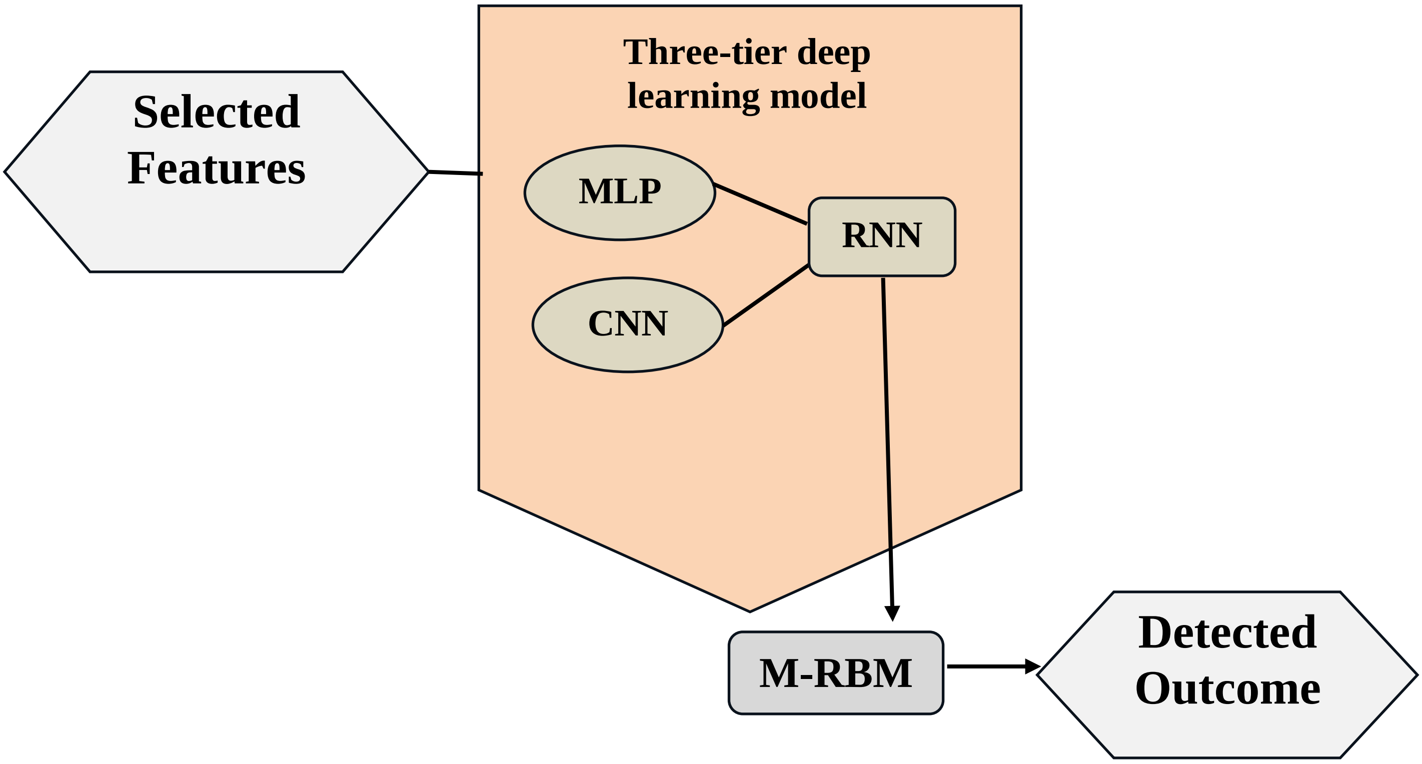


Figure 5: Three-Tier Attack Detection Model

- **CNN**

Using grid-like matrices, convolutional neural networks (CNN), which have evolved from ANN, are mostly used to extract features from datasets. Convolutional, pooling, and fully linked layers are among the layers that make up a CNN in most cases. Convolutional layers perform the work of extracting features from the input data, and the task of down sampling the feature maps by pooling layers reduces their spatial dimensions. The fully linked layers at the network's end carry out classification or regression tasks based on the learned properties. Eq (13) shows the mathematical representation of CNN,

$$x_{i,j}^{l} = \sum_{a=0}^{m-1}\sum_{b=0}^{m-1} w_{ab}\, y_{(i+a)(j+b)}^{l-1} \qquad (13)$$

- **MLP**

The implementation of multiple layers of perceptrons makes the MLP the most complex architecture in artificial neural networks, making it a popular choice for deep learning applications. This manuscript aims to provide guidance on constructing a neural network using the prevalent deep learning framework, TensorFlow. To comprehend the notion of a Multi-Layer Perceptron, it is necessary to develop one from scratch utilizing Numpy. MLPs are not capable of generating any type of output. Multiple layers of input nodes characterize an MLP, connected as a directed graph between the input and output layers. Backpropagation is utilized to train MLP networks. MLP represents a method of deep learning. The use of MLP networks is prevalent in supervised learning settings. A commonly employed learning algorithm for MLP networks is referred to as the back propagation algorithm.

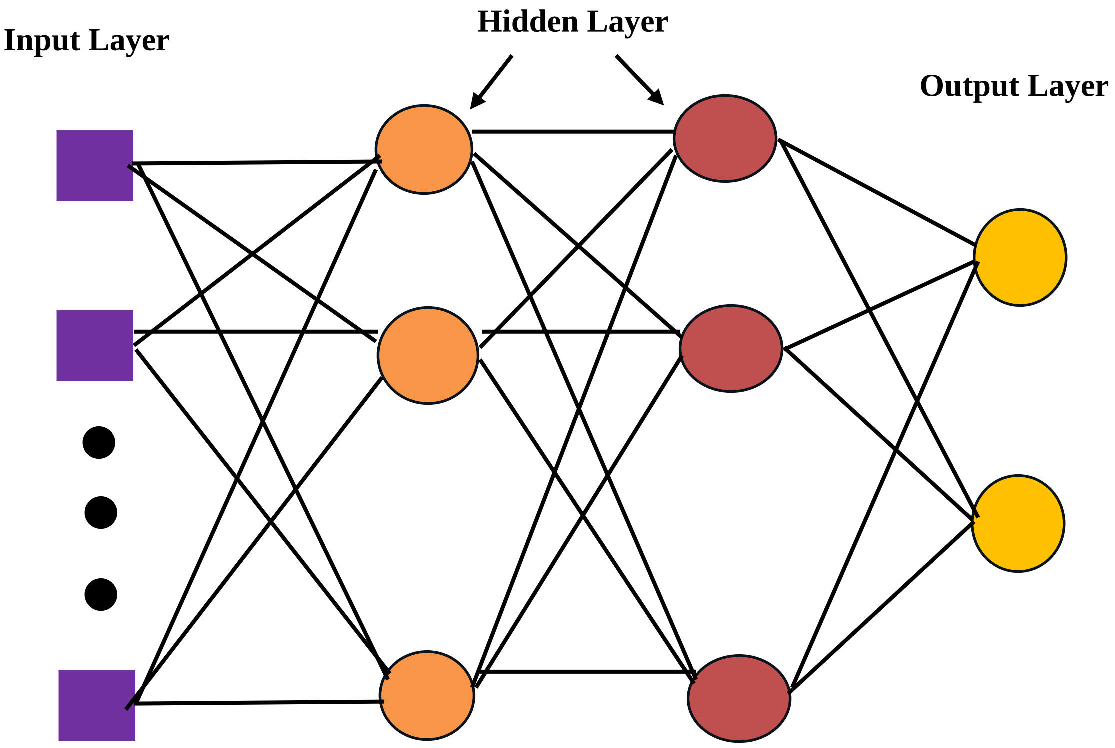


Figure 5: Pictorial Representation of multi-layer perceptron

In the illustration of the multi-layer perceptron depicted above, one may observe that the quantity of inputs is three, consequently, there are three nodes in the input layer. Moreover, the hidden layer encompasses three nodes. The output layer, on the other hand, yields two outputs, thereby signifying the presence of two output nodes. The nodes that form the input layer receive input and transmit it for further processing. In the aforementioned diagram, the nodes in the input layer send their results to all three nodes in the hidden layer. The processing and transmission of information to the output layer is carried out by the hidden layer. The utilization of the sigmoidal formula by each neuron in the multi-layer perceptron enables the transformation of real numerical inputs into values that range from 0 to 1. Eq (14) shows the mathematical representation of MLP,

$$(X) = \frac{1}{(1+Exp(-X))} \tag{14}$$

- **RNN**

An ANN architecture called a recurrent neural network (RNN) is made primarily to handle sequential input. RNNs have a recurrence relation that enables them to capture the temporal dependencies contained in sequential data, unlike other neural networks that process data linearly during the feed-forward and back-propagation processes. Natural language processing, speech recognition, machine translation, time series forecasting, music creation, and other sequential data-related applications have all had success with RNNs. They are effective tools for modelling and predicting data sequences because they can capture temporal dependencies. $H^{(t)}$represents the hidden units, $O^{(t)}$ denoted the outputs and $\hat{Y}^{(t)}$ display the targets.

$$A^{(t)} = B + wH^{(t-1)} + Ux^{(t)} \tag{15}$$

$$H^{(t)} = \tanh\left(A^{(t)}\right) \tag{16}$$

$$O^{(t)} = C + vH^{(t)} \tag{17}$$

$$\hat{Y}^{(t)} = softmax(O^{(t)}) \tag{18}$$

- **RBM**

Unsupervised learning is accomplished using artificial neural networks, namely the Restricted Boltzmann Machine (RBM). A probability distribution can be learned across a set of input data using this kind of generative model. Contrastive divergence, a variation of the stochastic gradient descent algorithm, is used to train the RBM. To increase the likelihood of the training data, the network modifies the weights of the connections between the neurons during training. After training, the RBM can be used to create fresh samples based on the discovered probability distribution. Because connections between neurons in the same layer are not allowed, the RBM is referred to as "restricted". And vice versa, every neuron in the visible layer is only related to neurons in the hidden layer. By lowering the dimensionality of the input, this enables the RBM to learn a compressed version of the input data.Eq (19) shows the mathematical representation of RBM,

$$E(v,h) = -A^t v - b^t h - v^t wh \qquad (19)$$

v denotes the visible layer, h denotes the hidden layer, A presents the weight matrix that connects the visible layer to hidden layer, b is the bias vector, w is the weight matrix that connects the hidden layer to visible layer.

4. Result and discussion

4.1 Experimental Setup

This work has been implemented in PYTHON. The projected model has been tested using two databases: database 1-(https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data) and database 2-(https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv). Among the collected data 70% of the information has been used for training, and the rest 30% has been utilized for testing purposes. The proposed model has been validated over the existing models MLP, CNN, RNN, RB, RF, CHOS, respectively. The assessment has been made in terms of accuracy, precision, MCC, F-measure, FPR, NPV, sensitivity, FNR, and specificity.Below are the performance measures and their calculation algorithms.

Accuracy:

The accuracy is calculated as the proportion of correctly sorted data to all other data in the log. The level of accuracy is defined as,

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

Precision:

Precision is the depiction of the complete number of authentic samples that are properly taken into account throughout the classification process by using the full number of samples utilized in the classification procedure.

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

Sensitivity:

To determine the sensitivity value, just divide the total positives by the percentage of genuine positive predictions.

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

Specificity:

The number of predicted negative outcomes is precisely divided by the total number of negatives to calculate specificity.

$$Specificity\ = \frac{TN}{TN+FP} \qquad (4)$$

MCC:

By taking into account TP, TN, FN, and FP, the MCC emerges as a reliable metric to assess the efficacy of binary classifiers. MCC quantifies the extent of association between predicted and true labels.

$$MCC = \frac{(TP*TN)-(FP*FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} (5)$$

Four potential classification alternatives can be discerned from this particular model:

- A True Positive (TP) relates to a valid and suitably classified individual who belongs to the Positive category.
- True negative (TN) pertains to instances wherein the negative class is deemed authentic and properly classified as such.
- A True Positive (TP) relates to a valid and suitably classified individual who belongs to the Positive category.
- True negative (TN) pertains to instances wherein the negative class is deemed authentic and properly classified as such.

F-Measure:

The F-measure that merges precision and Sensitivity, can be expressed through the ensuing equation.

$$\mathrm{F-measure} = \frac{2 * \mathrm{TP}}{2 * TP + FP + FN} \qquad (6)$$

False-positive rate (FPR)

The FPR, otherwise referred to as the aftermath, denotes the ratio of erroneous positive identifications (FP) to the aggregate of veritable negative occurrences.

$$FPR = \frac{FP}{FP+TN} \qquad (7)$$

False-negative rate (FNR)

The FNR also known as the miss rate, represents the percentage of individuals with a confirmed positive medical condition notwithstanding the disease, its diagnostic test findings are negative.

$$FNR = \frac{FN}{FN+TP} \qquad (8)$$

4.2 Performance Analysis

Table 4.1 manifests the information on the overall performance analysis. As per the acquired outcomes, the accuracy of the proposed approach is 96.7%, which is better than MLP=72.6%, CNN=77.3%, RNN=91.2%, RBM=80.8%, RF=90.9%, CHOS=87.3%, PROPOSED=96.7%. The major cause for this improvement is owing to the utilization of deep learning models. The precision recorded by the proposed model is 93.7%, which is better than MLP=7.2%. A precision model & accuracy may be calculated using a variety of assessment indicators. The proposed model has been put into practice using Python, and the related outcomes have been recorded. Comparison of the resulting MLP, CNN, RNN, RBM, RF, CHOS, and proposed predicted by the previously suggested ensembled models with the newly given ensembled models, depicted in Figure below andTable 4.1.

***Table 4.1:** Overall Performance Analysis 4.1 of the proposed model*

	MLP	CNN	RNN	RBM	RF	CHOS	PROPOSED
Accuracy	0.726589	0.773456	0.912	0.8089	0.90998	0.873498	0.967809
Precision	0.603465	0.697545	0.813465	0.830347	0.884609	0.86007	0.943465
Sensitivity	0.72765	0.84765	0.69001	0.81601	0.80875	0.87076	0.93709

Specificity	0.707654	0.87709	0.807004	0.760004	0.809654	0.71951	0.9478
F-Measure	0.71654	0.7613	0.8220654	0.87237	0.79009	0.70823	0.902345
MCC	0.786433	0.6931	0.810063	0.791453	0.852897	0.706783	0.934567
NPV	0.654328	0.7123	0.81001	0.69544	0.87906	0.798009	0.92776
FPR	0.445388	0.22334	0.5001	0.21009	0.5076	0.3098	0.12789
FNR	0.347689	0.456787	0.13459	0.229	0.12369	0.45609	0.09987

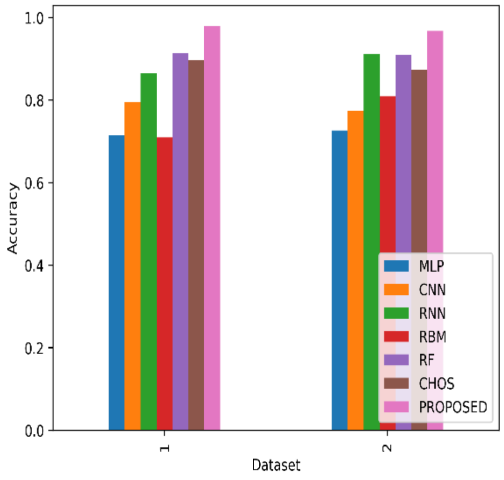


Figure 4.1. *Analysis of Accuracy.*

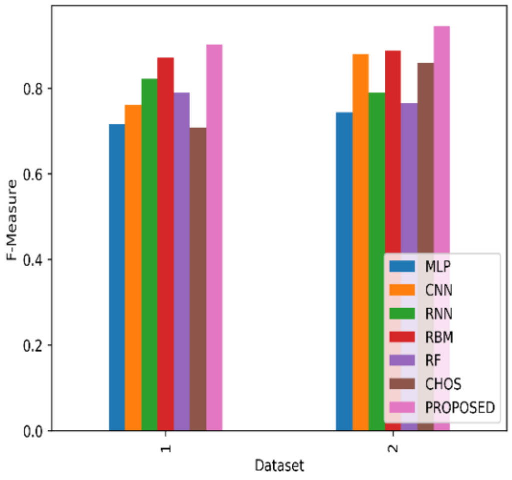


Figure 4.2. *Analysis of F-Measure.*

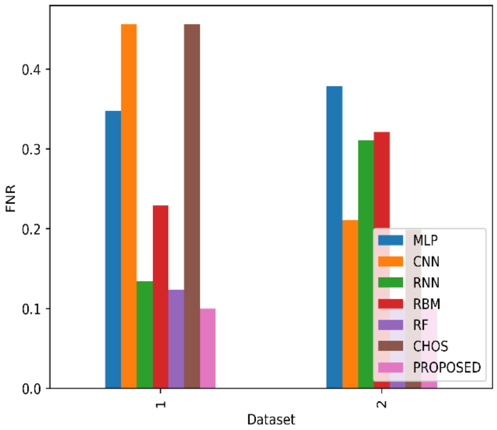


Figure 4.3. *Analysis of FNR.*

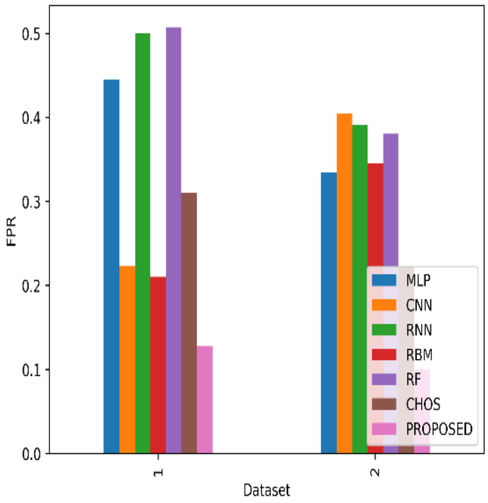


Figure 4.4. *Analysis of FPR.*

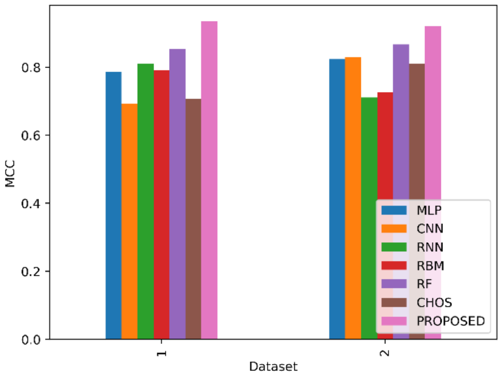


Figure 4.5. *Analysis of MCC.*

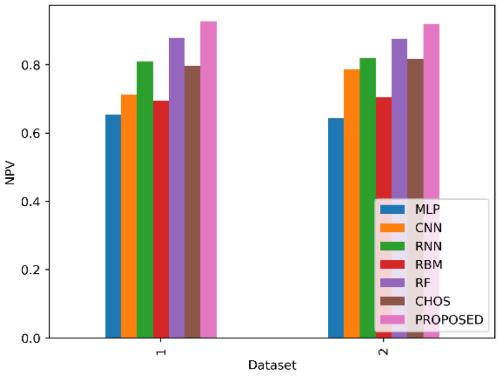


Figure 4.6. *Analysis of NPV.*

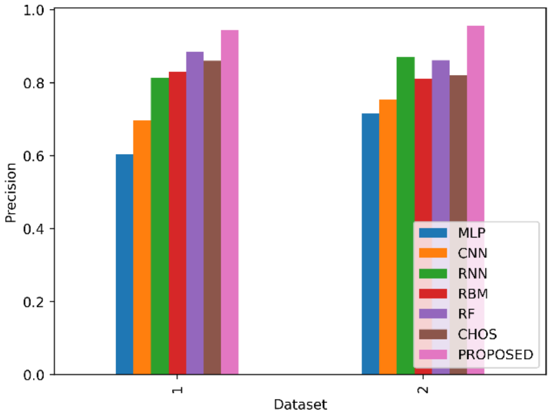


Figure 4.7. *Analysis of FPR.*

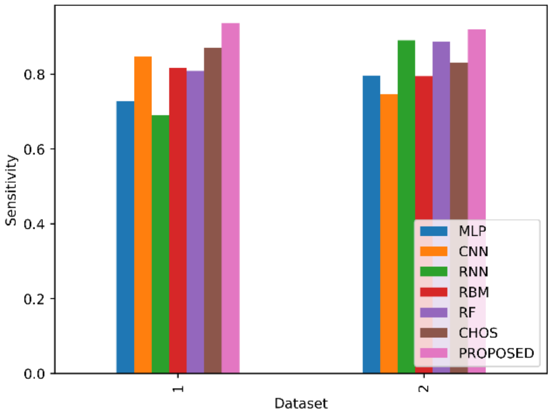


Figure 4.8. *Analysis of sensitivity.*

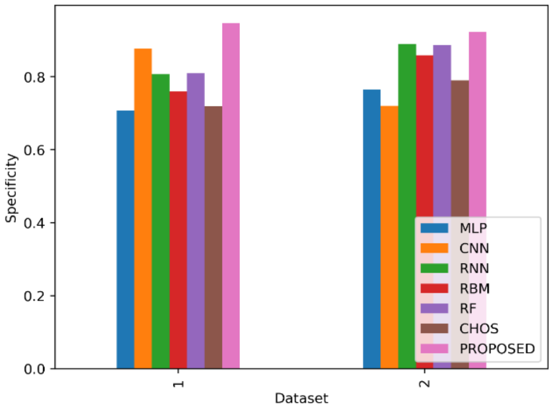


Figure 4.9.*Analysis of specificity.*

From Table 4.2, the proposed model has been validated over the existing models MLP, CNN, RNN, RBM, RF, CHOS, and the proposed in terms of accuracy, precision, MCC, F-measure, FPR, NPV, sensitivity, FNR, and specificity. As per the acquired outcomes, the accuracy of the proposed approach is 96.7%, which is better than MLP=72.6%, CNN=77.3%, RNN=91.2%, RBM=80.8%, RF=90.9%, CHOS=87.3%, PROPOSED=96.7%. The major cause for this improvement is owing to the utilization of deep learning models. The precision recorded by the proposed model is 94.3%, which is better than MLP=7.2%. A precision model & accuracy may be calculated using a variety of assessment indicators. The proposed model has been put into practice using Python, and the related outcomes have been recorded. Comparison of the resulting MLP, CNN, RNN, RBM, RF, CHOS, and proposed predicted by the previously suggested ensembled models with the newly given ensembled models, depicted in Figure below andTable 4.2.

Table 4.2: Overall Performance Analysis 4.2 of the proposed models

	MLP	**CNN**	**RNN**	**RBM**	**RF**	**CHOS**	**PROPOSED**
Accuracy	0.726589	0.773456	0.912	0.8089	0.90998	0.873498	0.967809
Precision	0.716523	0.753465	0.87043	0.810346	0.862009	0.82007	0.95678
Sensitivity	0.796783	0.7476	0.89012	0.7945	0.8877	0.83098	0.92009
Specificity	0.76534	0.720054	0.890211	0.859077	0.8875	0.79001	0.923456
F-Measure	0.74423	0.88004	0.790012	0.887653	0.765412	0.859003	0.94569
MCC	0.823456	0.830012	0.710912	0.726433	0.866434	0.810003	0.920008
NPV	0.64438	0.78743	0.819	0.704328	0.87546	0.816789	0.92065
FPR	0.334569	0.4053	0.39123	0.345689	0.380976	0.223	0.100089
FNR	0.37894	0.21076	0.31098	0.321	0.109324	0.198765	0.099876

Accuracy:

Accuracy is the issue faced by most works. To overcome this, the Proposed MLP, CNN, RBM, RNN, RF, CHOS, and PROPOSED are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest accuracy at 7.2%. The accuracy data obtained using the suggested model is 19.7% better than the CNN model. The accuracy data obtained using the suggested model is 5.2% better than the RNN model. The accuracy data obtained using the suggested model is 16.6% better than the RBM model. The accuracy data obtained using the suggested model is 6.25% better than the RF model. The accuracy data obtained using the suggested model is 9.3% better than the CHOS model. Thus, the resulting accuracy is compared to the other existing works.

Precision:

Precision is the foremost challenge encountered by a majority of works. To overcome this, the Proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest Precision at 7.1%. The Precision recorded by the proposed model is 92.1% better than the CNN model. The Precision recorded by the proposed model is 90.8% better than the RNN model. The Precision recorded by the proposed model is 91.5% better than the RBM model. The Precision recorded by the proposed model is 91.0% better than the RF model. The Precision recorded by the proposed model is 91.4% better than the CHOS models. Thus, the resulting precision is compared to the other existing works.

Sensitivity:

Sensitivity presents a notable obstacle for the majority of studies. To overcome this, the Proposed approaches MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest Sensitivity at 14.1%. The Sensitivity recorded by the proposed model is 16.3% better than the CNN model. The Sensitivity recorded by the proposed model is 3.2% better than the RNN model. The Sensitivity recorded by the proposed model is 88.2% better than the RBM. The Sensitivity recorded by the proposed model is 4.3% better than the RF model. The Sensitivity recorded by the proposed model is 9.7% better than the CHOS model. Thus, the resulting Sensitivity is compared to the other existing works.

Specificity:

The issue of specificity presents a noteworthy obstacle for a majority of literary works. To overcome this, the Proposed approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest Specificity at 7.6%. The Specificity recorded by the proposed model is 21.7% better than the CNN model. The Specificity recorded by the proposed model is 3.2% better than the RNN model. The Specificity recorded by the proposed model is 7.6% better than the RBM models. The Specificity recorded by the proposed model is 4.3% better than the RF models. The Specificity recorded by the proposed model is 14.1% better than the CHOS models. Thus, the resulting Specificity is compared to the other existing works.

F-Measure:

The F-measure presents a notable obstacle for the majority of studies. To overcome this, the Proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest F-measure at 7.4%. The F-measure obtained using the suggested model is 90.6% better than the CNN model. The F-measure obtained using the suggested model is 91.6% better than the RNN model. The F-measure obtained using the suggested model is 91.5% better than the RBM model. The F-measure obtained using the suggested model is 91.9% better than the RF model. The F-measure obtained using the suggested model is 91.0% better than the CHOS model. Thus, the resulting F-Measure is compared to the other existing works.

NPV:

NPV poses a significant challenge for a majority of works. To overcome this, the Proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest NPV at 6.4%. The NPV recorded by the proposed model is 15.2% better than the CNN model. The NPV recorded by the proposed model is 11.9% better than the RNN model. The NPV recorded by the proposed model is 23.9% better than the RBM. The NPV recorded by the proposed model is 4.3% better than the RF model. The NPV recorded by the proposed model is 11.9% better than the CHOS model. Thus, the resulting NPV is compared to the other existing works.

MCC:

MCC is the foremost challenge encountered by a majority of works. To overcome this, the Proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest MCC at 8.2%. The MCC that the suggested model recorded is 9.7% better than the CNN model. The MCC that the suggested model recorded is 22.8% better than the RNN model. The MCC that the suggested model recorded is 21.7% better than the RBM model. The FNR recorded by the proposed model is 6.52% better than the RF model. The MCC recorded by the proposed model is 11.9% better than the CHOS model. Thus, the resulting MCC is compared to the other existing works.

FPR:

The FPR presents a notable obstacle for the majority of studies. To overcome this, the proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest FPR at 3.3%. The FPR recorded by the proposed model is -3% better than the CNN model. The FPR recorded by the proposed model is -3.8 % better than the RNN model. The FPR recorded by the proposed model is -2.4% better than the RBM model. The FNR recorded by the proposed model is -2.8% better than the RF. The FNR recorded by the proposed model is 0.9% better than the CHOS models. Thus, the resulting FPR is compared to the other existing works.

FNR:

FNR is the issue faced by most works. To overcome this, the Proposed MLP, CNN, RNN, RBM, RF, CHOS, and PROPOSED approaches are developed in this research work. As per the acquired outcomes, the Proposed MLP has recorded the highest FNR at 3.7%. The FNR recorded by the proposed model is -1.3% better than the CNN model. The FNR recorded by the proposed model is -2.4% better than the RNN model. The FNR recorded by the proposed model is -2.5% better than the RBM model. The FNR recorded by the proposed model is -1.1% better than the RF model. The FNR recorded by the proposed model is -1.1% better than the CHOS model. Thus, the resulting FNR is compared to the other existing works.

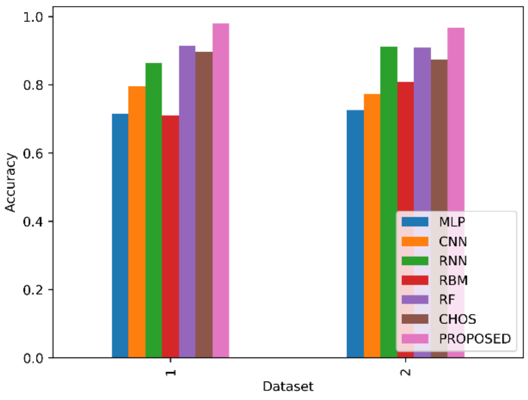


Figure 4.10. *Analysis of Accuracy*.

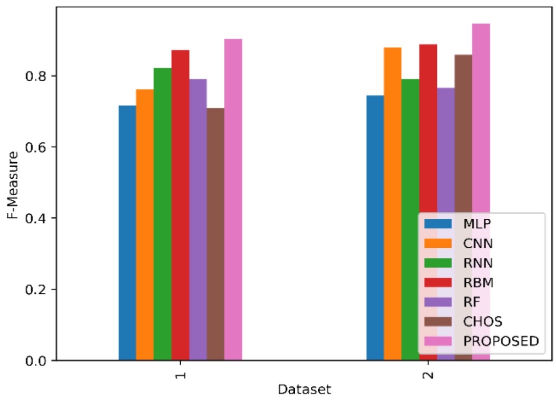
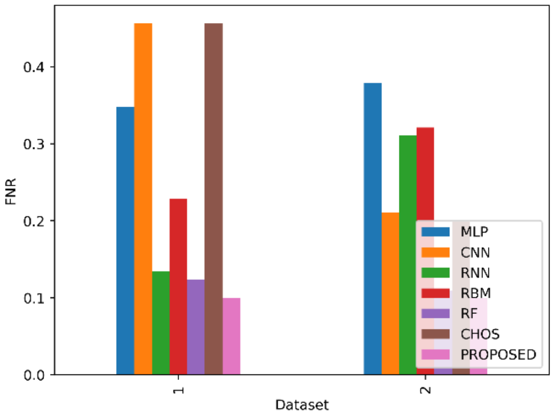


Figure 4.11. *Analysis of F-Measure.*



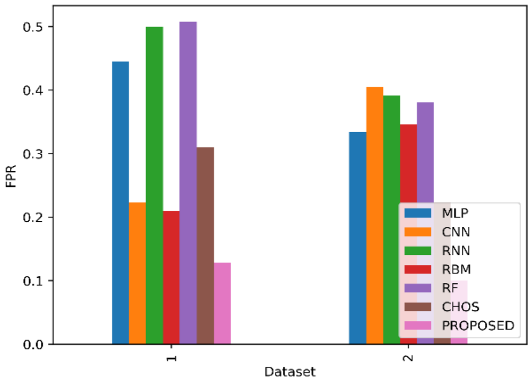
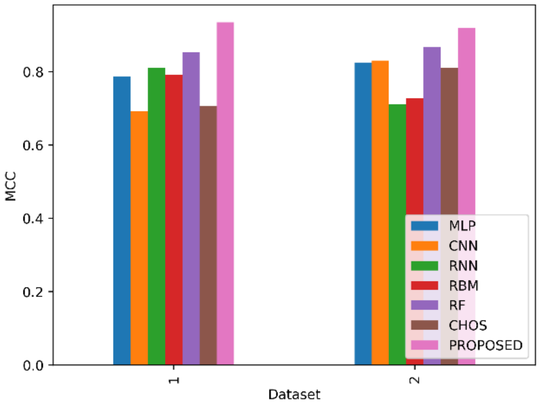


Figure 4.12. *Analysis of FNR.*

Figure 4.13. *Analysis of FPR.*



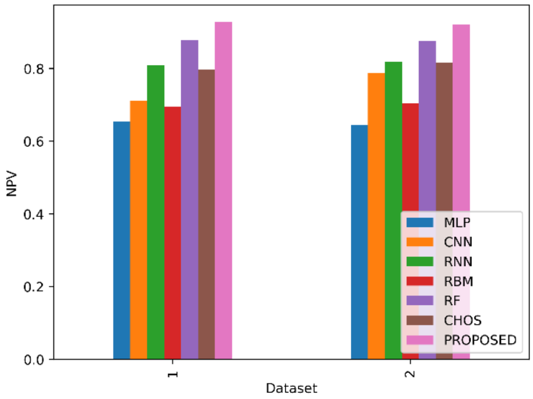


Figure 4.14. *Analysis of MCC*

Figure 4.15. *Analysis of NPV*

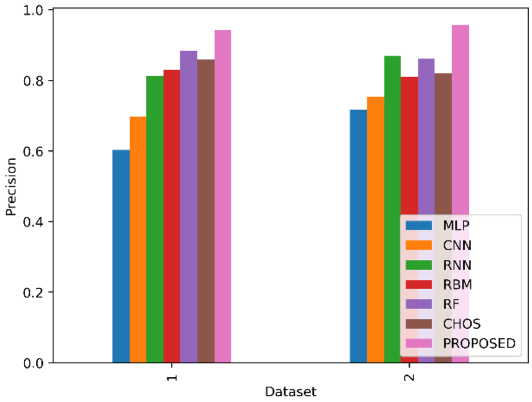


Figure 4.16. *Analysis of Precision*

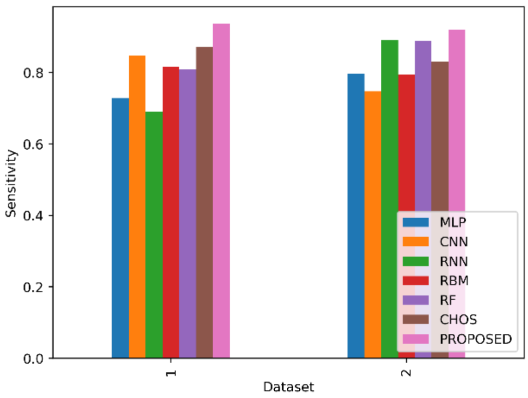


Figure 4.17. *Analysis of Sensitivity*

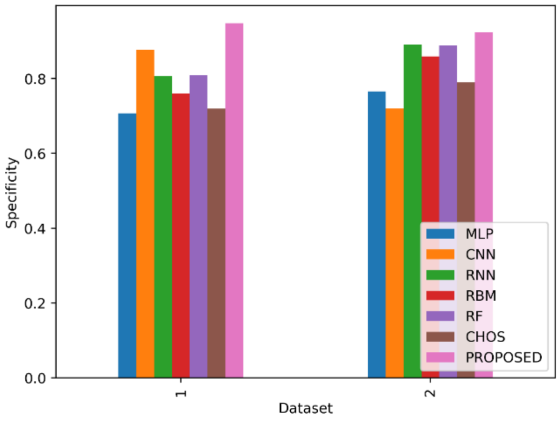


Figure 4.18*. Analysis of Specificity.*

5. CONCLUSION

The paper has developed a three tier deep-learning based model to detect the attacks in cloud system. The data were gathered by KDD Cup and CICIDS2017. The data were pre-processed via data cleaning (missing data removal) and Z-Score normalization. From the pre-processed data the features were extracted by central tendency, degree of dispersion and other features like (packet count, traffic volume, protocol distribution and entropy). And from the extracted features the relevant features were selected using the hybrid model which is a combination of BRA and CSA. Finally, the expected outcome was derived from the three-tier-deep-learning based attack detection model which includes CNN, RNN, MLP and MRBM. The proposed methodology was executed via the PYTHON platform.

REFERENCE

1. Hezavehi, S.M. and Rahmani, R., 2023. Interactive anomaly-based DDoS attack detection method in cloud computing environments using a third-party auditor. Journal of Parallel and Distributed Computing, 178, pp.82-99.
2. MM, G.A. and TF, M.R., 2022. An efficient SVM based DEHO classifier to detect DDoS attack in cloud computing environment. Computer Networks, 215, p.109138.
3. Wu, X., Jin, Z., Zhou, J. and Duan, C., 2023. Quantum Walks-based Classification Model with Resistance for Cloud Computing Attacks. Expert Systems with Applications, p.120894.
4. Maheswari, K.G., Siva, C. and Nalinipriya, G., 2023. Optimal cluster-based feature selection for intrusion detection system in web and cloud computing environment using hybrid teacher learning optimization enables deep recurrent neural network. Computer Communications, 202, pp.145-153.
5. Zhang, Y., Li, G., Duan, Q. and Wu, J., 2022. An interpretable intrusion detection method based on few-shot learning in cloud-ground interconnection. Physical Communication, 55, p.101931.
6. Mayuranathan, M., Saravanan, S.K., Muthusenthil, B. and Samydurai, A., 2022. An efficient optimal security system for intrusion detection in cloud computing environment using hybrid deep learning technique. Advances in Engineering Software, 173, p.103236.
7. Tian, Z., Luo, C., Qiu, J., Du, X. and Guizani, M., 2019. A distributed deep learning system for web attack detection on edge devices. IEEE Transactions on Industrial Informatics, 16(3), pp.1963-1971.
8. Agrawal, N. and Tapaswi, S., 2019. Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges. IEEE Communications Surveys & Tutorials, 21(4), pp.3769-3795.
9. Saxena, R. and Dey, S., 2020. DDoS attack prevention using collaborative approach for cloud computing. Cluster Computing, 23, pp.1329-1344.

10. Aydın, H., Orman, Z. and Aydın, M.A., 2022. A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defense system design in public cloud network environment. Computers & Security, 118, p.102725.
11. Dhanapal, A. and Nithyanandam, P., 2019. The slow HTTP distributed denial of service attack detection in cloud. Scalable Computing: Practice and Experience, 20(2), pp.285-298.
12. Divyasree, I.R., Selvamani, K. and Riasudheen, H., 2020. Detection of Colluded Black-hole and Grey-hole attacks in Cloud Computing. CoRR.
13. Alassafi, M.O., Alharthi, A., Walters, R.J. and Wills, G.B., 2017. A framework for critical security factors that influence the decision of cloud adoption by Saudi government agencies. Telematics and Informatics, 34(7), pp.996-1010.
14. Rao, P.R. and Sucharita, D.V., 2019. A framework to automate cloud-based service attacks detection and prevention. International Journal of Advanced Computer Science and Applications, 10(2), pp.241-250.
15. Abdullayeva, F.J., 2022. Distributed denial of service attack detection in E-government cloud via data clustering. Array, 15, p.100229.
16. Agarwal, A., Prasad, A., Rustogi, R. and Mishra, S., 2021. Detection and mitigation of fraudulent resource consumption attacks in cloud using deep learning approach. Journal of Information Security and Applications, 56, p.102672.
17. Kushwah, G.S. and Ranga, V., 2020. Voting extreme learning machine based distributed denial of service attack detection in cloud computing. Journal of Information Security and Applications, 53, p.102532.
18. Rani, D.R. and Geethakumari, G., 2020. Secure data transmission and detection of anti-forensic attacks in cloud environment using MECC and DLMNN. Computer Communications, 150, pp.799-810.
19. Pasha, M.J., Rao, K.P., MallaReddy, A. and Bande, V., 2023. LRDADF: An AI enabled framework for detecting low-rate DDoS attacks in cloud computing environments. Measurement: Sensors, p.100828.
20. Samunnisa, K., Kumar, G.S.V. and Madhavi, K., 2023. Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods. Measurement: Sensors, 25, p.100612.

21. Abdullayeva, F., 2023. Cyber resilience and cyber security issues of intelligent cloud computing systems. Results in Control and Optimization, p.100268.

22. Kushwah, G.S. and Ranga, V., 2021. Optimized extreme learning machine for detecting DDoS attacks in cloud computing. Computers & Security, 105, p.102260.

23. Shah, S.Q.A., Khan, F.Z. and Ahmad, M., 2021. The impact and mitigation of ICMP based economic denial of sustainability attack in cloud computing environment using software defined network. Computer Networks, 187, p.107825.