Turkish Journal of Computer and Mathematics Education *Vol.12 No.6 (2021), 2468-2482*
Research Article

An Efficient Intrusion Detection System Using Improved Bias Based Convolutional Neural Network Classifier

Mathiyalagan R[1], Pamela Vinitha Eric[2]

[1]Research Scholar, Visvesvaraya Technological University, Belagavi, India
E-mail: mathi.prajval@gmail.com

[2]Professor, New Horizon College of Engineering, Bengaluru, India
E-mail: pamela.vinitha@gmail.com

Abstract: Today's modern society has faced many challenges due to the rapid digitization and growing number of hackers, which makes the networking-based systems to become a target place for intruders. The attacks may allure the users, and it compromised the whole system and makes the security the biggest challenge. In this regard, the best way to combat the issues is by exploring new ways to defend the network against threats. More recently, Intrusion Detection Systems (IDS) is a key enabling technology in maintaining the novel network security. Indeed, some existing systems utilize Improved Relevance Vector Machine (IVRM) classifier for performing intrusion detection in network-based systems. In this work, feature selection is done by using Gaussian Firefly Algorithm and Improved Relevance Vector Machine (IRVM) based classification is performed according to the selected features. However, for large-scale intrusion dataset, the intrusion detection is not robust; hence, it leads to high attack rates. The proposed system designed an Improved Bias based Convolutional Neural Network (ICNN) for high attack intrusion detection. For embracing high-security factors and enhanced protection, the proposed system performs three phases, such as preprocessing, feature selection, and classification. The first phase employs the KDD dataset and Kalman filtering method followed by feature selection utilizes Inertia Weight based Dragonfly Algorithm (IWDA) and finally identified the intrusion attacks using Improved Bias based Convolutional Neural Network (IBCNN) classifier. In this work, a novel model performed with the KDD dataset. The suggested method evaluated in terms of accuracy, f-measure, recall, and precision for examining performance compared with existing systems.

Keywords: Intrusion Detection System (IDS), Improved Bias based Convolutional Neural Network (IBCNN), feature selection and Dragonfly Algorithm (DA)

1.Introduction

Nowadays, due to the growing advent of digital technology and web applications, computer-related crimes are becoming more pervasive and thus becoming a potential challenge [1]. New threats posed by hackers and cybercriminals are in a growing number, which leads the attackers to invade the whole network system. Intrusion in network-based systems causes malicious attacks, thereby leads to an operational fault in some cases. The attackers exploit intrusion as a pivotal factor for compromising integrity, availability, and confidentiality [2-3]. Henceforth, IDS are adopting as a defense mechanism for providing security and protection, and in turn, it performs anomaly and attack detection in the network. More recent researches revealed that data mining solutions for IDS are the best defending measure in detecting the attack patterns [4-7]. This assists in continuous monitoring of the actions that performed in the network. It is designed for impeding the unauthorized and malevolent nature if found. This system combats against Denial of Service attack (DoS), Users-to-Root attack (U2R), Remote-to-Local attack (R2L), and Probing attack (PROBE).

As per the recent survey, the KDD data set is a bench-marking technique widely accepted in IDS researches [8-9]. Numerous researches have been undergone for enhancing the present-day IDS strategies, in which more work is done on the training and testing the models used for detection. It is mainly accomplished for quality improvement in order to perform intrusion detection in offline. The detrimental factors of IDS are the factors that

compromise integrity, availability, and accountability. Nevertheless, the recent advances in IDS outperform the attacks posed by a cyber-hacker by performing novel feature selection and classification.

For performing the elimination of duplicate and extraneous traits from the datasets, feature selection is a widely accepted technique in many systems. It selects the most optimal subsets from the massive data and hence provides the enhanced characterization of patterns. More often, filter and wrapper methods are the techniques that are performed in feature selection [10]. In filter methods, the independent measure is chosen as a criterion in order to estimate the association of the features. Typically, information, distance, and consistency are considered as independent measures. Contrastingly, in wrapper methods, the value of the features has been evaluated by employing learning algorithms. Due to the intensive nature of the data, wrapper methods are the most acceptable feature selection method as it has the capability of dealing with massive data sets. Consecutively for dimensionality reduction, metaheuristic algorithms such as Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Genetic Programming (GP) are used in computationally intensive applications. Metaheuristics are now accepted as the preferable measure since they perform better by utilizing a reduced number of computational resources.

At present, the data mining techniques such as Decision Tree, Naïve Bayes (NB), Neural network, and Support Vector Machines (SVM)[11] are employed for modeling classification in IDS. Though the data mining techniques often used in some IDS, issues such as false-positive rates and data redundancy still been a challenge. It also possesses a dilemma in detection rates and thus yields apocryphal values. The critical factor of IDS is the adaptability in high-speed networks while handling large scales of data in a reduced time. Since the IDS are not robust, and the difficulty persists in data-intensive applications, which paves the way for the intruders to enter in.

Hence the proposed system investigates preprocessing and feature selection so as to enhance the accuracy rates of classification performed in IDS. The proposed system is classified into three phases. a) Preprocessing b) feature selection and c) classification. Also, for the better replacement of missing values Kalman filtering algorithm is applied in KDD dataset. Followed by preprocessing, Inertia Weight based Dragonfly Algorithm (IWDA) is used for performing feature selection. In this phase, accuracy is obtained by generating objective function for getting optimal solutions. Finally, for performing better classification, Improved Bias based Convolutional Neural Network (IBCNN) is applied.

2.Literature Review

Several researches have been done in investigating IDS in terms of security and privacy. Since the intrusion in some systems poses serious implications such as tampering of data and acquiring secret information across the trusted network. Some of the techniques and methodologies suggested to thwart from such issues are mentioned in this section.

In [12], Intelligent Water Drops (IWD)-based feature selection is proposed for IDS for maintaining accuracy of the system. It focuses on optimizing the feature selection by employing novel classifiers. This method generally uses a bio inspired algorithm that are combined with SVM classifier for the assessment of the selected traits. Consequently, the optimization is attained at a better level by performing several numbers of rounds. The authors have tested the values from KD CUP'99 dataset and the performance are analyzed with existing systems. The obtained results shown that the suggested method outperforms the recent models in terms of intended detection rate, false alarm rate and accuracy.

Followed by [12], Kuang et al. [14] put forth and IDS model which combines Chaotic Particle Swarm Optimization (ICPSO) and Kernel Principal Component Analysis (KPCA) to ameliorate the performance of support vector machines. In this method, KPCA performed for reducing the training time and dimension reduction. On the other hand, ICPSO used for the optimization of tube size, kernel parameters, and punishment factors. The overall system is intended for chaos optimization and premature processing. It further divides the training dataset into ten samples, which further yields a 96% detection rate and 1% of false alarm rate. Since this method offers some diversified features, the challenges due to the handling of seamless data sets remain challenging.

To overcome the disadvantages, Ingre et al. [15] examine the NSL-KDD with the Decision Tree-Based Intrusion Detection System. It models a Correlation Feature Selection (CFS) subset evaluation for performing feature selection. Further, it enhances the IDS performance by feature selection. To analyze the performance of the suggested venture, feature selection is carried out before and after the classification. Here two types of classification are performed, namely five class classification for normal and types of attack and binary class classification for normal and attack. The values that are obtained from the proposed method is further analyzed for high DR and accuracy by comparing it with the existing techniques. The simulation has shown that the binary class classification outperforms five-class classification. Though the system performs high-class classification, it possesses disadvantages due to the increasing number of new threats.

To address the above challenges, Zhao et al. [16] proposed an effectual 2 stage method to distinguish the intrusions in the network. It propagates the pool of solutions that are non-dominating and optimal; in turn, the ensembles are used for detecting the intrusions effectively. Additionally, it creates Pareto optimal solutions to express the chromosome structure at stage one with pareto front. Likewise, in the second stage to obtain auxiliary ensembles, another kind of nearness to the Pareto front is made. Consecutively, the voting approach is equipped for computing the prediction ensembles from self-predictions. Moreover, finally, the validation is done by using the benchmark NSL-KDD dataset. The obtained values from the simulation have clearly shown that the suggested method shows better performance when compared to the other existing systems. They have formulated the classification measure to handle the generation of optimal solutions to improve the detection accuracy while acting upon majority and minority threats. The simulation results have shown 97% detection accuracy and a 2% false-positive rate for KDD.

Though the 2 stage method paves the way for high prediction, the methods to detect the intrusions in the network-based systems is still sparse. Hence, Chandrasekhar et al. [17] have designed a model comprises of 4 steps in which the k-means clustering serves for generating diversified training subsets which depend on the acquired subset. Here, several neuro-fuzzy data models are trained to get optimal datasets. Followed by the clustering, vectors of the respective values have been acquired using SVM classification. At last, to detect the intrusions, radial SVM is adapted. The experimental results have shown that the proposed system possesses better applicability and ability when compared to BP, multiclass SVM, and decision trees.

Later, Kim et al. [18] utilized KDD Cup 99 datasets to design AI-based IDS exploiting Deep Neural Networks. This model assists the system in combating the growing network attacks. Initially, data preprocessing is done using data transformation and normalization for the input values obtained from the DNN model. Then the DNN algorithm applied to get the learning model through preprocessing and the KDD Cup 99 dataset used for the verification, respectively. Lastly, simulation using the latest models performed to analyze the detection and false alarm rate, and hence the detection efficacy ascertained. Thus, it paved the way for a better understanding of the novel way of performing intrusion detection.

Followed by Kim et al., in [19], a unique way of optimization of kernel parameters achieved through the combination of Principal Component Analysis and Support Vector Machines. It formally diminishes the training and testing time. Hence the accuracy is improved while performing identification of intrusions. It further tested on the KDD dataset. Moreover, by considering the minority attacks(U2R, R2L), the KDD datasets are further split into training and testing to predict the occurrence of future attacks.

In[20], a novel method for network-based IDS proposed by Belouch in which 2 stage classifier and a RepTree algorithm suggested. This method initially takes the input from UNSW-NB15 and the NSL-KDD data set. In the initial stage, the model splits the network traffic into TCP, UDP, and the rest of the other protocols. Further, it classifies the data for standard and anomaly. Consecutively, in the second stage, accurate intervention is chosen by employing a multiclass algorithm. It effectively classifies the anomaly detected in the initial phase. The features are further reduced to less than 20 features as per the design of the protocol by employing the feature selection techniques. The simulation results have revealed that the detection accuracy was estimated from 88%, 95%, and 89% 85% for a complete UNSW-NB15 and NSL-KDD dataset.

Later Ambusaidi et al. [21] proposed an enhanced method for selecting the optimal features required for classification analytically. The author has presented an information-based algorithm to handle linear and non-linear data features. In this way of feature selection, the effectiveness then evaluated for network-based IDS. Furthermore, Least Square Support Vector Machine based Intrusion Detection System is modeled by utilizing the features that are obtained through selection algorithms. Additionally, the performance of the system was analyzed for various parameters such as error rate and time efficiency. The simulation is done using the data taken from KDD Cup 99, NSL- KDD, and Kyoto 2006+. The results obtained have shown better performance compared to existing systems.

In [22], the authors have presented a novel approach for the effective classification of intrusion attacks. They have formulated Alternating Decision Trees (ADT) to the data obtained from the intrusions. Also, it has been extended to the further classification of the several types of attacks. Specifically, ADT is a fascinating approach that utilizes the decision trees intended for binary classification problems. Also, it is widely accepted as a supervised boosting algorithm. The authors have utilized the NSL-KDD data sets for further analysis. They have obtained an accuracy ranges from 97.15 to 97.61% in the case of DOS, Probe, U2R, and R2L.

TABLE 1.COMPARITIVE ANALYSIS OF INTRUSION DETECTION METHODS

S.no	Authors name	Methods	Merits	Demerits
1.	Acharya and Singh (2017)	Intelligent Water Drops (IWD)	It attains higher detection rateand low false alarm rate	It has issue with trapped into local minima
2.	Kuang et al. (2015)	Support Vector Machine (SVM)	It achieves 96% of detection rate	The SVM takes a long training time on large datasets.
3.	Ingre et al (2017)	Correlation Feature Selection (CFS) and Decision Tree (DT)	Simple to understand the decision tree	Decision tree training is relatively expensive as complexity
4.	Zhao et al (2019)	Artificial Intelligence based Ensemble Approach	It attains detection accuracy of 97%	It increase the storage space and computational time dueto the presence of a vast number of base classifiers in the ensemble learning
5.	Chandrasekhar and Raghuveer (2013)	k-means clustering and Support Vector Machine (SVM)	The SVM can efficiently handle non-linear data	Selecting an appropriate Kernel function is difficult
6.	Kim et al (2017)	Deep Neural Network (DNN)	It can able to solve complex problems very well	Expensive and intensive training required
7.	Thaseen et al (2014)	Principal Component Analysis (PCA) and Support Vector Machine (SVM)	It minimizes the training and testing overhead time	The PCA may miss some information as compared to the original list of features
8.	Belouch et al (2017)	Reduced Error Pruning Tree (REPTree) algorithm	It attains higher speed of detection	Need to improve the accuracy
9.	Ambusaidi et al (2016)	Least Square Support Vector Machine based Intrusion Detection System (LSSVM-IDS	Higher accuracy and lower computational cost	It suffers from the curse of dimensionality.

10.	Jabbar,et al (2016)	Alternating Decision Trees (ADT)	Minimum false acceptance rate	It often relatively inaccurate.

3. Proposed methodology

In this proposed research work, Improved Bias based Convolutional Neural Network is introduced for IDS. The designed system comprises of

(i) Preprocessing

(ii) Feature Selection

(iii) Classification

Figure 1 depicts the system architecture of the proposed system.

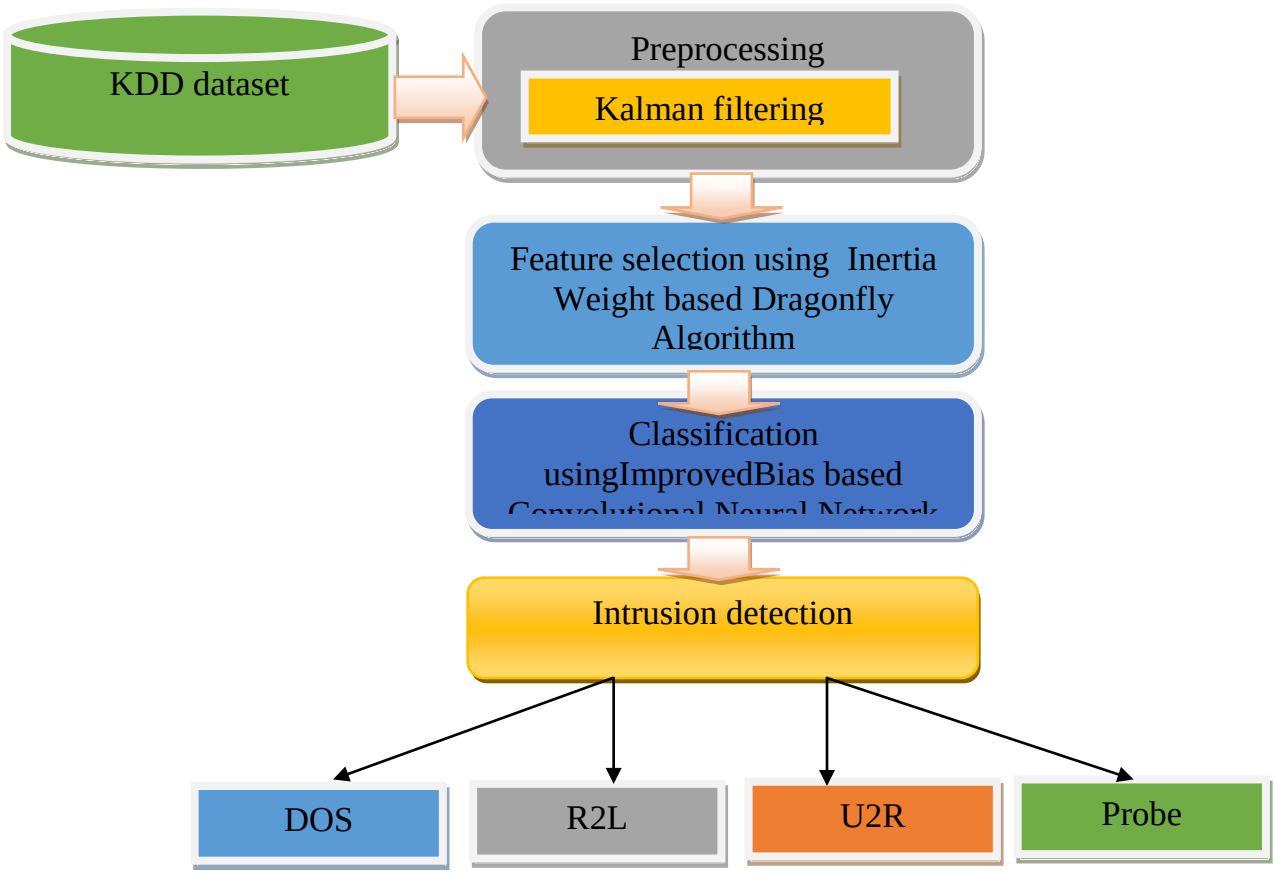

Figure 1: System architecture of the proposed system

3.1 Input datasets

The KDD dataset is used for exploring testing and training the input samples. This dataset is provided by UCIKDDArchive(1999). Generally, KDD is widely accepted dataset for its trustworthiness and benchmarking. This assist in assessing the intrusion detection systems. MATLAB simulation is used here for evaluating the KDD

dataset. The characteristics of our proposed setup are: (i) 41 features (ii) 9 absolute features (iii) 32 consecutive extracts. Additionally, the attacks are categorized into 4 key classes. (i) DoS attacks (ii) R2L (iii) U2R (iv) Probing.

3.2 Preprocessing

In our proposed system Kalman filtering is used for data preprocessing to remove the noisy data for further process. The KDD dataset usually consists of numerals and categorical values. In some cases, the tuple values may get missed due to some external values. Hence the proposed Kalman filtering mainly used for handling and processing the missing values to avoid ambiguous results.

Usually deriving the equation and calculating the covariance error are the means of discovering the missing values in the KDD dataset. Thus, the unambiguity can be reduced at a considerable rate and thereby classification accuracy can be ascertained.Also, Kalman filtering paves way for finding out and projecting the lost data.Moreover, prediction is done for the data while updating the coefficient approximation of the Kalmanfilterby taking lose of data into account. Meanwhile, the mean value of the data is also getting updated with the lost data.

Algorithm:

1. Prediction

$$A_{s|s-1} = FA_{s-1|s-1}$$
$$P_{s|s-1} = FP_{s-1|s-1}F^T + \mathrm{Q}$$

If the data is missing

Update filter

$$\begin{cases} K_s = P_{s|s-1}H^T(HP_{s|s-1}H^T + R)^{-1} \\ A_{s|s} = A_{s|s-1} + K_s(B_s - HA_{S|S-1}) \\ P_{s|s} = (1 - K_sH)P_{S|S-1} \end{cases}$$

2. If the data is existing

$$\begin{cases} K_s = P_{s|s-1}H^T(HP_{s|s-1}H^T + R)^{-1} \\ A_{s|s} = A_{s|s-1} + K_s(B_s - HA_{S|S-1}) \\ P_{s|s} = (1 - K_sH)P_{S|S-1} \end{cases}$$

3. Filter

$$\begin{cases} L_s = P_{s|s}\,F^T p_{s+1|s}^{p-1} \\ A_{s|s} = A_{s|s} + L_s(A_{s+1|T} - A_{S+1|S}) \\ P_{s|T} = P_{s|s} + L_s(P_{s+1|T} - P_{s+1|s})L_s^T \end{cases}$$

4. The formula given below calculates

Covariance matrix

$$\begin{cases} A_{s|j} = E(A_s|A_j) \\ P_{s|j} = E[(A_s - A_{s|j})^T] \end{cases}$$

Where,

A_s denotes the value of state vector (For ex. Heading, place of the origin and velocity)

F expresses the state transition matrix which consists of parameter outcomes of every system

P_s is the covariance matrix

H^T denotes the transformation matrix which intends to map the state vector values onto the domain of measuring, and

K_t is the Kalman filter

The finest estimated location of the train is determined by the combination of prediction and measurement data. Furthermore, the missing values are filled by the foreseen values that are present in the KDD dataset. Thus, the proposed system offers a finest recovery mechanism for the intrusion data by adopting the proposed Kalman Filter techniques.

3.3 Feature selection

In this section, feature selection is performed by exploiting the data set which gets preprocessed in the initial stage.Subsequently, Inertia Weight based Dragonfly Algorithm (IWDA) algorithm for accomplishing feature selection. The attribute selection or feature selection then facilitating the swarm behavior of dragon fly individuals by employing the Dragonfly Algorithm (DA). Since, DA is a metaheuristic optimization algorithm, it initiates the optimization with the clique of solutions that are obtained randomly. This often resembles most of the SI based optimization algorithms. According to that, the DA initialize the process by creating a random number of solutions for a given optimal problem. More specifically, the performance of DA is influenced by the dragon fly individuals(M). The global optima are getting measured by increasing the number of populations, iteration calculation time and the whole problem. Then the objective function has to be calculated for all dragon fly values. The following considerations are, for instance, the number of features is replaced as dragon fly and objective function is obtained from classification accuracy. Furthermore, the Improved Bias based Convolutional Neural Network (IBCNN) determines the classification accuracy. The position values are updated for each iteration with the step position vector of each and every feature followed by plotting the positions in upper and lower boundaries. The step vector is provoked by the following factors such as separation, arrangement, nourishment allurement and predator diversion. The simulation is performed for analyzing the swarming function of the dragon fly and the factors stated above are modeled as below [22].

The value of separation motion can be obtained by the equation (1)

$$S_{(i,t)}=-\sum_{j=1}^{N} X_{(i,t)} - X_{(j,t)} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad 1$$

Where$X_{(i,t)}$ is the location of ith feature plotted for every ith iteration ,$X_{(j,t)}$denotes the location of jthfeature that are obtained for very tth iteration; N is the number of adjoining features ; and $S_{(i,t)}$is the separation motion of the feature i for the iteration t. Furthermore, the alignment function is expressed below.

$$A_{(i,t)} = \frac{\sum_{j=1}^{N} V_{(j,t)}}{N} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad 2$$

Here$V_{(j,t)}$) is the velocity of the neighboring feature j for the iteration t followed by $A_{(i,t)}$which denotes the alignment motion of the featureifortheiteration t. Additionally, cohesion motion is expressed as

$$C_{(i,t)} = \frac{\sum_{j=1}^{N} X_{(j,t)}}{N} - X_{(i,t)} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad 3$$

$C_{(i,t)}$ is the cohesion motion for the feature i for the iteration t.

Consecutively, the food attraction motion is expressed as

$$F_{(i,t)}=X_{(foot,t)}-X_{(i,t)}\ldots\ldots\ldots\ldots\ldots\ 4$$

Where $X_{(foot,t)}$is the location of the food for every iteration t followed by $F_{(i,t)}$ is the attraction function of the food motion. Finally, the predator distraction value is obtained by

$$E_{(i,t)}=X_{(enemy,t)}-X_{(i,t)}(5)$$

$X_{(enemy,t)}$ is the exact location of the predator in the iteration t; and $E_{(i,t)}$ is the difference in the motion of the predator for the iteration t and the feature i. According to the observations, the predators yields the worst objective function and the combination of aforesaid factors will generates the predicted values of corrective patterns of feature in individual directions. The step vector is formulated as:

$$\Delta X_{(i,t+1)}=(\text{s}\times S_{(i,t)}+a\times A_{(i,t)}+c\times C_{(i,t)}+\text{f}\times F_{(i,t)}+\text{e}\times E_{(i,t)})+\omega\times\Delta X_{(i,t)}\ (6)$$

In equation 6 ,s is the counterweight of separation followed by a which denotes the load of the alignment and c which denotes the counterweight of cohesion. The food allurement is denoted as f and e is the difference in the predator weight. Moreover, w is the weight of inertia.

In the proposed system time varying inertia is expresses as equation 7

$$\text{Inertia weight }\omega=(\omega_1-\omega_2)\left(\frac{Iter_{max}-Iter}{Iter}\right)+\omega_2(7)$$

In which $\omega_1$is starting inertia weight and ω_2 are the end value of the weight of inertia.Iter denotes the running iteration and the $Iter_{max}$ is the utmost value of acceptable iterations. Additionally, the accurate and final position vectors are formulated as:

$$X_{(i,t+1)}=X_{(i,t)}+\ \Delta X_{(i,t+1)}(8)$$

The equation (9) expresses the positions of the features updated under the circumstances like no detection of vicinity.

$$X_{(i,t+1)}=X_{(i,t)}+\text{Levy (d)}\times X_{(i,t)}(9)$$

Where d denotes the number of decision variablesandLévy(d) is the Lévy flight function. Then the position updating is kept on continuing until the termination criteria is satisfied. The algorithm 1 depicts the overall flow of the proposed system

Algorithm 1Inertia Weight based Dragonfly Algorithm (IWDA)

1. Start
2. Set the initial population size as (M)
3. Initialize the counter value as 1 and set it as t
4. Begin and declare the no of features as Xi and i can take 1,2,…M
5. Check for the criteria of the end condition
6. Determine the accuracy of the classification of entire features
7. Mark the recent values for source and attacker
8. Ascertain the values of ω, s, a, c, f, and e
9. Compute the values of S, A, C, F, and E
10. Figure out the radius values of neighbors
11. Check if the feature has a single neighbor

12. Then formulate equation 6 and mark the value of velocity vector
13. And mark the very last values of the position vector by eqn 9
14. else
15. Take the value of position vector by applyingLévy flight function
16. End loop
17. Arrange the features from finest tothe least score and calculate the final best
18. Repeat and depict the results
19. End thewhile loop

3.4 Classification

After the feature selection is performed, the obtained margins are tending to posed for the classification by using the proposed Improved Bias based Convolutional Neural Network (IBCNN). One such distinct deep network is the Convolutional Neural Network which handles the umpteen hidden layers intended for convolution and sub sampling so as to get the ascending values of features that are obtained from the input dataset.

More specifically CNN comprising of 3 layers: (i) convolutional layer (ii) subsampling layer (iii) abundant connection layer[23]. The CNN mainly includes the input layer to initialize the input , output layer to get the output and hidden layer to get accuracy as shown in figure 2. The proposed Improved Bias based Convolutional Neural Network (IBCNN) efficiently classifies the maximum likelihood of attack patterns and extracts the relevant vectors. Furthermore, the values that poses biasing are getting optimized to yield the better values.

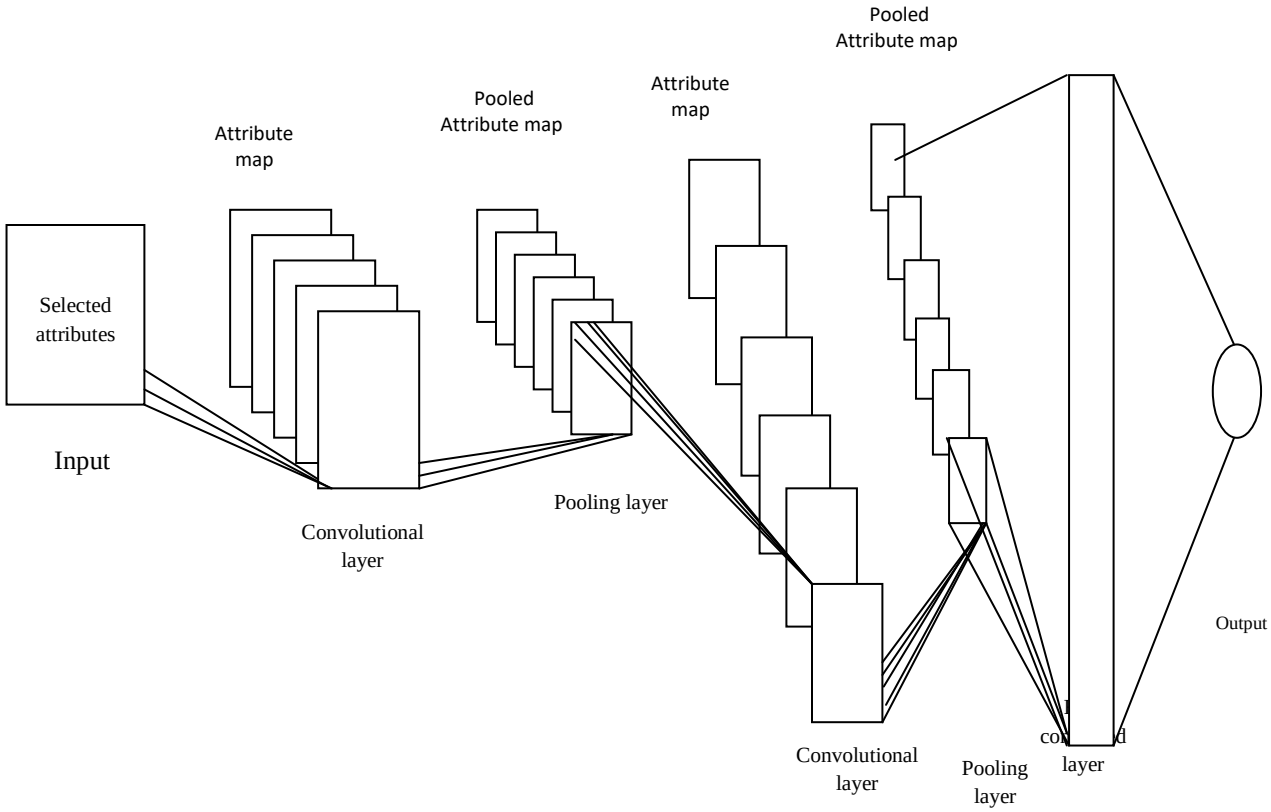

Figure 2: Convolutional Neural Network

Convolution layer

In the convolutional layer, the chosen optimal features are assumed as the input followed by the kernel convolutions. Here ever block of the matrix is convoluted without dependent on the adjacent attributes to generate

the output. The final outcome of the convolution is further used for the generation of the output. More often, the kernel is expressed as the filter and the corresponding output is obtained by the kernels and the attribute maps of the size i*i.

The proposed CNN can accomplish umpteen number of hidden layers and the input, output of the succeeding layer are the attributes as per concerned. Also, it contains several number of n filters for every convolutional layer. Further the filters are getting convoluted with the obtained inputs and the maps(n*) that are obtained from the prior process is same as the no.of filters that are formulated in the operation.

$C_j^{(l)}$ is the generated output of the layer l that is formulated below.

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} \quad (10)$$

The kernel generates attribute map. After the convolution layer, the activation function can be applied for nonlinear transformation of the outputs of the convolutional layer:

$$Y_i^{(l)} = Y(C_i^{(l)}) \quad (11)$$

$Y_i^{(l)}$denotes the activation function and $C_i^{(l)}$is considered as the received input.

The proposed system yields the value RELUs that can be expressed as $Y_i^{(l)} = \max(0, Y_i^{(l)})$. The function stated is widely used in the most advanced deep learning methods. It mainly helps in minimizing the synergy and multidimensional results. In the cases of receiving the input as negative, the RELU directly replaces the value of the output as 0. And, it yields the positive value if the input values received are same. The important feature of the activation function is its rapid training feature in the cases of error derivative and it comes to the lower value and gets saturated leads to the vanishing of weighted update and it is termed as vanishing gradient problem.

Sub sampling Layer

The convolution layer is succeeded by the sub sampling layer. The main function of the sub sampling layer is the dimensionality-reduction that are spatially distributed in which the attribute maps are generated by the preceding convolutional layer. To attain the values the masking of b*b is taken and the sub sampling is performed for the mask and maps

$$X_j^l = f(\beta_j^l \text{down}(X_j^{l-1}) + b_j^l) \quad (12)$$

Where, down (·) speaks to a sub-examining capacity. Regularly, this capacity will total over each particular n-by-n qualities in the information dataset with the goal that the yield is n-times littler along both spatial measurements. Each yield map is given its own multiplicative inclination β and an added substance predisposition b.

In this work ideal estimation of the b are refreshed dependent on the general mean estimation of predisposition of the quality guide. The predisposition mean is characterized as the aggregate of inclination estimations of the considerable number of properties isolated by the complete number of predisposition estimation of qualities.Bias mean $= \frac{\sum_{i=1}^{N} bias_i}{N}$ (13)

Where,

$N -$ Total number of attributes

$bias_i$ -Bias value of the attributes

Full Connection

The output layer uses Softmax activation function:

$$Y_i^{(l)} = f(z_i^{(l)}), \text{ where } z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_H y_i^{(l-1)} \quad (14)$$

where w_H are the weight value of the attributes that should be tuned by the complete fully connected layer in order to form the representation of each class and f is the transfer function which represents the nonlinearity. The proposed system classifies the input attributes into fourtypes of attack such as DOS, R2L, U2R and probe.

4. Experimental results

KDD dataset is used for exploring testing and training the input samples. This dataset is provided by UCIKDD Archive (1999). Generally, KDD is widely accepted dataset for its trustworthiness and benchmarking. This assist in assessing the intrusion detection systems. MATLAB simulation is used here for evaluating the KDD dataset. The characteristics of our proposed setup are:

(i) 41 features

(ii) 9 absolute features

(iii) 32 consecutive extracts.

Additionally, the attacks are categorized into 4 key classes. (i) DoS attacks (ii) R2L (iii) U2R (iv) Probing.

TABLE 2. CONFUSION MATRIX FOR SECURITY ATTACK CLASSIFICATION

		Predicted class	
		Attack	***Normal***
Actual class	***Attack***	TP	FN
	Normal	FP	TN

The performance of the proposed IBCNN with IWDA approach is compared with the existing IRVM with GFA approach in terms of accuracy, precision, recall, f-measure and time complexity.

1. Accuracy

The accuracy is computed as like :

$$\text{Accuracy} = \frac{T_p + T_n}{(T_p + T_n + F_p + F_n)} \quad (15)$$

2. Precision

It is the measure of the total number of positive samples that are presumed to be positive that are gathered from the positive samples. Specifically, the values are selected for the false positive and true positive.

$$\text{Precision(P)} = \frac{T_p}{T_p + F_p} \quad (16)$$

3. Recall

It is the measure assessed for predicting the positive instances from the dataset of positive instances already trained. It relies on the True positive rate and the detection rate.

$$\text{Recall(R)} = \frac{T_p}{T_p + F_n} \quad (17)$$

4. F-measure

F-measure computes the accuracy of the system by taking recall r and precision f

4. Time complexity

It takes the amount of time to classify the attacks in the KDD dataset with improved CNN

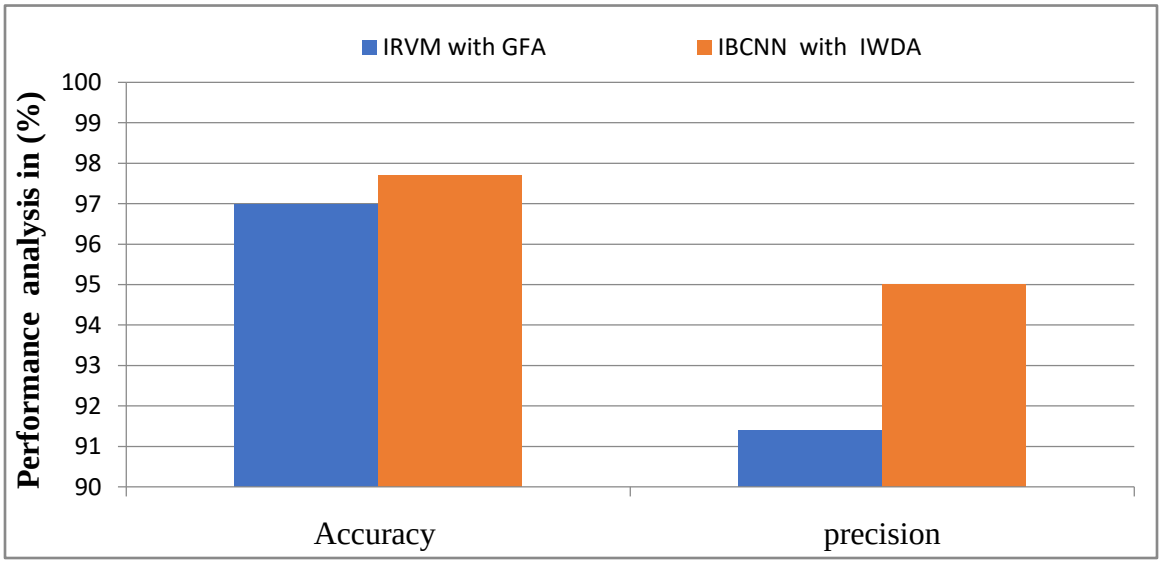

Figure 3: Accuracy and precision comparison

By plotting the experimental values of IBCNN and IWDA approach, it is evident that the proposed system outperforms the existing system for the factors such as accuracy and precision as shown above (figure 3). As stated in above sections, the optimal features that are obtained by IBCNN improves the detection and accuracy rate. The performance results are examined for IBCNN and IWDA , in which the proposed model yields 97.7% accuracy and existing model yields 97% accuracy. Moreover, the precision of IBCNN is 95% and IRVM is 91.4%

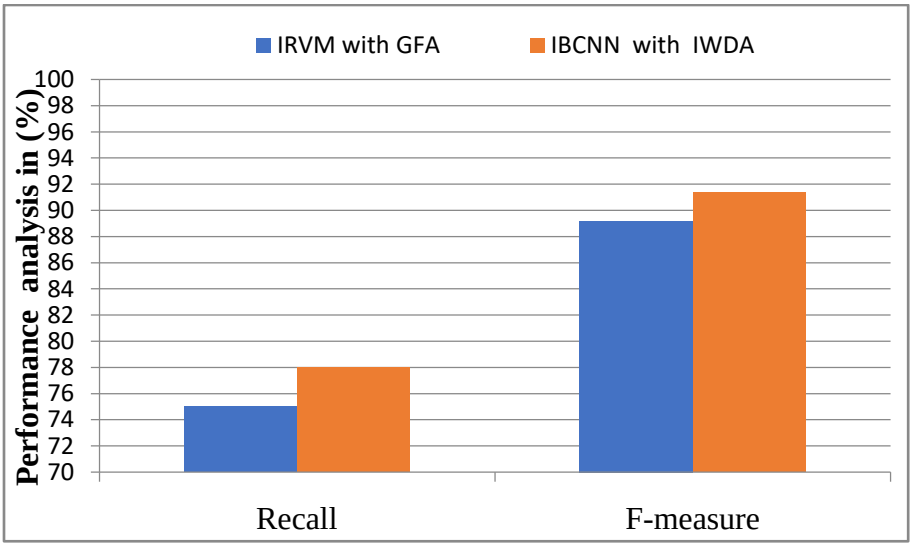

Figure 4: Recall and F-measure comparison

The Figure 4 depicts that the proposed system shows better enhancement with recall and f-measure values when compared with the IWDA approach. In IBCNN the values of the bias are optimized to get the better recall values. The experimental results revealed that the proposed system gives 78% of the recall accuracy as the IRVM produces only 75%. Additionally, the f-measure of proposed system is 91.4% whereas the IRVM yields 89.2%

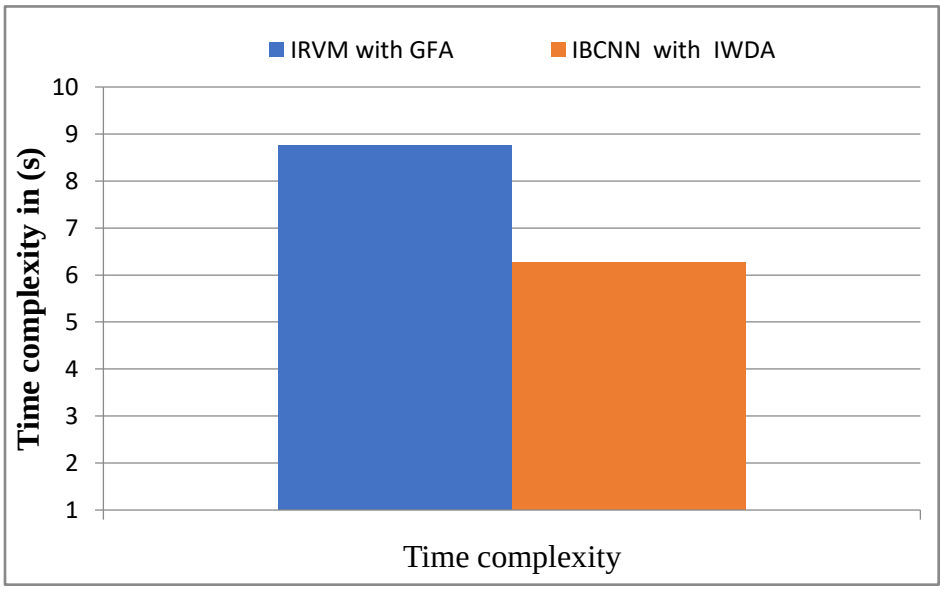

Figure 5. Time complexity of IBCNN with IWDA

The above graph (figure 6) reveals that the propose system outperforms the existing methods by producing good f-measure. It is evident that the proposed IBCNN gives reduced time complexity of 6.27s when compared to the IRVM with 8.75s

5. Conclusion and Future Work

The proposed Improved Bias based Convolutional Neural Network (IBCNN) model efficiently performs the enhanced IDS. Also, the proposed model facilitated the selection of optimal features that are obtained preprocessed information using Inertia Weight based Dragonfly Algorithm. Moreover, Kalman filtering is adopted for preprocessing the raw data. The suggested system utilizes the selected features for performing the effective IDS. The classification is then performed by the Improved Bias based Convolutional Neural Network (IBCNN) on the selected attributes. Since, the proposed model exploits the KDD dataset for effective classification. The simulation results have shown that the proposed system produces the better performance when compared with the existing systems in terms of recall, f-measure, accuracy and precision. Despite of the advantages, the combat measures for tackling the attacks with signature and buffer overflow needs addressing. Hence, enhanced deep learning have to be put forth to overcome the issues and challenges faced in the existing researches.

References

1. Sultana, A., &Jabbar, M. A. (2016, July). Intelligent network intrusion detection system using data mining techniques. In *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (ICATCCT)* (pp. 329-333). IEEE.

2. HuW, Gao J,WangY,WuO,Maybank S (2014) Online adaboost-based parameterized methods for dynamic distributed network intrusion detection. IEEE Transactions on Cybernetics 44(1):66–82.

3. Upadhyaya D, Jain S (2013) Hybrid approach for network intrusion detection system using k-medoid clustering and Naïve Bayes classification. Int J ComputSci Issues (IJCSI) 10(3):231–236.

4. Nadiammai, G. V., &Hemalatha, M. (2014). Effective approach toward Intrusion Detection System using data mining techniques. *Egyptian Informatics Journal*, *15*(1), 37-50.

5. Catania, C. A., &Garino, C. G. (2012). Automatic network intrusion detection: Current techniques and open issues. *Computers & Electrical Engineering*, *38*(5), 1062-1072.

6. Adebowale, A., &Idowu, S. A. (2013). An Enhanced Data Mining Based Intrusion Detection System (IDS) using Selective Feedback.

7. Mohammed, R. G., &Awadelkarim, A. M. (2011). Design and implementation of a data mining-based network intrusion detection scheme. *Asian Journal of Information Technology*, *10*(4), 136-141.

8. Aggarwal, P., & Sharma, S. K. (2015). Analysis of KDD dataset attributes-class wise for intrusion detection. *Procedia Computer Science*, *57*, 842-851.

9. Devaraju, S., &Ramakrishnan, S. (2014). Performance comparison for intrusion detection system using neural network with kdd dataset. *ICTACT Journal on Soft Computing*, *4*(3).

10. Bostani, H., &Sheikhan, M. (2017). Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems. *Soft computing*, *21*(9), 2307-2324.

11. Al Mehedi Hasan, M., Nasser, M., & Pal, B. (2013). On the KDD'99 dataset: support vector machine based intrusion detection system (ids) with different kernels. *Int. J. Electron. Commun. Comput. Eng*, *4*(4), 1164-1170.

12. Acharya, N., & Singh, S. (2018). An IWD-based feature selection method for intrusion detection system. Soft Computing, 22(13), pp.4407-4416.

13. Mathiyalagan R, Pamela Vinitha Eric.(2020). REVIEW ON INTRUSION DETECTION SYSTEM BASED METHODS ON KNOWLEDGE DISCOVERY DATA(KDD) DATASET. CR. 2020; 7(10): 3023-3029

14. Kuang F, Zhang S, Jin Z, Xu W (2015) A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection. Soft Comput 19:1187. doi:10.1007/s00500-014-1332-7

15. Ingre, B., Yadav, A., &Soni, A. K. (2017, March). Decision tree based intrusion detection system for NSL-KDD dataset. In International Conference on Information and Communication Technology for Intelligent Systems (pp. 207-218). Springer, Cham.

16. Zhao, H., Li, M., & Zhao, H. (2019). Artificial Intelligence based Ensemble Approach for Intrusion Detection Systems. Journal of Visual Communication and Image Representation, 102736.

17. A. M. Chandrasekhar, K. Raghuveer," Intrusion Detection Techniques by using K-means, Fuzzy Neural network and SVM classifier", ICCCI-2013, Jan. 04-06, 2013, Coimbatore, INDIA.

18. Kim, Jin, et al. "Method of intrusion detection using deep neural network." Big Data and Smart Computing (BigComp), 2017 IEEE International Conference on. IEEE, 2017.

19. Thaseen, I. Sumaiya, and ChAswani Kumar. "Intrusion detection model using fusion of PCA and optimized SVM."Contemporary Computing and Informatics (IC3I), 2014 International Conference on. IEEE, 2014.

20. Belouch, M., El Hadaj, S., &Idhammad, M. (2017). A two-stage classifier approach using reptree algorithm for network intrusion detection. *International Journal of Advanced Computer Science and Applications*, *8*(6), 389-394.

21. Ambusaidi, M. A., He, X., Nanda, P., & Tan, Z. (2016). Building an intrusion detection system using a filter-based feature selection algorithm. IEEE transactions on computers, 65(10), 2986-2998.

22. Jabbar, M. A., &Samreen, S. (2016, October). Intelligent network intrusion detection using alternating decision trees. In 2016 International Conference on Circuits, Controls, Communications and Computing (I4C) (pp. 1-6). IEEE.

23. KS, S. R., &Murugan, S. (2017). Memory based hybrid dragonfly algorithm for numerical optimization problems. *Expert Systems with Applications*, *83*, 63-78.

24. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354-377.

25. Shruthishree S.H1, Dr.Harshavardhan Tiwari2, Dr.DevarajVerma C*3,"ResNet Deep learning technique to improve breast cancer detection on screening mammography", JOURNAL OF CRITICAL REVIEWS, ISSN-2394-5125, VOL 7, ISSUE 20, 2020, AUG 2020.