INTELLIGENT-BASED ENSEMBLE DEEP LEARNING MODEL FOR SECURITYIMPROVEMENT IN REAL-TIME IOT ENVIRONMENT UNDER DIFFERENT NETWORK DATASET

Wali Mohammad Wadeed[1]
Assistant professor at IT Department, Faculty of Computer Science Kundoz University
Salih khan Salih[2]
Assistant Professor, Faculty Computer ScienceDepartment, Information System, Shaikh Zayed University
KhoshalRahman Rahmani[3]
Assistant professor at IT Department Faculty of Computer Science Kundoz University,
Md Sohel Rana[4]
NanjingUniversity of information science and technology

Abstract

In IoT environment, intrusion detection is the process of examining the identification of anomalous in the network. The intrusion detection system is subjected to challenges with the huge volume of data for the training and streaming which impacts the prediction process. Additionally, the imbalance in the intrinsic data exhibits challenges in the intrusion detection system. In this paper proposed anintelligent-basedsecurity model for the IoT environment. The proposed model is termed an Intelligent based ensemble classifier (IbeC) for the effective processing in intrusion detection in the network for handling transactions with an ensemble model. The proposed IbeC uses the bagging mode for the evaluation of the overlapping bags in the data sample for the training data. Every data in the need to be evaluated with respect to the base learner for the resultant prediction with the heuristics integrated voting process. The developed model comprises of the retraining model with the ensemble mechanism with the integrated signatures. The data is approximately pre-processed and evaluated with the deep learning model for the prediction. The developed IbeC model performance is evaluated for the three different dataset such as as NSL-KDD, KDD CUP 99 and Koyoto 2006 +. The deep learning model is appropriately fitted for the given network intrusion data and the final predictions are obtained. The comparative analysis expressed the hat proposed IbeC exhibits ~2% increased accuracy for the APID and HBM model.

Keywords: Security, intelligent model, ensemble model, Intrusion detection model, heuristics
DOI Number: 10.14704/nq.2022.20.6.NQ22625 **NeuroQuantology 2022; 20(6):6205-6222**

1. Introduction

Internet has become one of the mostly used resources. The huge explosion of internet usage has resulted in people performing digitized transaction [1]. The increase in number of internet users is 4% to 6% every year. In developing countries like India, the growth is observed to be much higher. The advancement of internet and computers leads to generation of vast range of generated data [2]. The advancement of technologies are indispensable for the human life components that are hard to evaluate in presence of online. The development of online causes the personal information sharing

or maintenance of personal details online. The higher in the convenience leads to data access for the computation of the vulnerable attacks or intrusion in the IoT environment. Those intrusion system subjected to the loss of money and relates the private information of the unintended users [3].

In an IoT environment, an Intrusion detection system is involved in computation of the unauthorized or anomaly activities. In the Intrusion Detection System (IDS) in the IoT environment is involved in the classification of user activities either anomalous or normal based on the data transmitted [4]. Conventional security scheme incorporates data encryption technique, authentication and firewalls. However, the intrusion in the IoT environment comprises of the highly sophisticated security mechanism for the conventional protection scheme. The resulted IoT environment increases the research domain with increases in the contribution of the domain research [5].

IDS models have several applications and requirements in the industry scenarios. A major application is the process of intrusion detection in personal systems, or in other words distributed scenarios. Current operating systems have intrusion detection mechanisms inbuilt into their architecture [6]. However, the handling capabilities of these systems are still in question. Hence most users tend to use commercial intrusion detection models for added security. Further, IDS for clustered environments that can be used in servers are also in demand. Several commercial IDSs are available, which includes Bro intrusion detection system by Vern Paxson from Lawrence Berkeley National Labs and the International Computer Science Institute, Prelude intrusion detection system for Linux, distributed under GNU, Snort intrusion detection system, Network Protocol Analyzer, Multi Router Traffic Grapher (MRTG),etc. However there is room for further improvements in computational requirements and accuracy of most of these systems [7].

The increased level of cybercrimes and the monetary losses associated with such crimes indicate the dire need for effective mechanisms for detecting network intrusions. Detecting intrusions in networks is one of the major requirements due to the high usage levels and transmission of large amount of sensitive information through networks [8]. One of the major issues to be considered in designing intrusion detection systems is concept drift that affects intrusion detection to a large extent. The data domain comprises of the concept drift for the affected domain data distribution basedon the time [9]. The implemented model for the training comprises of the static data that affect the large extent of concept drift. Those model are evaluated for a specific period of time with the trained signature towards invalid in the considerable time period. With the intrusion system, the developed model enables the continuous data monitoring mechanism for updated normal signatures up to date [10].

The security issue associated with the intrusion detection system in IoT security is evaluated with the data imbalance with the intelligence scheme. The intelligence- model evaluates the data imbalance for the dominance in the existing classes for the normal packet with the classification of anomalous packet data. In the intelligence model, the anomalous data training is over-trained with the normal data [11]. Hence, it is necessary to evaluate the intelligence model to design an effective security scheme in IoT environment. It is necessary to evaluate the performance accuracy of the intrusion detection scheme those needs to be realized with the most accurate possible model. However, the constraints are not reduced with the acceptable criteria domain time-constraint approach. With an effective prediction process model can be trained and processed effectively.

2. Related Works

In IoT environment, Intrusion detection is a mandatory component in the provision of interconnection between components. The security issues in the IoT environment is evaluated based

on the consideration of the different domain [12]. The security issue in the IoT environment is evaluated based on the constructed statistical model integrated with the Least Square Support Vector Machine (LS-SVM). The developed LS-SVM is examined based on the consideration of the subgroups in arbitrary forms. With appropriate model training the representative sample is performed and evaluated under the consideration of different subgroups. Another SVM-based model comprises the IDS-based intrusion detection scheme for security model in IDS system for IoT environment [13]. Basic machine learning techniques are also applied by several models and they also exhibit effective predictions. Such models include genetic and fuzzy algorithm based models by [14], clustering and k-nearest neighbor based models, IDS using Support Vector Machines (SVM) for training, etc. These models are also based on signature-based detection of intrusions. They are usually trained as binary classifiers and are trained on both normal andanomalous signatures. The models were observed to be computationally complex, leading to huge time requirements.

In the IoT environment, the layered IDS system evaluated the feature selection model based on the consideration of the different layer attacks and select the features at each layer to train the model with detection technique [15]. A real time intrusion detection based anomaly model is proposed for the security improvement in the IoT environment. The presented model uses the flexible signature database for update and reproduce the database in a real-time environment. The presented model uses the multi-objective feature selection approach for the effective identification of the attributes to improve the accuracy. To withstand the intrusion detection scheme predecessor are integrated with the embedded system multi-level intrusion model detection scheme [16]. The tree based anomaly detection scheme is developed with the integration firefly algorithm integrated with genetic algorithm in the intrusion detection system. With the clustering intrusion detection model semi-supervised multi-layered clustering model (SMLC) us developed with the for the partial labelling of the training data with flexible data detection technique. The security scheme with the intrusion detection comprises of the semi-supervised, cluster based IDS and natural neighbor model. The developed model exhibits the imbalance data evaluation in the security model.

An effective feature selection model for the KDD CUP 99 dataset was proposed in [17]. The model was able to achieve very high predictions with just 6 features. Similarly, the Flexible Neural Trees model in [18] was able to achieve 99.19% accuracy with just 4 features. A C# based intrusion detection model was proposed in [19]. This method makes a packet sniffer that can collect packets from an interactive TCP session to enable effective examining of packets. A major issue when working with packet sniffing models is that intruders tend to perform packet chaffing. The model also aids in helping researchers identify such packets by additional injection of packets into the network.

The featureprediction model improves the security model with intrusion detection through the elimination of unnecessary components. The security model in the ANN estimates the feature detection to improve the performance of the network. Through feature ranking based on attributes, information gain is achieved. By attribute filtering with correlation measures, the information gain is achieved for the ranks. The estimation is based on the consideration of the feature selection model SVM in the security feature model. In large features, the ranking features are computed based on the lightweight IDS and feature augmentation model with the SVM model [20].

The signature matching in the intelligent-based scheme uses the signature-based detection for faster performance to minimize the false alarm rates. Another model for the computation of the packet payload is computed based on the textual features in the IoT environment [21]. The estimation of the textual features are utilized for the extraction of label with n-gram model. The

model extracted features are involved in construction of the model repositories. The statistical mechanism uses the derived signature with the effective prediction model based on computation capabilities with feature enhancement for the improved dataset for the base data quality improvement. Additionally, the bigram feature model is developed for payload traffic encoding in the feature selection. The feature reduction comprises of the different techniques to minimize the data size and improves the model prediction efficiency. The proposed model comprises of different categories such as embedded, wrappers and filters. The proposed dimensionality reduction model uses the intelligent model with integrated ensemble classifier, Information Gain (IG) and Principal Component Analysis (PCA) to construct hybridized mode for the effective prediction model. The developed feature reduction model comprises of the evolutionary heuristics model integrated with the firefly algorithm for the security model in intrusion detection scheme [22].

The major advantage of feature selection based models is that they tend to reduce the data size. However, drift detection becomes extremely complex when features are eliminated. Hence the models may not prove to be effective for domains experiencing concept drifts.

3. Intelligent based ensemble classifier model for security improvement

The proposed IbEC comprises of the distinct components such as feature reduction, portioning of the data, ensemble prediction, aggregator and retraining component. The overall process involved in the proposed IbEC is presented in figure 1.

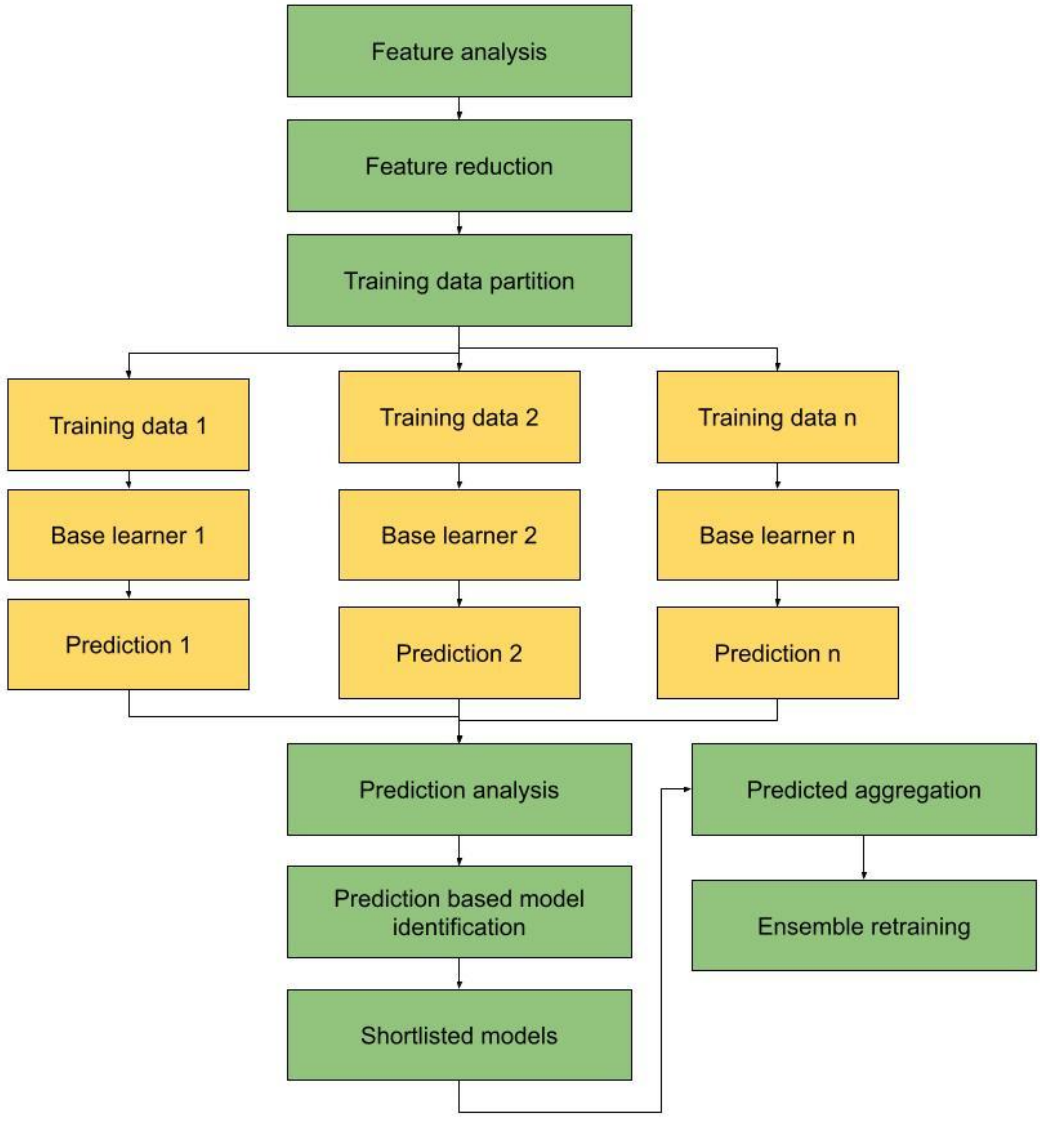


Figure 1: Steps in IbEC

As stated in figure 1 initially, Feature reduction is performed which ispreceded by feature analysis. The analysis of the features aisbased on the dependency of the feature identities the correlation level in the class label. The accurate prediction is observed with the computation of effective correlation between the features for the prediction. The negative correlation between variables are computed based on zero and neural correlation. Based on the estimation of complexity level features are computed for the zero and negative factors in large extent. Consider data with the n instances with k classes, composed of d dimensions.The feature vectors are given in equation (1)

$$X_i = [x_{i1}, x_{i2}, \dots \dots x_{id}] \forall_i = 1,2, \dots, n \quad (1)$$

The attributes correlation between the xi, xj is represented as in equation (2)

$$r_{xi,xj} = \frac{\sum_k x_{ik} x_{jk} - n\overline{\overline{s_k x_j}}}{\sqrt{(\sum_k x_i \ ^2 - n\overline{x_t x_j})(\sum_k x_i \ ^2 - n\overline{x_t x_j})}} \quad (2)$$

The positive correlation provides the direct proportional correlation with the direct connection, the proportionality of the model exhibits the negative correlation between variables. The data final labels are computed with the determining role in the data with satisfaction of the features to retained the data threshold features. Each of these data subsets is passed to a base learner, the machine learning model used for training and prediction. The base learners can be homogeneous or heterogeneous, depending on the requirement of the base data. All the models can be run in parallel, hence reducing the time complexity levels. The architecture ensures that each model is independent and does not depend on the other models. The final predictions are obtained by combining the predictions of all these models and aggregating them to a single entity.The process of bagging is shown in Figure 2.

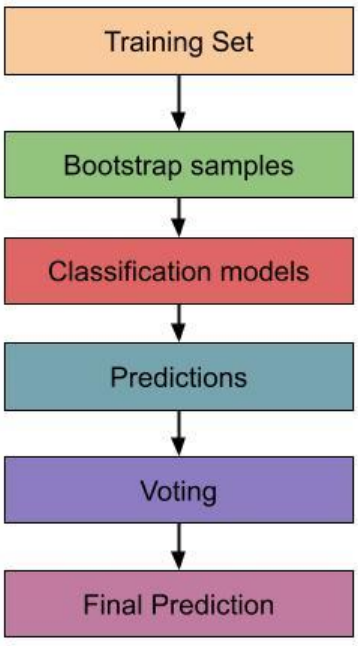


Figure 2: Process in Bagging

3.1 Ensemble based heuristics model for security in IoT

In the proposed IbeC the ensemble classification is performed with the heuristics approach for the IoT security. With the base learner and prediction the data is passed and evaluated for the entire data without any division in the data. The prediction is performed with the heuristics

combiner model to determine the final prediction model with multiplication of the different prediction feature instances in the data. In the end phase of the proposed architecture modet two prediction groups are implemented prediction segregation to determine the initial part for the final prediction. The segregation criteria (SA) is represented as in equation (3) and equation (4)

$$S_A = \{(x_i, C_i) \mid x_i \in \text{Pred}_A \wedge C_i = 1\} \quad \forall 1 \leq i \tag{3}$$

$$S_N = \{(x_i, C_i) \mid x_i \in \text{Pred}_N \wedge C_i = 0\} \quad \forall 1 \leq i \leq |\text{Pred}_N| \tag{4}$$

The prediction is performed base don the consideration of certain instances such as SN and SA. Those instances are discrepancies with the filtered sets those comprises of the common data. The process of filtering is represented as in equation (5)

$$\text{Common} = \{(x_i, C_i) \mid (x_i, 1) \in \text{Pred}_A \wedge (x_i, 0) \in \text{Pred}_N\} \tag{5}$$

The final prediction is derived based on the combination of the prediction of segregated data as SA and SN those are represented as in equation (6)

$$\text{Final} = (S_A \cup S_N) - \text{Common} \tag{6}$$

Based on certain instances the prediction is performed with the common representation those are identified as in equation (7) – equation (9)

$$\text{AdditionalN} = \{(x_i, C_i) \mid (x_i, 0) \in \text{Pred}_A\} \forall 1 \leq i \leq |\text{Pred}_A| \tag{7}$$

$$\text{AdditionalP} = \{(x_i, C_i) \mid (x_i, 1) \in \text{Pred}_N\} \forall 1 \leq i \leq \| \text{Pred}_N \|_1 \tag{8}$$

$$\text{Common} = \text{Common} \cup \text{AdditionalN} \cup \text{AdditionalP} \tag{9}$$

With the prediction algorithm the prediction is performed to obtain the final prediction set for the processing.Intelligent based ensemble classifier (IbEC) model for fast and effective intrusion detection. Neural network is considered to be one of the major models that aim to provide effective predictions. Performing intrusion detection with deep networks can effectively uncover several intrinsic patterns and can also effectively handle data imbalance and concept drift. The proposed deep learning architecture for intrusion detection consists of four major phases: the data pre-processing phase, data segregation phase, network construction phase and the model fitting phase. The architecture of the proposed model is shown in Figure 3 and pseudocode for the same is provided below.

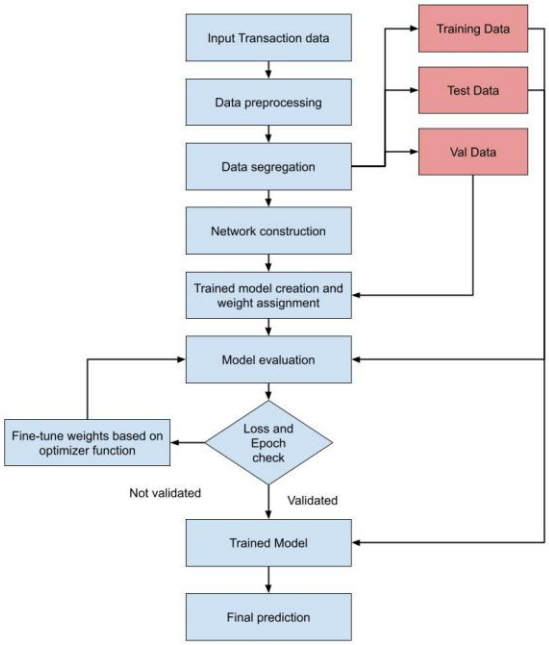


Figure 3: Flow chart of IbEC

Data pre-processing phase in Neural networks do not typically handle all the data types contained in the network data. They are capable of handling only double data types and also require the data to be in the same range, so as to provide equal significance to all the attributes. These requirements are handled by the pre-processing phase. Data imputation is followed by data normalization. Data normalization is one of the mandatory pre-processing steps when working with real-time data. The major need for normalization arises from the operational nature of the machine learning models.Hence it becomes mandatory to convert all the data within similar ranges such that consistency can be achieved during the prediction process. Three major and mostly used normalization techniques include Min-Max normalization, Z-Score normalization and Decimal Scaling. The Min-Max normalization (Dodge 2003) provides linear transformation on the original range of data. This is given and represented as in equation (10)

$$\text{Prediction} = \frac{w_1v_1+w_2v_2+\cdots w_nv_n}{n}, \quad x' = \left(\frac{x-\min(A)}{\max(A)-\min(A)}\right) * (D-C) + C \qquad (10)$$

where, x' denoted as the normalized value and attribute of the actual value defined as A.C and D with the predefined boundaries [C,D] in the scaled data. Z-Score normalization is another technique used for the process of normalization. This model normalizes the data between 0 and 1 intervals. The computation is presented in equation (11) as follows:

$$x_i = \frac{x_i-\bar{A}}{\text{std }(A)} \qquad std(A) = \sqrt{\frac{1}{(n-1)}\sum_{i=1}^{n}(x_i-\bar{A})^2} \quad x' = \frac{x}{10^j} \qquad (11)$$

The decimal scaling is the simplest method that provides results based on the current value and the maximum value in the attribute.Neural network or artificial neural network is a system of neurons that operate together to perform effective machine learning. Neural networks are composed of singular processing components called neuron or perception. A single neuron takes multiple inputs to provide a single output as depicted in Figure 4.

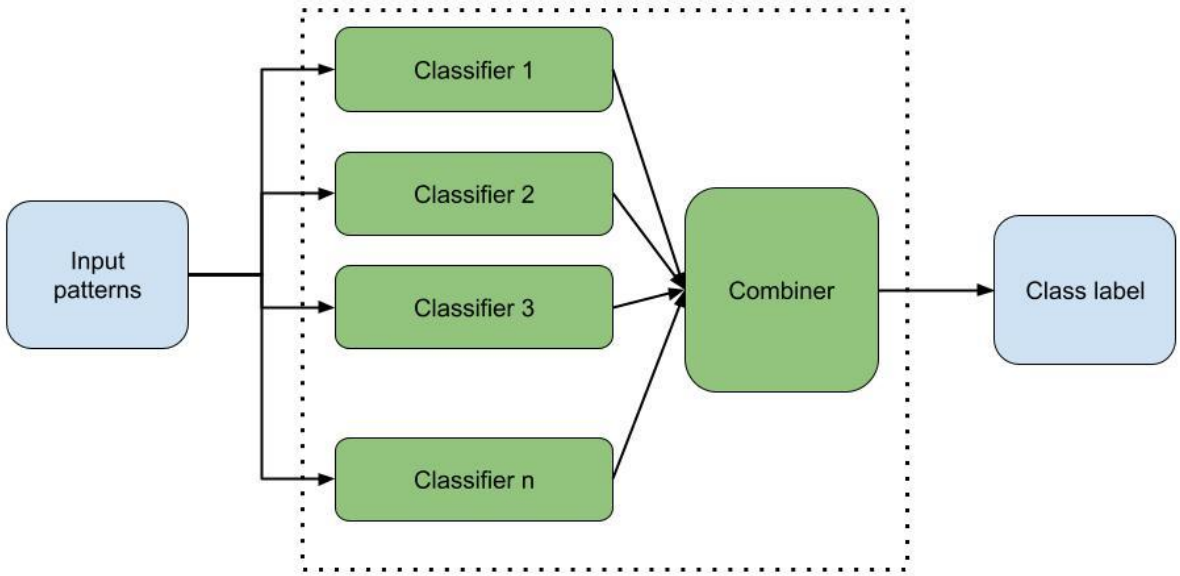


Figure 4: IbEC classification label

These inputs however cannot be operated upon in isolation. Hence every input is provided weights, which represents the significance of the input. The weights, represented as w1,w2…wn are usually real numbers. The output of a neuron is usually a weighted aggregation of the input value and its corresponding weights. The process in attack classification is denoted in equation (12)

$$\text{Output} = \varphi\sum_{i=1}^{n} \text{w}_\text{i}x_\text{i} \qquad (12)$$

where, φ is the activation function, w and x are the weights and inputs of the neuron. A neural network is generally composed of multiple layers, usually an input layer, one or two processing or hidden layers and an output layer. Every layer is composed of multiple neurons operating on the inputs given in that layer and providing the appropriate outputs. Every layer in the network operates from the input provided by the previous layer, and provides its output to the next layer. The overall process involved in proposed IbEC for deep learning model is presented in figure 5.

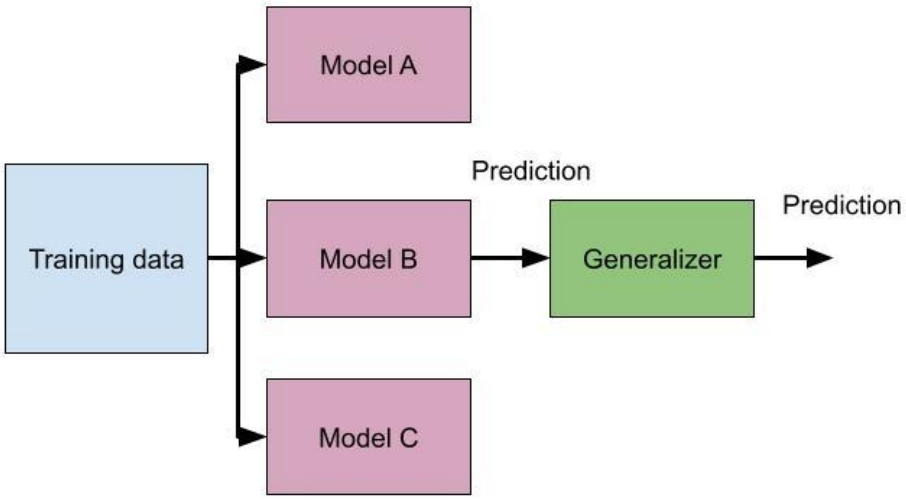


Figure 5: Neural Network Process in IbEC

The neural networks model is created using the sequential API. The network consists of three types of layers, the input layer, multiple hidden layers and an output layer. This work uses a learning rate of 0.5. Learning rate defines the rate at which the model has to move towards the optimal solution. Too small values signify slower convergence, while larger values might result in the model skipping the optimal solution. Hence setting the optimal value for the problem under analysis is mandatory. The epoch level is set to 50. Epochs define the number of times the training data is provided to the neural network model to enable learning. Higher epochs will result in better models. However, care should be taken to avoid overfitting of the model. The parameters used are presented in Table 1.

Table 1: Neural Network Parameters for IbEC

Parameter	Value
Network Type	Sequential
Size of batch	64
Epochs	50
Learning Rate	0.1
Optimizer	Adam

The proposed IbEC neural network is constructed with varied layer configurations depending on the dataset used for analysis. All these layers are designed as dense layers. A layer is considered to be dense if all nodes of a layer are connected with all the nodes of the next layer. This aids in creating a network that passes all its results to all the available nodes in the network. The actual output depends on the activation function that is being used. Several activation functions are available, however, four of the functions that are mostly used in neural networks include; sigmoid, hyperbolic tangent (tanh), Rectified Linear Units (ReLU) and linear.

*e*ISSN 1303-5150

4. Results and Discussion

The proposed IbEC architecture is implemented in Python using Keras libraries. The model has been verified using standard benchmark datasets like NSL-KDD, KDD CUP 99 and Koyoto 2006+ datasets. The network is built using Sequential API and the layers are added to create the neural network. A separate network architecture has to be created for each dataset. The structures created for each of the used datasets are shown in table 2, 3 and 4.

Table 2: Configuration of NSL-KDD dataset

Layer (Type)	**Activation Function**	**Input Dimension**	**Output Shape**	**No. of Parameters**
Input (Dense)	Linear	41	(None,80)	3360
Processing 1 (Dense)	ReLU	80	(None,100)	8100
Processing 2 (Dense)	ReLU	100	(None,50)	5050
Processing 3 (Dense)	ReLU	50	(None,50)	1530
Output (Dense)	Linear	30	(None,1)	31
Total No. of Parameters				18,071
Trainable Parameters				18,071
Non-trainable Parameters				0

Table 3: Configuration of KDD CUP 99

Layer (Type)	**Activation Function**	**Input Dimension**	**Output Shape**	**No. of Parameters**
Input (Dense)	Linear	38	(None,50)	1950
Processing 1 (Dense)	ReLU	50	(None,250)	12,750
Processing 2 (Dense)	ReLU	250	(None,100)	25,100
Processing 3 (Dense)	ReLU	100	(None,50)	5050
Output (Dense)	Linear	50	(None,1)	51
Total No. of Parameters				44,901
Trainable Parameters				44,901
Non-trainable Parameters				0

Table 4: Configuration of Koyoto 2006+

Layer (Type)	**Activation Function**	**Input Dimension**	**Output Shape**	**No. of Parameters**
Input (Dense)	Linear	18	(None,50)	950
Processing 1 (Dense)	ReLU	50	(None,200)	10,200
Processing 2 (Dense)	ReLU	200	(None,100)	20,100
Processing 3 (Dense)	ReLU	100	(None,20)	2020
Output (Dense)	Linear	20	(None,1)	21
Total No. of Parameters				33,291
Trainable Parameters				33,291
Non-trainable Parameters				0

A performance analysis of the proposed IbEC model on NSL-KDD, KDD CUP 99 and Koyoto 2006+ datasets in terms of their ROC plots is shown in Figure 6. The ROC plots exhibit high performances, as the plots are aligned towards the top right of the graph. This implies very low False Positive Rate (FPR) and very high True Positive Rate (TPR), which is the major requirement for any classifier.Performance in terms of precision and recall is shown in Figure 7. The graph shows that the proposed model exhibits high precision and high recall levels, demonstrating the performance of an effective classifier model.

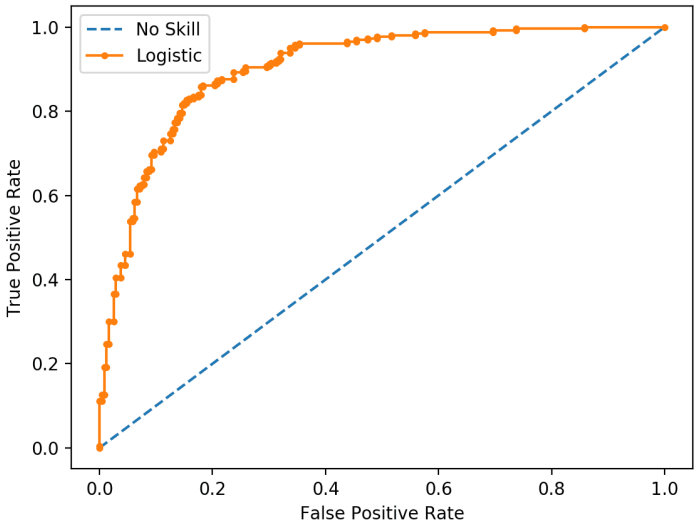


Figure 6: ROC of datasets

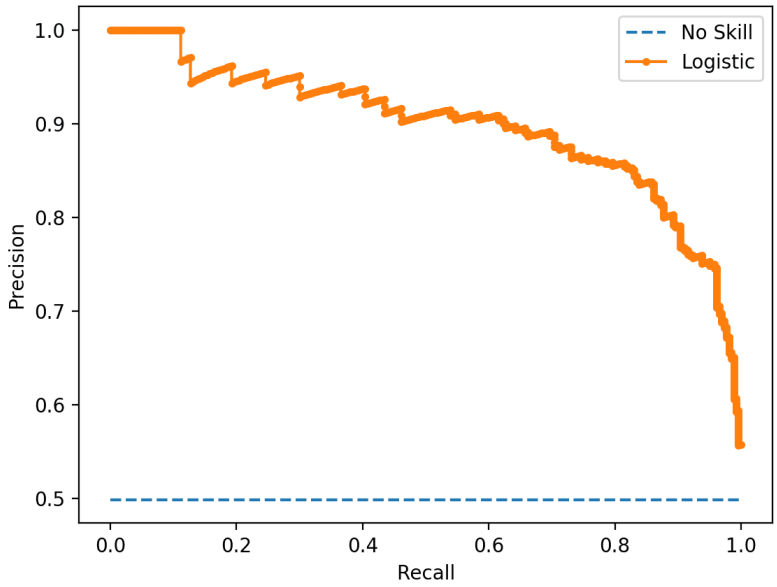


Figure 7: Comparison of precision Vs Recall

The values obtained for various performance measures such as FPR, TPR, Recall, Precision etc. On NSL-KDD, KDD CUP99 and Koyoto 2006+ datasets are presented in the Table 5.The proposed IbEC model offers very high TPR and Precision levels indicating its efficiency in predicting intrusion signatures. Similarly, high TNR level indicates that the proposed model also exhibits high prediction efficiency when predicting normal transmission signatures. Similarly, low FPR and FNR levels of< 1% indicates that the model exhibits very low false prediction levels. From this it is evident that the proposed IbEC model is effective and provides high performance.

Table 5: Performance Analysis of proposed IbEC

Measures	**NSL-KDD**	**KDD CUP 99**	**Koyoto 2006+**
FPR	0.001934	0.001254	0.005605
TPR	0.991718	0.998519	0.93578
Recall	0.991718	0.998519	0.93578
Precision	0.997917	0.995079	0.953271
TNR	0.998066	0.998746	0.994395
FNR	0.008282	0.001481	0.06422
Accuracy	0.995	0.9987	0.988012
F-Measure	0.994808	0.996796	0.944444
AUC	0.994892	0.998632	0.965087

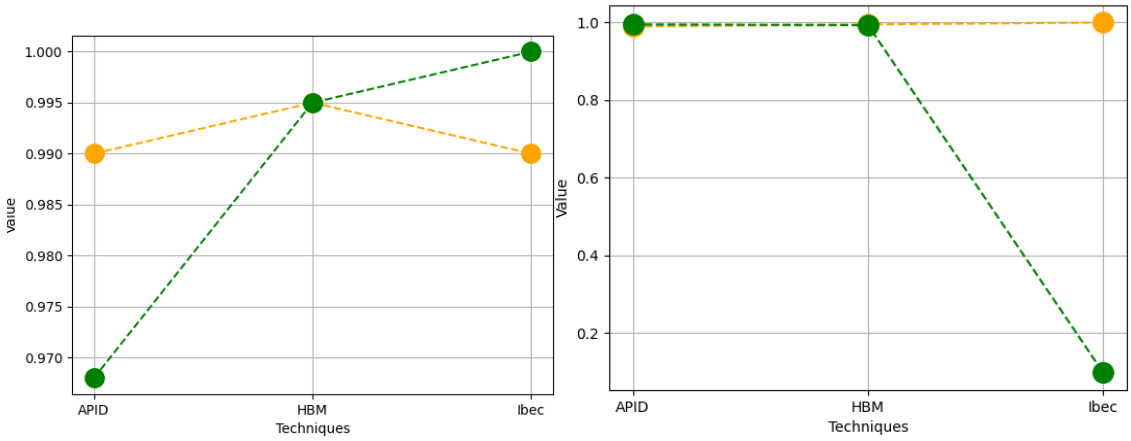
The parameter settings were varied and a sensitivity analysis was performed on all the three datasets to identify the effects of variedlearning rate and number of epochs. Multiple parameter pairs were used for analysis and accuracy obtained for each of the parameter set is used for analysis. Results obtained are presented in Table 6.Learning rate is varied and epoch is kept constant in the first five sets (P1 to P5). It could be observed that, as the learning rate is reduced, the performance on all tree datasets tend to reduce to a slight extent (P1 and P2). Asthe learning rate is reduced, the model moves towards the optimal solution in very small steps. Hence 50 epochs were not sufficient to achieve convergence. Increasing the number of epochs as in P11 shows that the model was able

to achieve convergence. Increasing the learning rate as in P4 and P5 shows reduced performances. This is attributed to the fact that an increasedEpochs are varied and learning rate is kept constant in parameter sets P6 to P10. Reduced epochs (P6 and P7) show that the model is not given sufficient amount of time to converge. Hence the best accuracy is not obtained. As the epoch is moved to 50 (P3) and beyond (P8 to P10), the best accuracy is obtained. It could be observed that best accuracy is observed with 50 epochs. Increasing the number of epochs after this point exhibits no impact on the performance, as convergence has already been achieved.

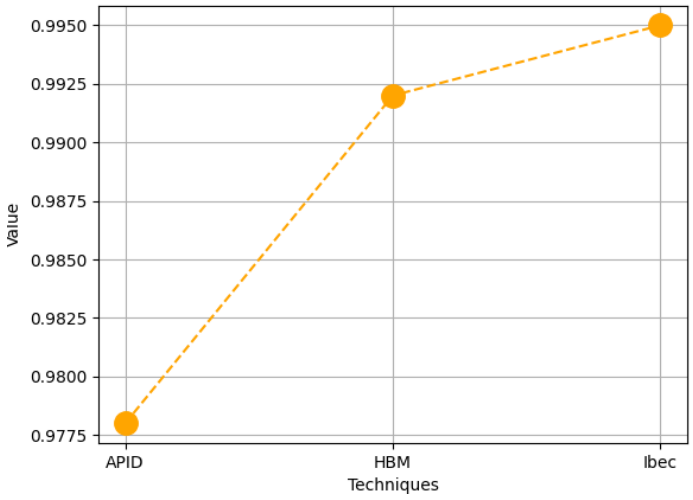
Table 6: Analysis of Sensitivity

Parameter Set	Learning Rate	Epochs	NSL-KDD	KDD CUP 99	Koyoto 2006
P1	0.1	50	0.94	0.937	0.913
P2	0.3	50	0.97	0.959	0.944
P3	0.5	50	0.995	0.999	0.979
P4	0.7	50	0.991	0.999	0.973
P5	1	50	0.89	0.926	0.851
P6	0.5	10	0.72	0.69	0.583
P7	0.5	25	0.79	0.829	0.811
P8	0.5	70	0.995	0.999	0.979
P9	0.5	100	0.995	0.999	0.979
P10	0.5	200	0.995	0.999	0.979
P11	0.3	200	0.995	0.999	0.978

It could be summarized that learning rate plays a vital role in achieving effective results. It is necessary to identify the sweet spot with the optimal convergence level. Any value below this point requires more time to converge, any value beyond this point will make the model miss the convergence point. Epochs are the number of times the model should iterate through the training data to achieve convergence. Lesser number of epochs will not provide sufficient time for the model to converge and providing higher number of epochs than necessary will just be an additional time overhead with no performance improvement. Further, it will also lead to overfitting, hence should be avoided.The proposed IbEC model is compared with the HBM and the APID models, in terms of TPR, TNR, Precision, F-Measure and AUC on NSL-KDD, KDD CUP 99 and Koyoto 2006+ datasets which are shown in Figures 5.11to 5.19.



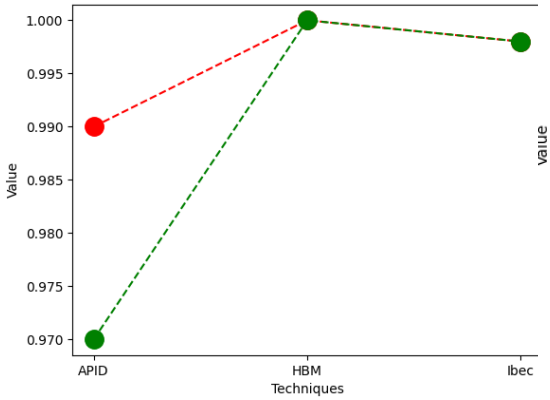
(a) (b)



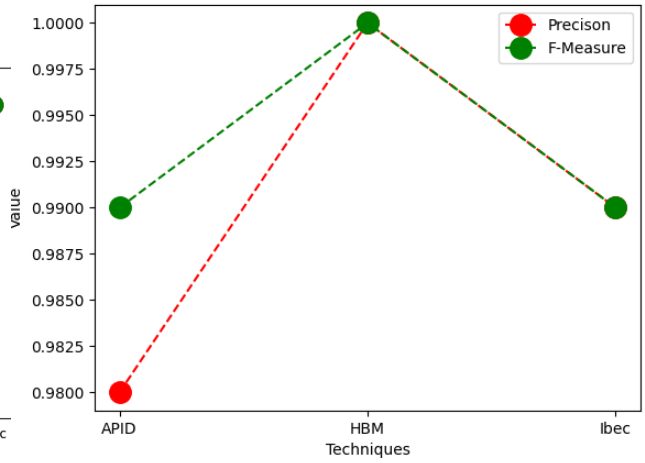
(c)

Figure 8: NSL – KDD (a) comparison of TPR and TNR (b) Comparison of F – measure (c) Comparison of AUC

A comparison of TPR, TNR, Precision, F-Measure and AUC on NSL-KDD data figure 8 shows that the IbeC model exhibits better prediction levels when compared to APID and HBM.

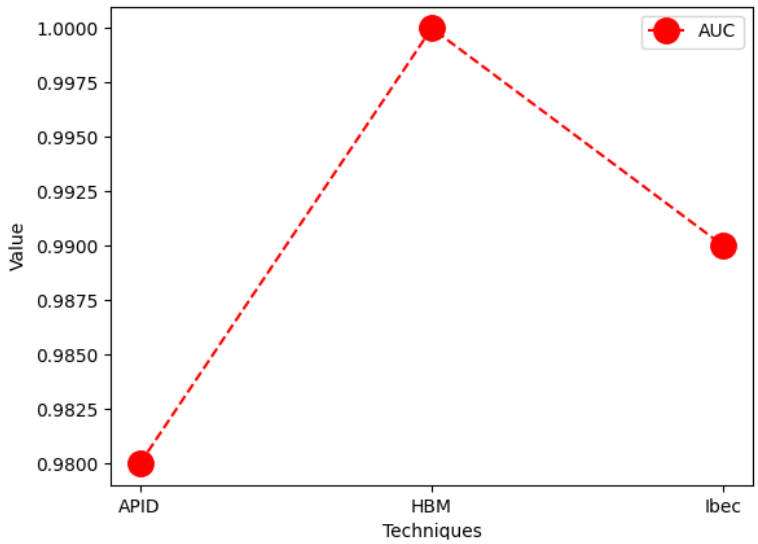


(a)



(b)

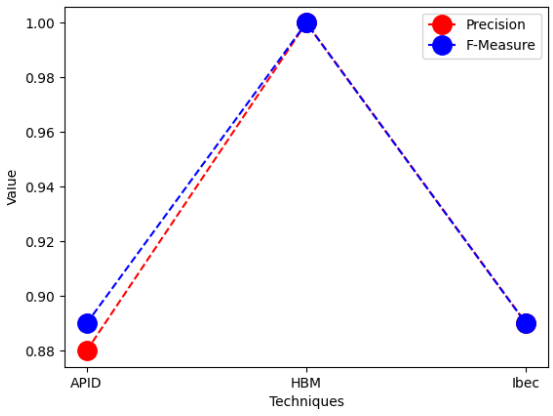
*e*ISSN 1303-5150



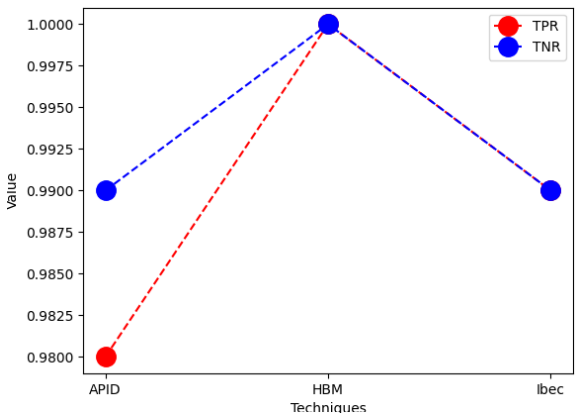
(C)

Figure 9: KDD – CUP 99 (a) comparison of TPR and TNR (b) Comparison of F – measure (c) Comparison of AUC

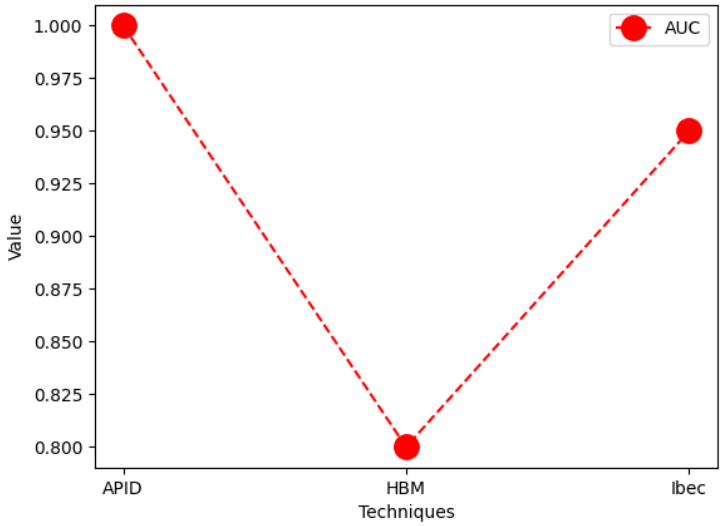
Analysis of performance on KDD CUP 99 dataset shown in Figures 9 shows that the proposed IbeC model exhibits better performance compared to APID model. However, the performance levels show a slight reduction of ~0.1% when compared with the HBM model. The reduction levels arenegligibly very low and hence can be ignored.



(a)



(b)



(c)

Figure 10: Koyoto 2006 (a) comparison of TPR and TNR (b) Comparison of F – measure (c) Comparison of AUC

The Performance of the proposed IbeC model on Koyoto 2006+ datasets also shows a slight decrease in the performance with respect to certain metrics such as TPR, F-Measure when compared to the other models. Here also, the reductions are very low, hence are negligible. A tabulated view of the performance comparisons is shown in Table 7. The table shows that, although there are slight reductions and elevations in the performance levels of the proposed models, the overall performance was found to be high and effective.

Table 7: Comparative Analysis of Proposed IbeC

Measures	**NSL-KDD**			**KDD CUP 99**			**Koyoto 2006+**		
	APID	**HBM**	**IbeC**	**APID**	**HBM**	**IbeC**	**APID**	**HBM**	**IbeC**
TPR	0.99	0.99	0.99	0.99	1.00	1.00	0.98	0.88	0.94
Precision	0.99	0.99	1.00	0.99	1.00	1.00	0.94	0.88	0.95
TNR	0.97	0.99	1.00	0.97	1.00	1.00	0.99	0.99	0.99
Accuracy	0.99	0.99	1.00	0.99	1.00	1.00	0.99	0.98	0.99
F-Measure	0.99	0.99	0.99	0.99	1.00	1.00	0.96	0.88	0.94
AUC	0.98	0.99	0.99	0.98	1.00	1.00	0.99	0.93	0.97

The prediction accuracy exhibited by the proposed models in comparison with the existing models in literature is shown in Figures 11, 12 and 13. It could be observed that, in terms of the performance, the IbeC model achieves the highest performance, followed by the HBM model with the next best performance.

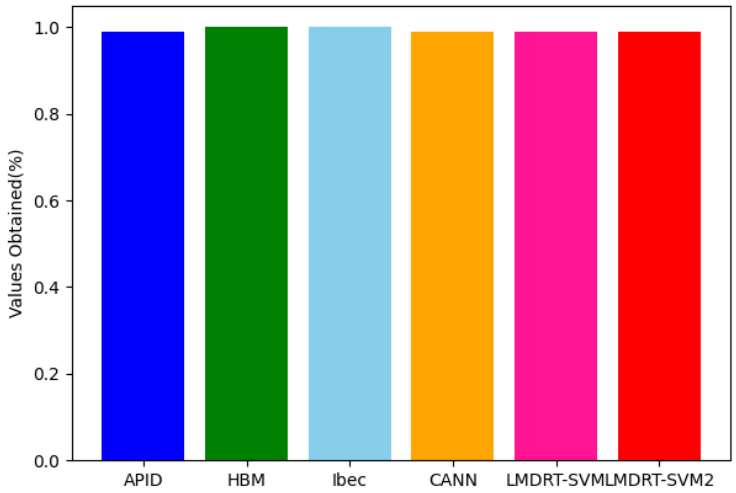


Figure 11: Comparison of KDD CUP 99 Accuracy

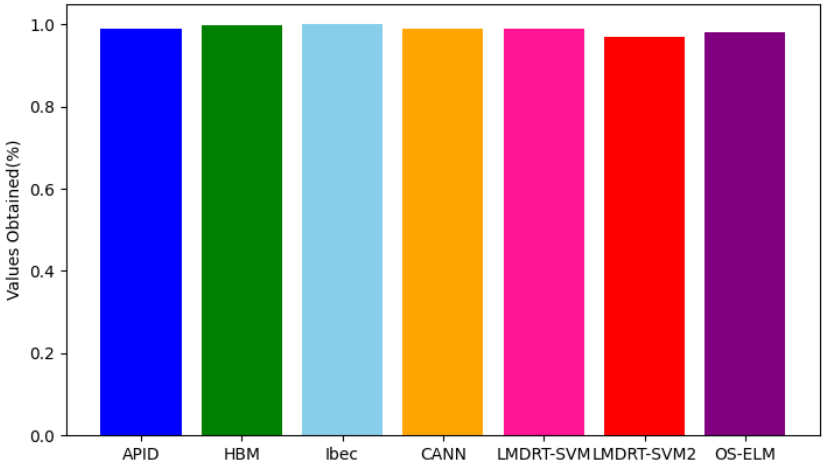


Figure 12: Comparison of NSL - KDD

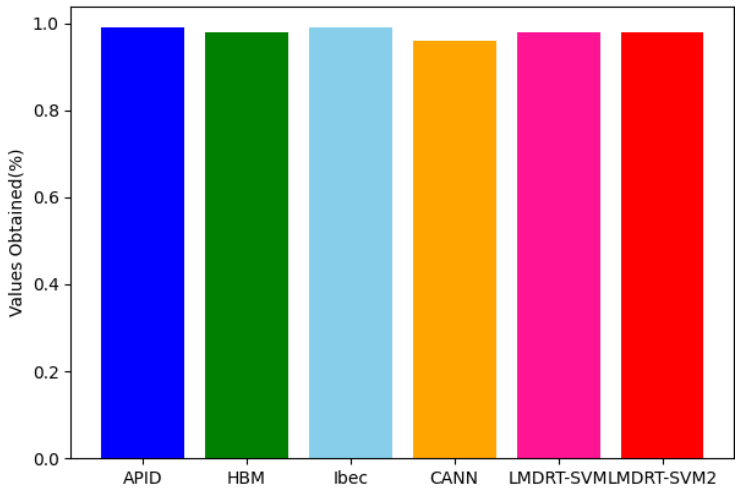


Figure 13: Comparison of Koyoto 2006+

It could be observed that the proposed models exhibit higher performances when compared to all the existing models in literature used for comparison. In general, techniques are constructed based on the data. Such data specific models fail to provide effective results when operated upon different data. This case could be effectively observed in the models from literature, where models exhibit better results in certain scenarios and reduced performances in certain other scenarios. The proposed models are generic; hence it could be observed that the proposed model exhibits effective performance irrespective of the dataset being used.

5. Conclusion

A neural networks based model for effective intrusion detection in networked environments has been discussed in this work. The proposed IbeC model was constructed using Keras. Experiments were conducted using standard benchmark datasets and comparisons were performed with existing models in the literature. The values obtained for various performance measures indicate that the proposed IbeC model exhibits high performance compared to the existing models, proving the high generalizability and the effective performance achieved by the proposed architecture.

REFERENCES

1. Wang, F., Jiang, D., Wen, H., & Song, H. (2019). Adaboost-based security level classification of mobile intelligent terminals. *The Journal of Supercomputing*, *75*(11), 7460-7478.
2. Khalid, T., Khan, A. N., Ali, M., Adeel, A., & Shuja, J. (2019). A fog-based security framework for intelligent traffic light control system. *Multimedia Tools and Applications*, *78*(17), 24595-24615.
3. Pirbhulal, S., Wu, W., Muhammad, K., Mehmood, I., Li, G., & de Albuquerque, V. H. C. (2020). Mobility enabled security for optimizing IoT based intelligent applications. *IEEE Network*, *34*(2), 72-77.
4. Al-Omari, M., Rawashdeh, M., Qutaishat, F., Alshira'H, M., & Ababneh, N. (2021). An intelligent tree-based intrusion detection model for cyber security. *Journal of Network and Systems Management*, *29*(2), 1-18.
5. Al-Khafajiy, M., Otoum, S., Baker, T., Asim, M., Maamar, Z., Aloqaily, M., ... & Randles, M. (2021). Intelligent control and security of fog resources in healthcare systems via a cognitive fog model. *ACM Transactions on Internet Technology (TOIT)*, *21*(3), 1-23.
6. Al-Saud, M., Eltamaly, A. M., Mohamed, M. A., & Kavousi-Fard, A. (2019). An intelligent data-driven model to secure intravehicle communications based on machine learning. *IEEE Transactions on Industrial Electronics*, *67*(6), 5112-5119.
7. Lei, W., Wen, H., Wu, J., & Hou, W. (2021). MADDPG-based security situational awareness for smart grid with intelligent edge. *Applied Sciences*, *11*(7), 3101.
8. Sarker, I. H., Furhad, M. H., & Nowrozy, R. (2021). Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. *SN Computer Science*, *2*(3), 1-18.
9. Siriwardhana, Y., Porambage, P., Liyanage, M., & Ylianttila, M. (2021, June). AI and 6G security: Opportunities and challenges. In *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)* (pp. 616-621). IEEE.
10. Zwane, Z. P., Mathonsi, T. E., & Maswikaneng, S. P. (2021, May). An Intelligent Security Model for Online Banking Authentication. In *2021 IST-Africa Conference (IST-Africa)* (pp. 1-6). IEEE.

11. Zhang, Z., Cao, Y., Cui, Z., Zhang, W., & Chen, J. (2021). A many-objective optimization based intelligent intrusion detection algorithm for enhancing security of vehicular networks in 6G. *IEEE Transactions on Vehicular Technology*, *70*(6), 5234-5243.

12. Javeed, D., Gao, T., Khan, M. T., & Shoukat, D. (2022). A Hybrid Intelligent Framework to Combat Sophisticated Threats in Secure Industries. *Sensors*, *22*(4), 1582.

13. Devi, A., Therese, M. J., & Premalatha, G. (2021). Cloud computing based intelligent bank locker system. In *Journal of Physics: Conference Series* (Vol. 1717, No. 1, p. 012020). IOP Publishing.

14. Kim, J., & Kim, M. (2021). Intelligent mediator-based enhanced smart contract for privacy protection. *ACM Transactions on Internet Technology (TOIT)*, *21*(1), 1-16.

15. Veeramakali, T., Siva, R., Sivakumar, B., Senthil Mahesh, P. C., & Krishnaraj, N. (2021). An intelligent internet of things-based secure healthcare framework using blockchain technology with an optimal deep learning model. *The Journal of Supercomputing*, *77*(9), 9576-9596.

16. Rathore, S., Park, J. H., & Chang, H. (2021). Deep learning and blockchain-empowered security framework for intelligent 5G-enabled IoT. *IEEE Access*, *9*, 90075-90083.

17. Chen, L., Zheng, M., Liu, Z., Lv, M., Zhao, L., & Wang, Z. (2021). SDAE+ Bi-LSTM-Based Situation Awareness Algorithm for the CAN Bus of Intelligent Connected Vehicles. *Electronics*, *11*(1), 110.

18. Haseeb, K., Din, I. U., Almogren, A., Ahmed, I., & Guizani, M. (2021). Intelligent and secure edge-enabled computing model for sustainable cities using green internet of things. *Sustainable Cities and Society*, *68*, 102779.

19. Tai, Y., Gao, B., Li, Q., Yu, Z., Zhu, C., & Chang, V. (2021). Trustworthy and intelligent covid-19 diagnostic iomt through xr and deep-learning-based clinic data access. *IEEE Internet of Things Journal*, *8*(21), 15965-15976.

20. Deebak, B. D., & Fadi, A. T. (2021). Lightweight authentication for IoT/Cloud-based forensics in intelligent data computing. *Future generation computer systems*, *116*, 406-425.

21. Gonçalves, F., Macedo, J., & Santos, A. (2021). An Intelligent Hierachical Security Framework for VANETs. *Information*, *12*(11), 455.

22. Sarker, I. H. (2022). Ai-based modeling: Techniques, applications and research issues towards automation, intelligent and smart systems. *SN Computer Science*, *3*(2), 1-20.