iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** X **Month of publication:** October 2023

DOI: https://doi.org/10.22214/ijraset.2023.56020

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue X Oct 2023- Available at www.ijraset.com

A Predictive Model Using Deep Learning Neural Network for Efficient Intrusion Detection

A. Priyavengatesh, Dr. R. Kannan
[1]*Ph.D Scholar, Department of Computer Science, Sri Ramakrishna Vidhyalaya Arts & Science College Coimbatore-641020*

Abstract: ***Network intrusion detection system helps to detect exploitations and mitigate damages. A network intrusion detection system detects the network traffic that deviates from the normal behavioral pattern. Developing an efficient intrusion detection system has many challenges and the patterns associated with one type of intrusion differ from other intrusions. In such situations, understanding different patterns and differentiating intrusions becomes essential to detect anomalies and attacks in the network. Deep learning models offer more power and intelligence to the detection system and extend the ability to differentiate & understand the network feature characteristics, also machine learning models with feature selection showed high performance in intrusion detections. This paper evaluates the proposed deep learning neural network model and machine learning models using feature selection for efficient intrusion detection using real world dataset.***

I. INTRODUCTION

Intrusion detection system (IDS) refers to detecting abnormal patterns in the network and threats to the network. Establishing IDS is to secure the network from various exploitations and intrusions. Intrusions are broadly classified into anomalies and signature-based which are incorporated into the detection system as two main strategies. Anomaly based intrusion detection strategy compares with normal behavioral patterns while signature-based IDS detects the distinguishing patterns and prevents the network attacks and threats (Sultana et al., 2019). The present trends of exploitations demands for an effective IDS that could evolve to the next level of migrating threats and exploitations (Louati et al., 2020). The emergence of new behavioral patterns in the network occur daily, understanding and identifying new attacks becomes essential.

Many machine learning models are extensively studied for IDS and more capabilities are added to those models. Deep learning models are gaining popularity in network security for its feature learning ability and intelligence to discover newer and hidden patterns (Lin et al., 2022). Since network traffic data contain several different traffic features, traditional machine learning models suffers from high dimensional data which resulted in inferior learning and poor detection of threats and attacks while deep learning models are capable of training large volume of data and extracting deep feature information which enhance the prediction quality of attack patterns (Yin et al., 2017). The advantage of deep learning models is that it can be tuned for optimal results without feature selection, capable of handling large volume of data and high dimensional features, automatically learns the hidden patterns and the trained models can be used for different applications and arising problems (Sahu et al., 2023). Feature selection methods utilize feature information that are derived statistically and more often performs better in terms of higher classification and prediction accuracy, processing time and dimensionality reduction but information are removed along with the features and also, the selected features cannot be generalized for other problems. This paper evaluates the proposed deep learning neural network model (1DCNN) and bagged trees model using chi square feature selection for efficient intrusion detection on NSL-KDD and CICIDS2017 dataset.

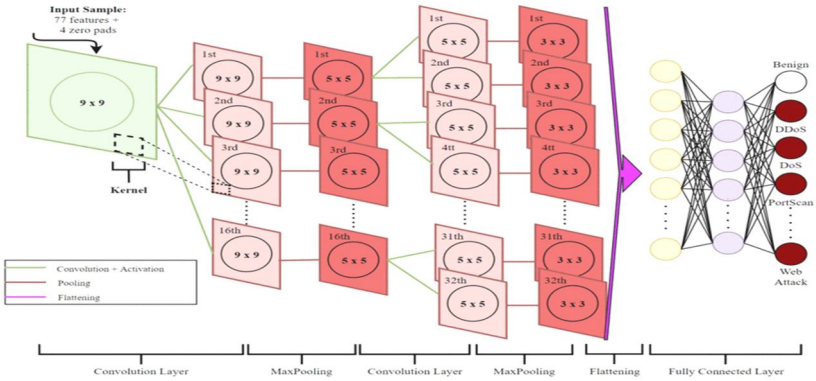


Fig 1 CNN architecture

II. RELATED WORKS

Building efficient IDS using machine learning models and relying on its performance is subjected to its accuracy of detection of various attacks. Generally, performance of machine learning models and deep learning models rely on training data (NG, B. A., & Selvakumar, S., 2020), where large volume of data affects the performance of machine learning models in terms of computation, accuracy and training time while deep learning models require expert parameter optimization (Diro, A. A., & Chilamkurti, N., 2018). (Henry et al., 2023) proposed a CNN-GRU deep learning model for IDS. The performance of the model is enhanced through correlation based feature selection. The proposed model showed a highest accuracy of 98.73% over CNN and LSTM on CICIDS2017 dataset. The feature selection contributed to the highest accuracy of the model by reducing the features to 44 from the total set of features. (Talukder et al., 2023) combines machine learning models with deep learning models to address dimensionality problems and achieve feature reduction & feature selection. The features are selected using XGBoost. The hybrid model achieved high performance of 99.99% and 100% on KDD-CUP99 and CIC-Malmem2022 datasets.

(Sharma et al., 2023) proposed an anomaly based IDS using deep neural network in IOT. The deep learning model is constructed using filter based feature selection and DNN. The feature selection removes the highly correlated features and the performance of the model is boosted with GAN network. The GAN network is used to remove class imbalance in the data. The performance of the model improved to about 7% while addressing the class imbalance using GAN and the model scores an accuracy of 84% without GAN. (Disha & Waheed, 2022) proposed an Intrusion detection system based machine learning models. The proposed Gini impurity based weighted random forest model is an embedded method that includes features selection. The feature selection utilizes gini-impurity to split the trees which adjust the feature weights according to the corresponding class levels. The performance of the model is compared with DT, GBT, MLP, LSTM and GRU on two different datasets. The performance of DT model showed improved accuracy after feature selection with 93.01% on UNSW-NB15 and 99.9% on Network TON_Iot dataset. The feature selection weights were adjusted to the level of class imbalance in the datasets which reduced the features and improved the class accuracy. (Priyavengatesh & Kannan, 2022) proposed a machine learning based IDS to capture cyber attacks. The proposed work utilizes chi-square features selection to reduce the features and ensemble bagged tress for classification. The proposed model achieved accuracy of 99.63% on NSL-KDD dataset with 5 features and outperforms LDA (86.21%), LR (86.32%), LSVM (87.39%) and NN (91.97%) models. The improvement in the performance of the model contributes to the features that have high association with the target class.

(Ho et al., 2021) proposed CNN based IDS to detect cyber attacks. The three layer CNN model is designed to classify multi-classes with pooling layer and fully connected layer. The performance of the model is compared with MLP, RF, NB and DT and the proposed model achieved highest accuracy of 99.64% on CICIDS2017 dataset outperforming other models. (Injadat et al., 2020) proposed a machine learning based IDS to predict network intrusions. The proposed model is multi stage optimized model which reduced the computational complexity through features selection and sampling methods. The proposed model reduced the feature size using correlation and info gain methods. The proposed method claimed that smote oversampling reduce the sampling size to about 30% and 60% in UNSW-NB and CICIDS 2017 dataset. The two feature selection methods reduced the feature size to about 60% and improved the classification accuracy to 99% on both the dataset.

III. METHODOLOGY

A. CNN Network Architecture

CNN is comprised of three layers namely convolution layer, pooling layer and fully connected layer where feature engineering is carried out in convolution layer and pooling layer while fully connected layer perform the classification. The convolution refers to the mathematical function that linearly operates and expresses a dot product of two functions and returns a third function. The output of the dot product of two functions is termed as ‘feature map or kernel map’ which is passed on to other layers to learn the features of the input data. For non linear functions, sigmoid, ReLU and Tanh functions are used. The pooling layer is used to reduce the feature dimensions and maintain relevant feature information discarding noises. Some of the pooling methods are min pooling, max pooling, global average pooling and global max pooling. The third layer is the fully connected layer where the features are converted into single vector. The elements of the vector are attached to the classes and each element is associated with a weight that defines the association to a particular class. The activation function in the fully connected layer defines the type of problem i.e. softmax for classification and sigmoid for prediction which converts the output values to the probability score for each class. The parameters and hyper parameters of Convolution Neural Network are given in Table 1.

Table 1 Parameters and Hyperparameters of CNN

	Parameters	Hyperparameters
Convolution layer	Kernels	Kernel size, number of kernels, stride, activation function
Pooling layer	-	Filter size, method, stride
Fully Connected layer	Weights	Number of weights, Activation function
Others	-	Learning rate, optimizer, loss function, batch size, epochs, regularization

B. Random Forest

A bagged tree is an ensemble method that builds decision trees from bagged samples and used for both classification and prediction problems. The process of building decision trees starts with bootstrap samples (bagging) where the samples are drawn into a bag with replacement. A decision tree is a construction that resembles a tree like structure. Each decision tree cast vote to a particular class and based on majority of voting, a particular class is assigned. In bagged trees, the class score for each class is given by equation 1 and the majority of vote in bagged trees is given in equation 2.

$$Cs(X, C_1) = \frac{v(X,C_1)}{btrees} \quad (1)$$

$$C^i = mvote\{C^{btrees}\}bCL_n \quad (2)$$

Where Cs is the class score, btrees is the bagged trees, X is the vector of input features, C_1 is the class score of the classifier; bCL_n is the ensemble of classifiers accuracy in OOB samples. The error rate of OOB samples is given in equation 3.

$$\text{MSE}^{oob} = \frac{1}{oob_n}\sum_n (Y - Y'^{oob})^2 \quad (3)$$

C. Chi-Square

Chi-square measures the degree of association between target class and the features in a dataset. High chi-square value indicates high association of a feature to its target class. The chi-square value is calculated using the equation 4.

$$C_i = \sum \frac{(O_c - e_c)2}{e_c} \quad (4)$$

Where O_c is the observed frequency, e_c is the expected frequency. Observed frequency is the number of cases for target class c and expected frequency is the expected number of cases for target class c when there is no relationship between feature and target class c. The lower chi-square score shows the lighter or no association between the feature and the class and it depicts that the feature and the class are independent.

D. Proposed 1D CNN

The performance of the convolution neural network depends on the architecture of the network design and varies according to different problem and sizes. It is difficult to find the suitable hyperparameters for a specific problem and adjusting the hyperparameters affects the model performance and accuracy. For optimal model performance fine tuning different parameters is required which requires more number of iterations to arrive the optimal values. Finding the right combination of hyperparameters improves the model performance and the model correctly classifies intrusions into normal and attack. Underperformance of the model shows that the model stability, processing time and computing resources is affected which can be improved through optimal hyperparameters tuning. To effectively classify intrusions, 1D CNN is constructed for binary problem. 1D CNN refers to the kernel that slides in one dimension. Three convolution layers are added after the input layer followed by two fully connected layers and an output layer. The first layer has 128 neurons and second convolution layer has 64 neurons and the second layer is interlaced with max pooling layer with pool size of 3 and kernel size of 3 x 3 with activation function relu. The third convolution layer has 32 neurons with kernel size of 3 x 3. To avoid overfitting a dropout layer is added to the third convolution layer in order to preserve the feature information. The output of the last convolution layer is flattened and passed on to fully connected layer. The fully connected layer has 10 neurons with softmax activation function predicts the network intrusion types. The loss function is given in equation 1.

$$\text{Loss} = -\sum_{i=1}^{k} y_j \log(y`_i) \quad (2)$$

where k is the number of classes, y is the actual class and y` is the predicted class. The proposed 1DCNN for CICIDS2017 and NSL-KDD datasets is given in Table 2 and Table 3.

Table 2 Proposed 1D CNN for CICIDS2017

Model	Layer	Output shape	Parameters	Filters	Kernel Size	Activation
ID CNN	Conv1d	(67,128)	384	128	3	relu
	Conv1d	(66,64)	16448	64	3	relu
	Dropout	(66,64)	0			
	MaxPooling1D	(33,64)	0			
	Conv1d	(32,64)	8256	64	3	relu
	Dropout	(32,64)	0			
	Flatten	(2048)	0			
	Dense	(10)	20490		3	relu
	Dense	(2)	22			softmax
Total Params: 45600 Trainable Params: 45600 Non trainable params: 0						

Table 3 Proposed 1D CNN for NSL-KDD dataset

Model	Layer	Output shape	Parameters	Filters	Kernel Size	Activation
ID CNN	Conv1d	(37,64)	256	65	3	relu
	MaxPooling1D	(18,64)	0			
	Conv1d	(16,64)	12352	64	3	relu
	Dropout	(16,64)	0			
	Flatten	(1024)	0			
	Dense	(10)	10250		3	relu
	Dense	(2)	22			softmax
Total Params: 22880 Trainable Params: 22880 Non trainable params: 0						

IV. EXPERIMENT AND ANALYSIS

A. Dataset

The proposed deep learning neural network model (1DCNN) and bagged trees model using chi square feature selection for efficient intrusion detection is evaluated on NSL-KDD and CICIDS2017 dataset. The KDD CUP 99 dataset was collected by DARPA using network traffic TCP dump (Stolfo et al., 2000) and it is an improved version of DARPA98 dataset. The data is collection of network traffic for three weeks with 48, 38,430 records as a part of IDS evaluation program (Lippmann et al., 200). The KDD CUP 99 dataset contains 41 features and labeled into DOS, R2L, U2R, probe, and normal. The traffic data with 41 features is grouped into basic feature, traffic feature and content feature. The CICIDS2017 dataset contains real-time traffic data captured over a five day period. The five day traffic data has a total of 2,299,308 instances which require a high computing power and long training hours. To test the proposed model, Thursday morning working hour's dataset is taken and the dataset has a total of 170368 instances with 78 features and 1 label column. 10 features that have zero values are removed and rest of the features is included in the study. The final dataset set contains 68 features and 1 class label with attack types and benign traffic. This study does not consider multi-class classification.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue X Oct 2023- Available at www.ijraset.com

B. Evaluation Metrics

The performance of the proposed deep learning model is evaluated using confusion matrix. Confusion matrix helps to compute quantitative performance metrics such as Accuracy, sensitivity, specificity and F1 score. The confusion matrix for binary classification is given in Table 4. It represents the actual values and predicted values.

Table 4 Confusion Matrix Predicted

Actual		Positive	Negative
	Positive	True Positive TP	False Positive FP
	Negative	False Negative FN	True Negative TN

The performance of a classifier is evaluated using TP, TN, FP and FN. TP refers to True Positives and it is the number of positive instances correctly classified as Positives. TN refers to True Negatives and it is the number of negative instances classified correctly as Negatives. FP refers to False Positives and it is the number of Negative instances incorrectly classified as Positive. FN refers to False Negatives and it is the number of Positive instances incorrectly classified as Negative.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{F1-score} = \frac{2TP}{2*TP + (FP+FN)}$$

V. RESULTS AND DISCUSSION

The proposed deep learning neural network model (1DCNN) and bagged trees model using chi square feature selection for efficient intrusion detection is evaluated on NSL-KDD and CICIDS2017 dataset. The performance of the proposed 1DCNN on all features for NSL-KDD and CICIDS2017 dataset is given in table 5. Comparing the results of 1DCNN and bagged trees on CICIDS 2017 dataset, the results shows that bagged trees achieved a highest accuracy of 99.98% on CICIDS dataset and 99.81% of accuracy on KDD dataset while 1DCNN achieved 99.70% on CICIDS dataset and 99.43% on KDD dataset which is 0.2% higher than 1DCNN model on CICIDS and 0.4% higher in KDD dataset. The higher result of bagged trees is the due to selection of best features among the bagged samples than searching for the most important features on all samples which reduces the variance in the features. Sensitivity refers to true positive rate which shows that ability of the model to predict positive cases as positive. The sensitivity of 1DCNN and bagged trees achieved 99.98% on CICIDS dataset and 1DCNN achieved sensitivity of 99.34% on KDD while bagged trees achieved sensitivity of 99.80%, which is 0.05% of higher than 1DCNN. Specificity refers to true negative rate which show the models ability to detect negative cases as negative. 1DCNN achieved specificity of 82.10% on CICIDS dataset and 99.53% on KDD dataset set while bagged trees scored specificity of 100% on CICIDS dataset and 99.81% on KDD dataset. The ability of 1DCNN to classify true negative cases as negative is less (18% on CICIDS and 0.2% on KDD dataset) than the bagged tree models.

Table 5 performance metrics of 1DCNN using all features

Accuracy	Sensitivity	Specificity	F1 score	Precision	Dataset
99.70%	99.98%	82.10%	99.85%	99.72%	CICIDS
99.43%	99.34%	99.53%	99.46%	99.59%	KDD

Table 6 performance metrics of bagged trees using all features

Accuracy	Sensitivity	Specificity	F1 score	Precision	Dataset
99.98%	99.98%	100%	99.99%	100%	CICIDS
99.81%	99.80%	99.81%	99.82%	99.84%	KDD

According to table 7, the performance of bagged trees using selected features on CICIDS dataset achieved accuracy of 98.80% and 99.84% on KDD dataset. Using chi-square feature selection, the features are reduced to 5 from 41 features in KDD and 68 features in CICIDS dataset. The selection of relevant features to the target class improved the bagged trees model performance on KDD dataset. The accuracy of the bagged tree model improves from 99.81% to 99.84% on KDD dataset while the accuracy decreases from 99.98% to 98.80% on CICIDS dataset. The decrease in model performance shows that the features selected from CICIDS does not supply enough information on the target class to distinguish the class benign and attack. Similarly the specificity also decreases from 100% to 96% on CICIDS dataset.

Table 7 performance metrics of bagged trees using feature selection

Accuracy	Sensitivity	Specificity	F1 score	Precision	Dataset
98.80%	98.80%	96%	99.40%	100%	CICIDS
99.84%	99.77%	99.93%	99.85%	99.94%	KDD

According to Table 5, the performance of 1DCNN when compared to bagged trees using feature selection showed an improved performance. The accuracy of the 1DCNN model achieved accuracy above 99% on both the dataset while bagged trees achieved 99% on KDD dataset and 98.80% on CICIDS dataset. The higher performance of 1DCNN is due to the fact that deep learning models are robust in feature engineering and the ability to map features that are important to the target class. Also the features are reduced to about 90% which could have removed the feature information on the target class leading to decreased bagged trees model performance on CICIDS dataset. Comparatively the proposed deep learning model has superior performance on both the dataset while feature selection slightly decreased the bagged tree model performance on CICIDS dataset

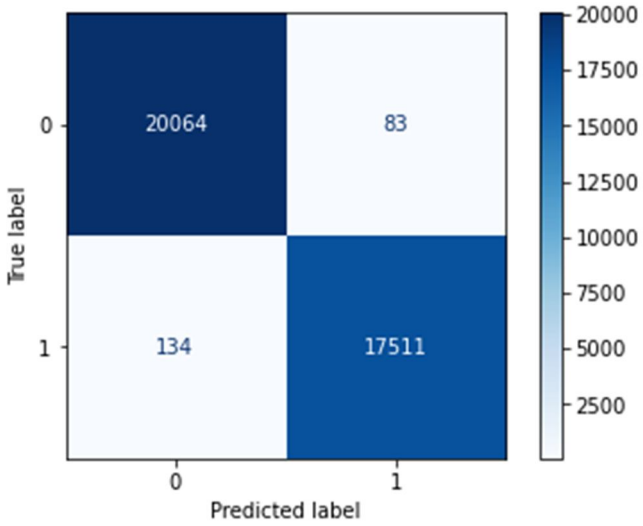


Figure 2 Confusion matrix for IDCNN on NSL-KDD dataset using all features

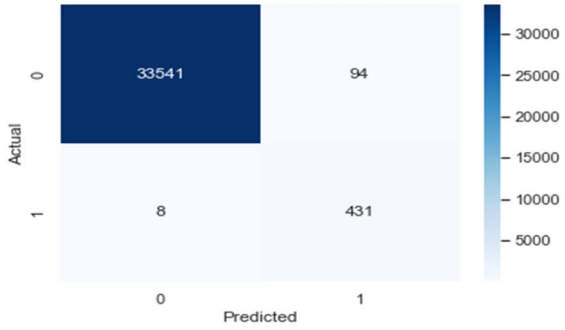


Figure 3 Confusion matrix for IDCNN on CICIDS2017 dataset using all features

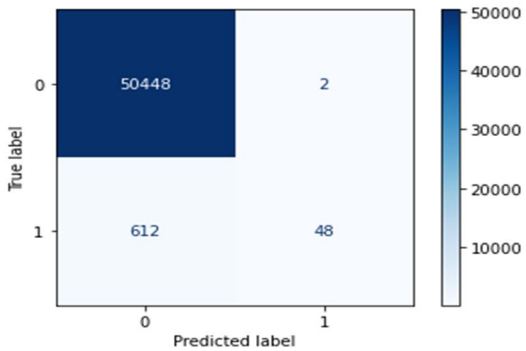


Figure 4 Confusion matrix for bagged trees on CICIDS2017 dataset using FS

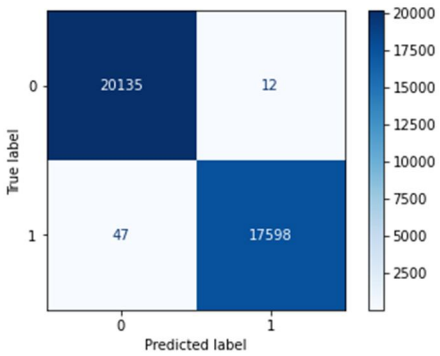


Figure 5 Confusion matrix for bagged trees on NSL-KDD dataset using FS

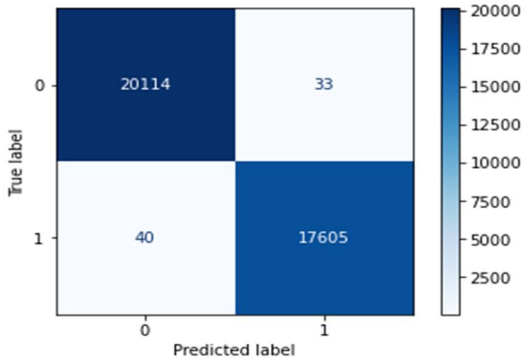


Figure 6 Confusion matrix for bagged trees on NSL-KDD dataset using all features

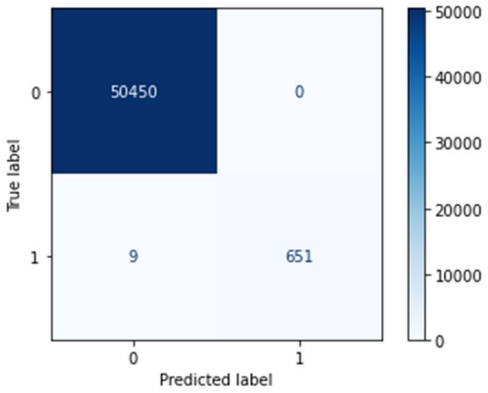


Figure 7 Confusion matrix for bagged trees on CICIDS2017 dataset using all features

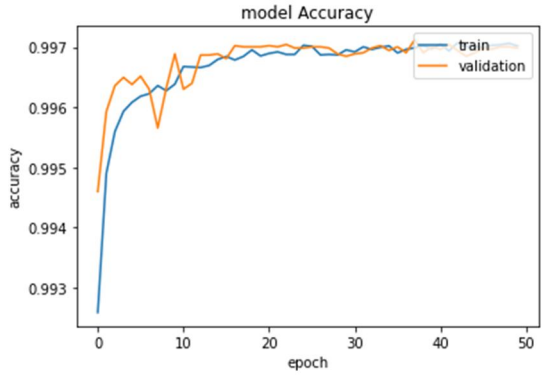


Figure 8 Accuracy for 1DCNN on CICIDS2017 dataset using all features

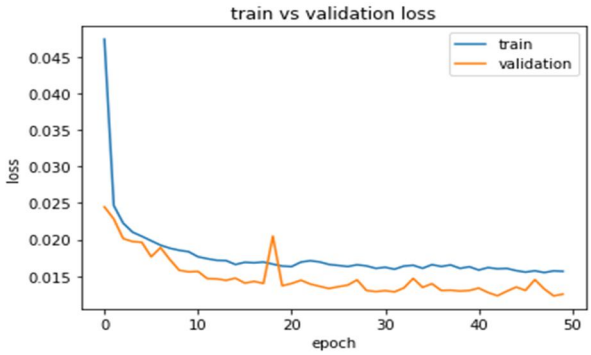


Figure 9 Validation loss for 1DCNN on CICIDS2017 dataset using all features

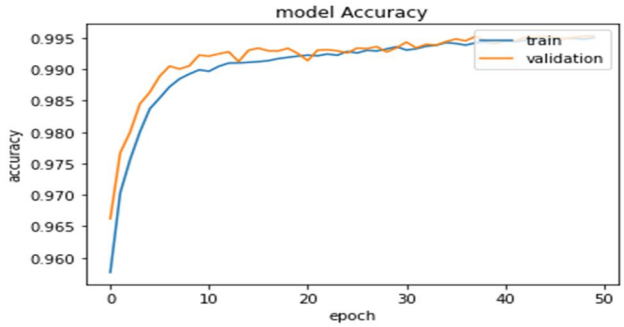


Figure 10 Accuracy for 1DCNN on NSL-KDD dataset using all features

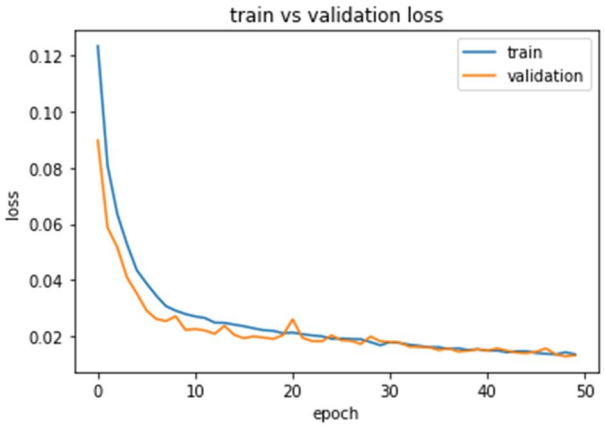


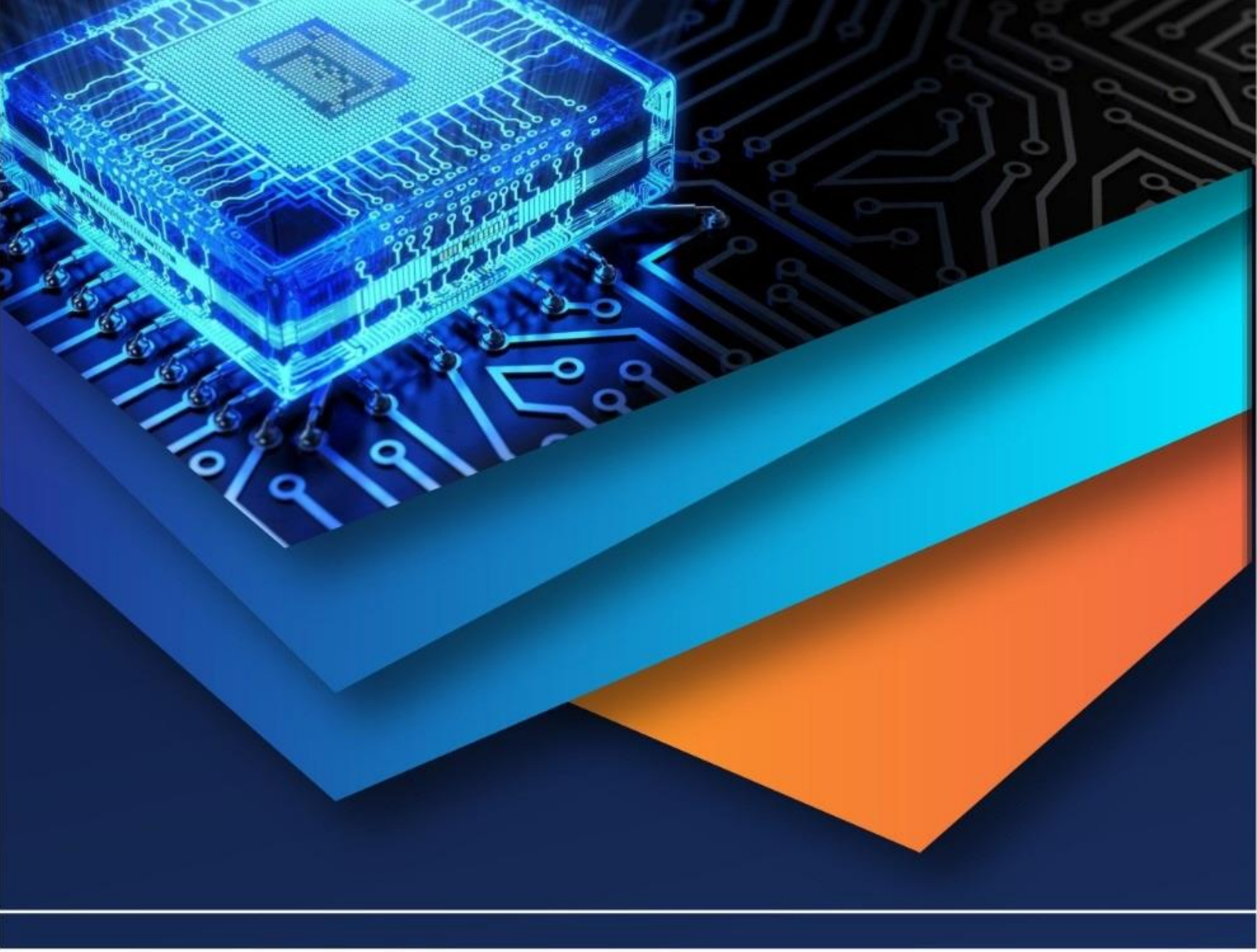
Figure 11 Validation loss for 1DCNN on NSL-KDD dataset using all features

VI. CONCLUSION

The efficiency of IDS to capture network anomalies and threats can be mitigated using deep learning models and machine learning models. The ability of the model to capture intrusions in the network relies on selecting the relevant feature subset and optimal parameters of the model. Using two real time dataset, the performance of proposed 1DCNN deep learning model is compared with bagged trees with feature selection. The experimental results showed that the proposed 1DCNN model with all the features achieved higher performance when compared with bagged tree model with feature selection on CICIDS2017 dataset and NSL-KDD dataset. Different combination of feature subsets should be explored with deep learning models and machine learning models to detect abnormal traffic behavior. As a future work, the proposed deep learning model will be evaluated for multi-class problems on different dataset with varying features and instances. Varying dataset and features could potentially help to understand the generalization of the models towards newer or unknown threats.

REFERENCES

[1] Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761-768.
[2] Disha, R.A., Waheed, S. Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 5, 1 (2022).
[3] Henry, A., Gautam, S., Khanna, S., Rabie, K., Shongwe, T., Bhattacharya, P., ... & Chowdhury, S. (2023). Composition of hybrid deep learning model and feature optimization for intrusion detection system. Sensors, 23(2), 890.
[4] Ho, S., Al Jufout, S., Dajani, K., & Mozumdar, M. (2021). A novel intrusion detection model for detecting known and innovative cyberattacks using convolutional neural network. IEEE Open Journal of the Computer Society, 2, 14-25.
[5] Injadat, M., Moubayed, A., Nassif, A. B., & Shami, A. (2020). Multi-stage optimized machine learning framework for network intrusion detection. IEEE Transactions on Network and Service Management, 18(2), 1803-1816.
[6] Lin, C. Y., Chen, B., & Lan, W. (2022, January). An efficient approach for encrypted traffic classification using CNN and bidirectional GRU. In 2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE) (pp. 368-373).
[7] Louati, Faten, and Farah Barika Ktata. "A deep learning-based multi-agent system for intrusion detection." SN Applied Sciences 2.4 (2020): 675.
[8] NG, B. A., & Selvakumar, S. (2020). Anomaly detection framework for Internet of things traffic using vector convolutional deep learning approach in fog environment. Future Generation Computer Systems, 113, 255-265.
[9] Priyavengatesh, M., & Kannan, R. (2022). An Efficient Intrusion Detection System Using Machine Learning Model. Journal of Algebraic Statistics, 13(3), 4984-5002.
[10] Sahu, S. K., Mokhade, A., & Bokde, N. D. (2023). An Overview of Machine Learning, Deep Learning, and Reinforcement Learning-Based Techniques in Quantitative Finance: Recent Progress and Challenges. Applied Sciences, 13(3), 1956.
[11] Sharma, B., Sharma, L., Lal, C., & Roy, S. (2023). Anomaly based network intrusion detection for IoT attacks using deep learning technique. Computers and Electrical Engineering, 107, 108626.
[12] Sultana, N., Chilamkurti, N., Peng, W., & Alhadad, R. (2019). Survey on SDN based network intrusion detection system using machine learning approaches. Peer-to-Peer Networking and Applications, 12, 493-501.
[13] Talukder, M. A., Hasan, K. F., Islam, M. M., Uddin, M. A., Akhter, A., Yousuf, M. A., ... & Moni, M. A. (2023). A dependable hybrid machine learning model for network intrusion detection. Journal of Information Security and Applications, 72, 103405.
[14] Vengatesh, M. P., & Kannan, R. (2023). Network intrusion detection system using deep learning models to capture cyber attacks. Journal of Data Acquisition and Processing, 38(2), 175.
[15] Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access, 5, 21954-21961.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY