DOI: https://doi.org/10.24113/ijoscience.v6i7.317

Convolutional Neural Network based Intelligent Network Intrusion Detection System

Levina Bisen
M.Tech. Scholar
Department of CSE, VITS
Bhopal (M.P), India
levisbisen@gmail.com

Sumit Sharma
Professor
Department of CSE, VITS
Bhopal (M.P), India
sumit_sharma782022@yahoo.co.in

***Abstract*— Today cyberspace is developing tremendously, and the Intrusion Detection System (IDS) plays a key role in information security. The IDS, which operates at the network and host levels, should be able to identify various malicious attacks. The job of network-based IDSs is to distinguish between normal and malicious traffic data and trigger an alert in the event of an attack. In addition to traditional signature-based and anomaly-based approaches, many researchers have used various deep learning (DL) techniques to detect intruders, as DL models are capable of automatically extracting salient features from the input data packets. The application of the Convolutional Neural Network (CNN), which is often used to solve research problems in the visual and visual fields, is not much explored for IDS. In this research work the proposed model for intrusion detection is based on feature selection and reduction using CNN and classification using random forest. As compared to some existing work the proposed algorithm proves its efficiency in terms of high accuracy and high detection rate.**

***Keywords*— Intrusion Detection System, Deep Learning, Convolution neural network, Random Forest, NSL-KDD, UNSW-NB 15.**

I. INTRODUCTION

Intrusion Detection System (IDS) are implemented over host computer or network as a security tool or application to avoid malicious attacks over them. IDS is implemented as individual host based or network based. The function of host-based IDS is to detect attack over a single computer or host computer. But when IDS is applied over multiple systems, connected in a network, then it is termed as Network Intrusion Detection System (NIDS). In NIDS, the intrusions are detected and analyzed over network traffic and it is installed over network gateway to capture data packets and analyze its behavior. Host based IDS is categorized in four types [1][2], File System Monitors, Log file analyzers, Connection analyzers, Kernel-based IDS.

Similarly, NIDS is mainly categorized into two types, such as Signature-based and Anomaly based. In signature-based NIDS, the data packets are analyzed and their behavior or signature is stored in database. In such type of NIDS, only known attacks are detected as their signature is stored in database. But in anomaly-based NIDS system, the behavior of data packets is analyzed as how much it is deviated from normal behavior and are capable to detecting new attacks [3]. Ryan et al. [3] proposed anomaly detection model by using back-propagation 3-layer Multi-Layer Perceptron (MLP) to detect possible attack in network. This model analyzed each session logs and analyzed the behavior of data packets. The MLP model analyzed 22/24 anomaly cases correctly.

Ghosh et al. [4] proposed a similar model as [3]. In this model, the prediction of coming data packets are analyzed by generalized study of previous known packets. For analysis this model is designed by applying artificial neural networks (ANNs) in order to detect malicious behavior of coming network traffic. Similar approach was applied in [5] and [6] using Self-Organizing Maps (SOM). These models are trained on the basis of previously recognized packets and tested over real-time data packets or network traffic.

Meng et al. [8] analyzed network anomaly by using artificial neural network, support vector machine and decision tree machine learning approach. The performance of decision tree gives better result. Decision tree also detects U2R and R2L attacks with high efficiency as such attacks occurs with low frequency.

Feng et al. [9] integrated SVM and Self-Organized Ant Colony Network for intrusion detection. This model is hybrid by merging classification and clustering techniques.

Manjula et al. [10] proposed a predictive intrusion detection tool using machine learning classification algorithms such as Logistic Regression, Gaussian Naive Bayes, Support Vector Machine and Random Forest. Out of all random forest gives highest accuracy rate.

Saad Mohamed et al. [11] presented a hybrid approach to anomaly detection using of K-means clustering and Sequential Minimal Optimization (SMO) classification.

Shone et al. [12] proposes network-based intrusion detection system that used the nonsymmetric deep autoencoder (NDAE) for feature learning and anomaly

detection. The proposed model was analyzed over KDD-99 and NSL-KDD datasets. The proposed model gives more accurate result on KDD-99 dataset. But in this paper, false alarm rate of U2R is 50 which is very high and requires more training time and samples for improved detection accuracy.

Vinayakumar et al. [13] proposed a hybrid intrusion detection system based on a deep neural network (DNN) is explored to develop a flexible and effective IDS to detect and classify unforeseen and unpredictable cyberattacks. DNN model learns the abstract and high-dimensional feature representation of the IDS data.

II. CNN ARCHITECTURE

The CNN is trained using the backpropagation mechanism. To combat the issues of fully connected neural network, convolutional neural networks (CNNs) were created and will be used as the foundation of this work. The key property of CNNs is that they are not fully connected, where every node in a layer is connected to every node in the previous layer. There are four main elements to a CNN [14]:

- Convolutional layer
- Batch Normalization
- Max Pooling layer
- Fully connected layer with Rectified Linear Unit

Convolution: The feature set generated according to the network is reduced by the convolutional operations.

Batch Normalization: To speeds up the training process this layer is added.

Max Pooling Layer: The pooling layer is used only to reduce the dimensionality of the previous level so that it is more suitable for the next level of the network. A CNN can have an unlimited number of convolutional and pooling levels in any order, the only limit is the power and time of calculation and the risk of overfitting of network.

Rectified Linear Unit: Rectified Linear Unit (ReLU) are used in many CNN architectures as an activation function for the network. In this activation function, the negative co-efficient are replaced with zero value which is represented by the local features of the input image. The function is represented as:

$$f(x) = \begin{Bmatrix} 0 \; for \; x < 0 \\ x \; for \; x \geq 0 \end{Bmatrix} \quad (1)$$

Some of the neurons dropped because they do not contribute to forward passage and do not participate in backpropagation. Every time an input is presented, the neural network analyzes another architecture, but all these architectures share a common weight. This technique reduces the complex adaptations of neurons because a neuron cannot rely on the presence of some other neurons.

Fully-Connected Layer: The fully connected layer of CNN is a normal neural network and is generally used as a last step in a convolutional network. Generally, it is used for classification purpose where the desired output is an array of elements m.

III. PROPOSED MODEL

This work proposes a unique CNN-RF architecture for NIDS and HIDS composed of an input layer, hidden layers and an output layer. The hierarchical layers in the CNN facilitate to extract highly complex features and do better pattern recognition capabilities in IDS data. Each layer estimates non-linear features that are passed to the next layer and the last layer in the CNN passes the feature vector to random forest to performs multi-classification. The proposed system architecture is shown in figure 1.

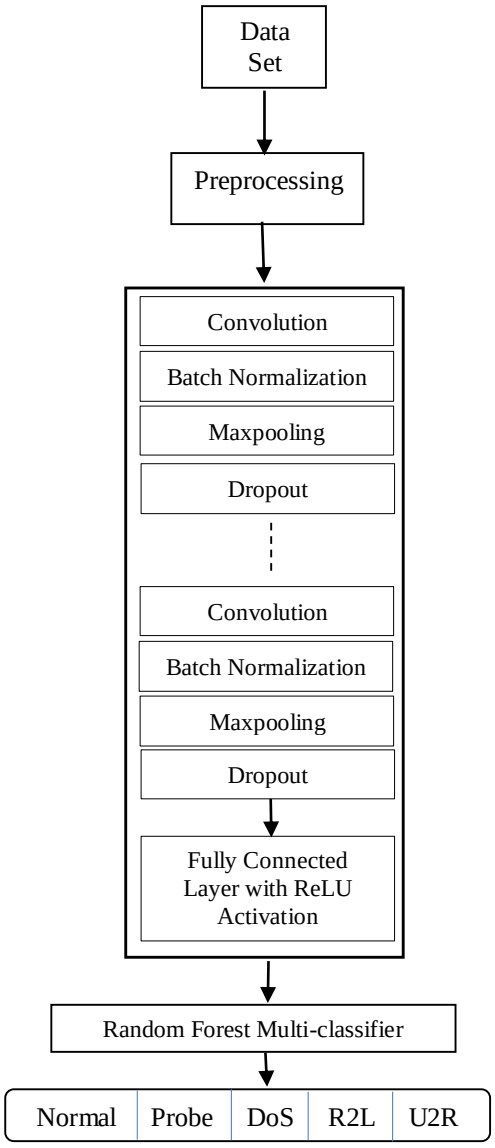

Figure 1: Proposed Flow Diagram of Intrusion Detection System

An approx. 10% of NSL-KDD dataset is used to train the proposed model. If the entire dataset is used for training the

proposed model, then several problems can occur. The most important ones are the training time and a system goes to a single point of failure because of memory over-flow. The algorithm flow of the proposed method is described as follows:

A. Preprocessing

This stage purpose is to preprocess the database file in which there is conversion of symbolic attributes protocol, service, and flag in numerical is done. Further data is normalized. The purpose of statistical normalization is to convert data derived from any Normal distribution into standard Normal distribution with mean zero and unit variance. The statistical normalization is defined as:

$$\mathrm{Data_i} = \frac{\mathrm{x_i} - \mu}{\sigma} \tag{2}$$

Where , xi = original data of the feature or attribute

μ= mean of data value

$$\mu = \frac{1}{\mathrm{n}}\sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{x_i} \tag{3}$$

σ=standard deviation

$$\sigma = \sqrt{\frac{1}{\mathrm{n}}\sum_{\mathrm{i=1}}^{\mathrm{n}} (\mathrm{x_i} - \mu)} \tag{4}$$

However, using statistical normalization, the data set should follow a Normal distribution, that is, the number of sample n should be large according to central limit theorem. The statistical normalization does not scale the value of the attribute into [0,1].

B. Post-Processing Phase

The optimal features are extracted using the proposed CNN model. The CNN model contains the convolution 1D layer which uses a one-dimensional filter that slides over the connection record in order to form a feature map. This feature map, in turn, is passed into a max-pooling layer which facilitates the dimensionality reduction. The batch normalization process is employed between the convolution and max-pooling layer to speeds up the training process and also for performance enhancement. Dropout is placed after the max-pooling layer which acts as a regularization term. Since CNN has parameters, the hyperparameter tuning approach is followed to identify the optimal parameters. The value 0.01 is assigned as the learning rate and adam optimizer is utilized. The number of filters is 32 in the initial CNN layer, 64 in the next CNN layer and 128 in the final CNN layer. The parameter max-pooling length is set to 2 in all the max-pooling layers and dropout to 0.01. When the number of CNN layers increased from 3 to 4, the performance decreased and hence 3 level CNN is used. Finally, two dense layers are included along with the CNN layer and the first dense layer composed of 512 neurons and the second one is composed of 128 neurons. These layers use ReLU as the activation function.

C. Classification

The classification is done after the fully connected layer with random forest multiclassification layer.

IV. RESULT ANALYSIS

The performance of proposed methodology is evaluated on the basis of following parameters:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} * 100 \tag{5}$$

$$Precision = \frac{(TP)}{(TP + FP)} * 100 \tag{6}$$

$$Recall = \frac{(TP)}{(TP + FN)} * 100 \tag{7}$$

$$F\_Measure = \frac{2 * Precision * Recall}{(Precision + Recall)} * 100 \tag{8}$$

Where,

True Positive (TP) = If predicted and actual data packet, both are anomaly in nature.

True Negative (TN) = If predicted and actual nature of data packet, both, are not anomaly.

False Positive (FP) = If predicted nature of data packet is anomaly but actual nature of data packet is normal.

False Negative (FN) = If predicted nature of data packet is normal but actual nature of data packet is anomaly.

Table 2-3 shows the performance evaluation of multilevel classification algorithm over NSL-KDD dataset. From the result analysis it has been analyzed that performance rate of multilevel ensemble SVM classification achieved best result. Table 4 shows the performance evaluation of multilevel classification algorithm over UNSW-NB 15 Dataset. Different test samples are taken to analyze performance on this dataset and proposed algorithm also outperforms better.

Table 2: Performance Evaluation on NSL-KDD Dataset

Performance	Accuracy	Precision	Recall	F_Measure
Sample 1	89.91	89.44	89.64	89.53989
Sample 2	89.79	88.62	89.8	89.2061
Sample 3	89.82	88.39	88.88	88.63432

Sample 4	88.78	88.52	88.6	88.55998
Sample 5	89.79	88.78	89.28	89.0293
Average	89.618	88.75	89.24	88.99392
Kumar et al. [13]	78.5	81.0	78.5	76.5

Table 4: Performance Evaluation on UNSW-NB 15 Dataset

Performance	Accuracy	Precision	Recall	F_Measure
TestSet-1	76.84	80	72.37	75.99
TestSet-2	75.66	80.1	78.79	79.43
TestSet-3	77.58	80.2	72.7	76.26
TestSet-4	77.9	81	74.11	77.40
TestSet-5	78.61	79.21	76	77.57
Average	77.31	80.10	74.79	77.33
Kumar et al. [13]	58.1	58.6	58.1	49.6

VII. CONCLUSION

In this paper, the effectiveness of the CNN model is studied for intrusion detection by modeling the network traffic data. The proposed CNN-RF outperforms the other relevant approaches where models like DNN are used. The simulation results are performed on NSL-KDD dataset as well UNSW-NB 15 dataset and result shows accuracy improvement of approximate 15% on NSL-KDD dataset whereas it shows approximately 33% improvement on UNSW-NB15 dataset.

REFERENCES

[1] De Boer, P., Pels, M, “Host-Based Intrusion Detection Systems”, Amsterdam University, Amsterdam, 2005.

[2] Garcia-Teodoro, P., “Anomaly-based network intrusion detection: techniques”, systems and challenges. Comput. Security Vol. 28.Issue, pp. 18–28, 2009.

[3] J. Ryan, M. Lin, and R. Miikkulainen, “Intrusion Detection with Neural Networks,” Conference in Neural Information Processing Systems, 943–949.

[4] A. K. Ghosh and A. Schwartzbard, “A Study in Using Neural Networks for Anomaly and Misuse Detection,” Conference on USENIX Security Symposium, Volume 8, pp. 12–12, 1999.

[5] P. L. Nur, A. N. Zincir-heywood, and M. I. Heywood, “Host-Based Intrusion Detection Using Self-Organizing Maps,” in Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 1714–1719, 2002.

[6] K. Labib and R. Vemuri, “NSOM: A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps,” 2000.

[7] Sharma, R.K., Kalita, H.K., Issac, B., “Different firewall techniques: a survey”, International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2014.

[8] Meng, Y.-X., “The practice on using machine learning for network anomaly intrusion detection”, International Conference on Machine Learning and Cybernetics (ICMLC), Vol. 2, IEEE, 2011.

[9] Feng, W., “Mining network data for intrusion detection through combining SVMs with ant colony networks”, Future Generation Computer, System, Vol. 37, pp. 127–140, 2014.

[10] Manjula C. Belavagi and Balachandra Muniyal, “Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection, Procedia Computer Science”, Elsevier, 2016.

[11] Saad Mohamed Ali Mohamed Gadal and Rania A. Mokhtar, “Anomaly Detection Approach using Hybrid Algorithm of Data Mining Technique”, International Conference on Communication, Control, Computing and Electronics Engineering, IEEE, 2017.

[12] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," in IEEE Transactions on Emerging Topics in Computational Intelligence, Vol. 2, No. 1, pp. 41-50, Feb. 2018.

[13] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," in IEEE Access, Vol. 7, pp. 41525-41550, 2019.

[14] M. Ishaque and L. Hudec, "Feature extraction using Deep Learning for Intrusion Detection System," 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 2019, pp. 1-5.