# DCNN-IDS: Deep Convolutional Neural Network Based Intrusion Detection System

S. Sriram[1(✉)], A. Shashank[1], R. Vinayakumar[1,2], and K. P. Soman[1]

[1] Center for Computational Engineering and Networking, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
sri27395ram@gmail.com, vinayakumarr77@gmail.com

[2] Division of Biomedical Informatics, Cincinnati Children's Hospital Medical Centre, Cincinnati, OH, USA
Vinayakumar.Ravi@cchmc.org

**Abstract.** In the present era, cyberspace is growing tremendously and the intrusion detection system (IDS) plays a key role in it to ensure information security. The IDS, which works in network and host level, should be capable of identifying various malicious attacks. The job of network-based IDS is to differentiate between normal and malicious traffic data and raise an alert in case of an attack. Apart from the traditional signature and anomaly-based approaches, many researchers have employed various deep learning (DL) techniques for detecting intrusion as DL models are capable of extracting salient features automatically from the input data. The application of deep convolutional neural network (DCNN), which is utilized quite often for solving research problems in image processing and vision fields, is not explored much for IDS. In this paper, a DCNN architecture for IDS which is trained on KDDCUP 99 data set is proposed. This work also shows that the DCNN-IDS model performs superior when compared with other existing works.

**Keywords:** Intrusion detection · Deep learning · Convolutional neural network · Cyber security

## 1 Introduction

Information Technology (IT) systems play a key role in handling several sensitive user data that are prone to several external and internal intruder attacks [1]. Every day, the attackers are coming up with new sophisticated attacks and the attacks against IT systems are growing as the internet grows. As a result, a novel, reliable and flexible IDS is necessary to handle the security threats like malware attacks which could compromise a network of systems that can be used by the attackers to perform various attacks using command and control servers. Though there are various other security systems like firewall, IDS plays a major role in defending the network from all kinds of cyberattacks. IDS is divided into


S. Balusamy et al. (Eds.): ICC3 2019, CCIS 1213, pp. 85–92, 2020.
https://doi.org/10.1007/978-981-15-9700-8_7

two categories. The first one is network IDS (NIDS) which monitors the network traffic and raises alerts when it detects any kind of attack. The second one is host-based IDS (HIDS) which detects both internal and external intrusion and misuse by monitoring the system in which it is installed. It constantly records the user activities and alerts the designated authority in case of an attack. Both IDS are represented in Fig. 1.

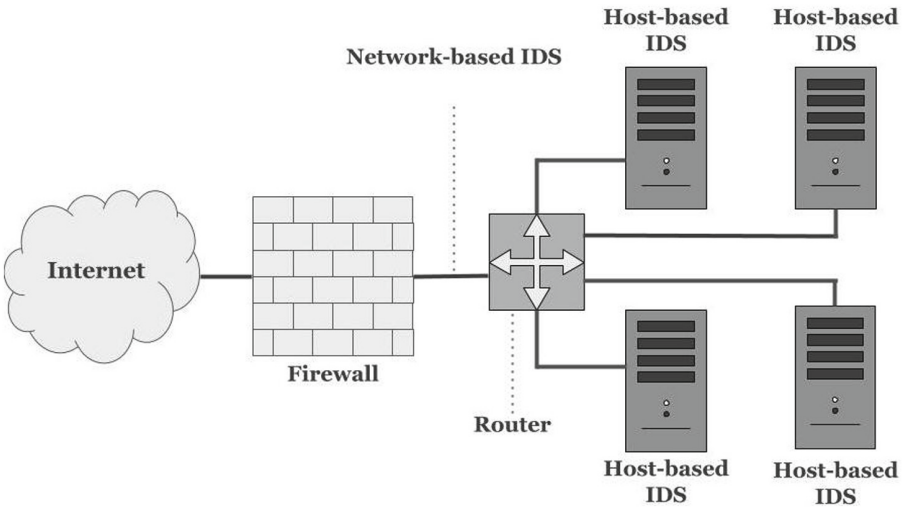


**Fig. 1.** Model of IDS

The job of NIDS is to monitor the network traffic and to identify whether the network traffic records as either malicious or normal (benign). Several machine learning (ML) and deep learning (DL) classifiers are widely employed for the detection of intrusion as it is a classification problem. DL models like autoencoders (AE), recurrent structures, deep neural network (DNN), etc. are used for IDS by many researchers. The convolutional neural network (CNN) model is quite often utilized for solving research problems in fields like computer vision, image processing, etc. due to its capability to extract location invariant features automatically. The application of CNN for IDS is not explored much. Therefore, in this paper, deep CNN (DCNN) is trained on the most popular benchmark data set called KDDCup 99 which has more than 8,00,000 data points. It is also shown that the DCNN-IDS gives superior outcomes when compared to previous works. Further, this paper is arranged as follows. Sections 2 and 3 includes the related works and data set description. Sections 4 and 5 describes the statistical measures and the proposed model respectively. Sections 6 and 7 covers the results and conclusion.

## 2 Related Works

Several ML based approaches are proposed for IDS. [2] analyses several ML based approaches for intrusion detection for identifying various issues. Issues related to the detection of low-frequency attacks are discussed with possible solutions to improve the performance further. The disadvantage of ML based approach is that ML models operate on manual features extracted by the domain expert. Since DL models can extract relevant features automatically without human intervention, many researchers propose various DL based solution for IDS. Self-Taught learning based NIDS is proposed in [3], where a sparse autoencoder and softmax regression is used. The proposed model is trained on the NSLKDD data set and it achieves an accuracy around 79.10% for 5-class classification which is very close to the performance of existing models. Apart from this, 23-class and 2-class classification also achieved good performance. A recent study [4] claims that the deep networks perform better than shallow networks for IDS as the deep network is capable of learning salient features by mapping the input through various layers. In [5], the performance of RNN based NIDS is studied. The model is trained on the NSL-KDD data set and both multi-class and binary classification are performed. The performance of RNN based IDS is far superior in both classification when compared to other traditional approaches and the author claims that RNN based IDS has strong modeling capabilities for IDS. Similarly in [6] and [7], various recurrent structures are proposed for IDS.

In [8], a new stacked non-symmetric deep autoencoder (NDAE) based NIDS is proposed. The model is trained on both KDDCUP and NSLKDD benchmark data sets and its performance is compared with DBN based model. It can be observed from the experimental analysis that the NDAE based approach improves the accuracy up to 5% with 98.8% training time reduction when compared to DBN based approach. In [9], the effectiveness of CNN and hybrid CNN recurrent structures are studied and it can be observed that CNN based model outperforms hybrid CNN-RNN models. In [10], the authors have claimed that analyzing the traffic features from the network as a time series improves the performance of IDS. They substantiate the claim by training long short-term memory (LSTM) models with KDDCUP data set with a full and minimal feature set for 1000 epochs and have obtained a maximum accuracy of 93.82%. In [11], a scalable DL framework is proposed for intrusion detection at both the network and host levels. various ML and DNN models are trained on data sets such as KDDCUP, NSLKDD, WSN-DS, UNSW-NB15, CICIDS 2017, ADFA-LD and ADFA-WD and their performance are compared. In this work, the effectiveness of the proposed model is evaluated using standard performance metrics and it is compared with other works such as [10] and [11].

## 3 Data Set Description

The tcpdump data of the 1998 DARPA intrusion detection evaluation data set is pre-processed to build KDDCUP 99 data set. The feature extraction from

tcpdump data is facilitated by the MADMAID data mining framework [11]. Table 1 represents the statistical information about the data set. This data set was built by capturing network traffic for ten weeks from thousands of UNIX systems and hundreds of users accessing those systems in the MIT Lincon laboratory. The data captured during the first 7 weeks were utilized for training purpose and the last 3 weeks data were utilized for testing purposes.

This data set has a total of 5 classes and 41 features. The first one is the normal class which denotes benign network traffic records. The second one is DoS. It is a kind of attack that works against resource availability. The third one is the probing attack. This class represents all attacks that are used by the attackers to obtain detailed information about the system and its security structures and configurations. This kind of attack is performed by the attacks initially in order to gain insights about the network so that they could perform many critical attacks later. The next one is R2L which denotes root to local attacks. This kind of attack is performed in order to acquire illegal remote access to any system in a network. The last one is U2R which is user to root attacks. It represents attacks that are using to gain root-level access to a system.

**Table 1.** Statistics of KDDCUP 99 data set

| Attack types | Description | KDDCUP 99 (10% of Data) | |
|---|---|---|---|
| | | Train | Test |
| Normal | It denotes normal traffic records | 97,278 | 60,593 |
| DoS | Attacker works against the resource availability | 3,91,458 | 2,29,853 |
| Probe | Obtaining detailed statistics of system and network configuration details | 4,107 | 4,166 |
| R2L | Illegal access originated from remote computer | 1,126 | 16,189 |
| U2R | Obtaining root or superuser level permissions illegally on a particular system | 52 | 228 |
| Total | | 4,94,021 | 3,11,029 |

## 4 Statistical Measures

The proposed DCNN-IDS model is evaluated using some of the most commonly used metrics such as recall, precision, f1-score, and accuracy. The Error matrix gives an overall idea about the performance of the model and These metrics are computed using terms that can be found in the error matrix. The first one is True Positive (TP) which indicates the count of malicious traffic data points

that are rightly considered as malicious by the model. The second one is False Positive (FP) which indicates the count of benign traffic data points that are wrongly considered as malicious by the model. Similarly, True Negative (TN) indicates the count of benign traffic data points that are rightly considered as benign by the model. False Negative (FN) is the final term that indicates the count of malicious traffic data points that are wrongly considered as benign by the model. Based on these four terms, we can define a number of metrics:

- **Accuracy:** This term denotes the total count of right predictions (TP and TN) made by the model over total count of all predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

- **Precision:** This term denotes the count of right positive results over the amount of all positive results predicted by the model.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

- **Recall:** This term points to the total count of right positive results over the total count of all samples that are relevant.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

- **F1-score:** This term represents both recall and precision by taking subcontrary mean between them.

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

## 5 Proposed Model

The DCNN-IDS architecture is represented by the Fig. 2 The structure of the DCNN-IDS model is shown in Table 2. The proposed architecture is composed of the following sections

- **Pre-processing of network connection records**: the symbolic data in the connection records are transformed into numeric and normalized the data using L2 normalization.
- **Feature generation**: 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

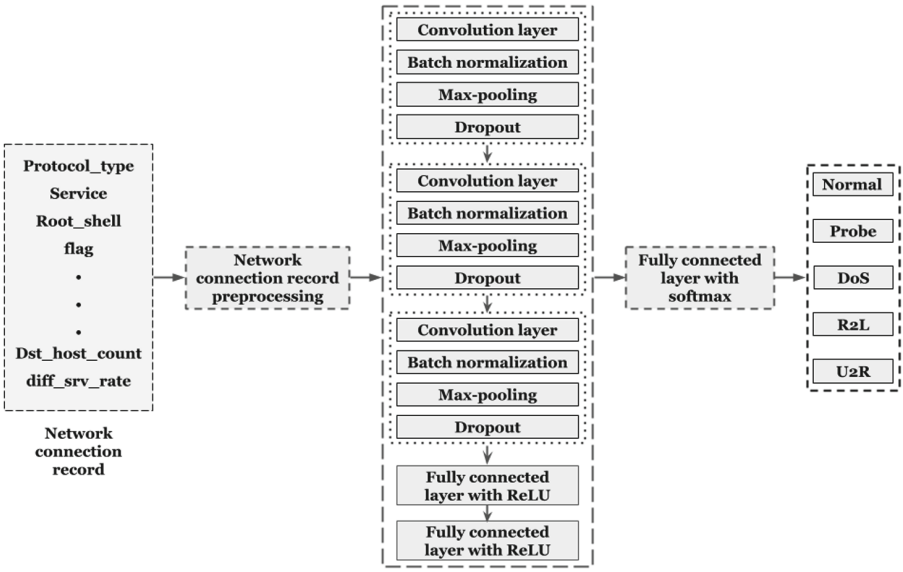


**Fig. 2.** Architecture of DCNN-IDS

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.

– **Classification**: The classification is done using the fully connected layer which composed of 5 neurons with a softmax activation function.

## 6 Results

The proposed CNN model is designed and trained using one of the most commonly used python 3 library called Keras[1] with tensorflow[2]. The model performance is tested on the KDDCup 99 data set and the obtained results are tabulated in Table 3. The proposed CNN model outperforms than the existing LSTM [10] and DNN [11] based intrusion detection models.

[1] https://keras.io.
[2] https://www.tensorflow.org.

**Table 2.** Details about the structure proposed model

| Layer type | Output shape | Parameters # |
|---|---|---|
| 1D Convolution | ( – , 41, 32) | 128 |
| Batch normalization | ( – , 41, 32) | 128 |
| Max Pooling | ( – , 21, 32) | – |
| Dropout | ( – , 20, 32) | – |
| 1D Convolution | ( – , 20, 64) | 6,208 |
| Batch normalization | ( – , 20, 64) | 256 |
| Max Pooling | ( – , 10, 64) | – |
| Dropout | ( – , 10, 64) | – |
| 1D Convolution | ( – , 10, 128) | 24,704 |
| Batch normalization | ( – , 10, 128) | 512 |
| Max Pooling | ( – , 5, 128) | – |
| Dropout | ( – , 5, 128) | – |
| Flatten | ( – , 640) | – |
| Dense | ( – , 512) | 3,28,192 |
| Dropout | ( – , 512) | – |
| Dense | ( – , 128) | 65,664 |
| Dropout | ( – , 128) | – |
| Dense | ( – , 5) | 645 |
| Total parameters: 426,437 | | |

**Table 3.** Evaluation of DL models on test set

| Architecture | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| LSTM [10] | 93.82 | 82.8 | 58.3 | 68.4 |
| DNN [11] | 93.5 | 92 | 93.5 | 92.5 |
| CNN (Proposed method) | **94.1** | **92.4** | **94.1** | **93** |

## 7 Conclusion

In this paper, the effectiveness of the deep CNN model is studied for intrusion detection by modeling the network traffic data. The proposed 1D-CNN outperforms the other relevant approaches where models like DNN and LSTM are used. The proposed model uses only 425,989 parameters and does not incorporate any complicated prepossessing techniques. Therefore, it has the potential to be used in various low-powered IoT devices which has a very limited computation power. In the future, hybrid models can be used where the features are extracted from hidden layers of DL models and fed into other ML or DL models for further improvement of performance.

**Acknowledgement.** This work was in part supported by Paramount Computer Systems and Lakhshya Cyber Security Labs. We are grateful to NVIDIA India, for the GPU hardware support to the research grant. We are also grateful to the center of Computational Engineering and Networking, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore for encouraging the research.

## References

1. Mukherjee, B., Heberlein, L.T., Levitt, K.N.: Network intrusion detection. IEEE Netw. **8**(3), 26–41 (1994)
2. Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E.S.: A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Commun. Surv. Tutorials **21**(1), 686–728 (2018)
3. Javaid, A., Niyaz, Q., Sun, W., Alam, M.: A deep learning approach for network intrusion detection system. In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), pp. 21–26. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering) (2016)
4. Hodo, E., Bellekens, X., Hamilton, A., Tachtatzis, C., Atkinson, R.: Shallow and deep networks intrusion detection system: a taxonomy and survey. arXiv preprint arXiv:1701.02145 (2017)
5. Yin, C., Zhu, Y., Fei, J., He, X.: A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access **5**, 21954–21961 (2017)
6. Vinayakumar, R., Soman, K.P., Poornachandran, P.: A comparative analysis of deep learning approaches for network intrusion detection systems (N-IDSs): deep learning for N-IDSs. Int. J. Digit. Crime Forensics (IJDCF) **11**(3), 65–89 (2019)
7. Vinayakumar, R., Soman, K.P., Poornachandran, P.: Evaluation of recurrent neural network and its variants for intrusion detection system (IDS). Int. J. Inf. Syst. Model. Des. (IJISMD) **8**(3), 43–63 (2017)
8. Shone, N., Ngoc, T.N., Phai, V.D., Shi, Q.: A deep learning approach to network intrusion detection. IEEE Trans. Emerg. Top. Comput. Intell. **2**(1), 41–50 (2018)
9. Vinayakumar, R., Soman, K.P., Poornachandran, P.: Applying convolutional neural network for network intrusion detection. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1222–1228. IEEE (2017)
10. Staudemeyer, R.C.: Applying long short-term memory recurrent neural networks to intrusion detection. S. Afr. Comput. J. **56**(1), 136–154 (2015)
11. Vinayakumar, R., Alazab, M., Soman, K.P., Poornachandran, P., Al-Nemrat, A., Venkatraman, S.: Deep learning approach for intelligent intrusion detection system. IEEE Access **7**, 41525–41550 (2019)
12. Vinayakumar, R., Soman, K.P., Poornachandran, P.: Evaluating effectiveness of shallow and deep networks to intrusion detection system. In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1282–1289. IEEE (2017)