Network Intrusion Detection using Deep Learning
(CNN)

PROJECT REVIEW-1

Submitted By:-

Yuvraj Kumar

Registration number: 18BIT0276

Submitted for the course : Information Security Analysis and Audit (CSE3501)

Team Members:

Aman Agarwal	**18BIT0256**
Shivansh Srivastava	**18BIT0324**
Yuvraj Kumar	**18BIT0276**

Slot: G2

Name of faculty:
SUMAIYA
THASEEN I

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

FALL SEMESTER – 2020-21

Paper Title 1: Using Convolutional Neural Networks to Network Intrusion Detection for Cyber Threats

Authors : **Year of publish: 2018**

Wen-Hui Lin1, Hsiao-Chung Lin , Ping Wang,

Bao-Hua Wu, Jeng-Ying Tsai

Technique/algorithm used and why it was chosen (motivation):

The technique / algorithm used here is CNN because the convolutional neural network (CNN) is a type of deep, feed-forward artificial neural networks which learn features that are concatenated with the original feature vectors and used for classification.

This algorithm was chosen because so as to improve classification accuracy in real-time detection, the CNN was regularly selected using the combinational convolution and pooling operations by several fully connected or sparsely connected layers followed by a final classification layer to determine the abstract weights of an input data.

Architecture/ model/pseudocode developed:

Diagrammatical representation of the architecture used here :

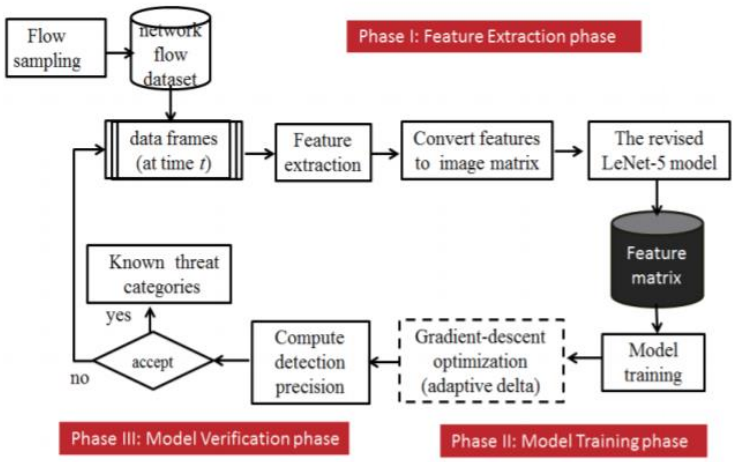


Figure 1. Basic concept of network intrusion detection by using LeNet-5 model.

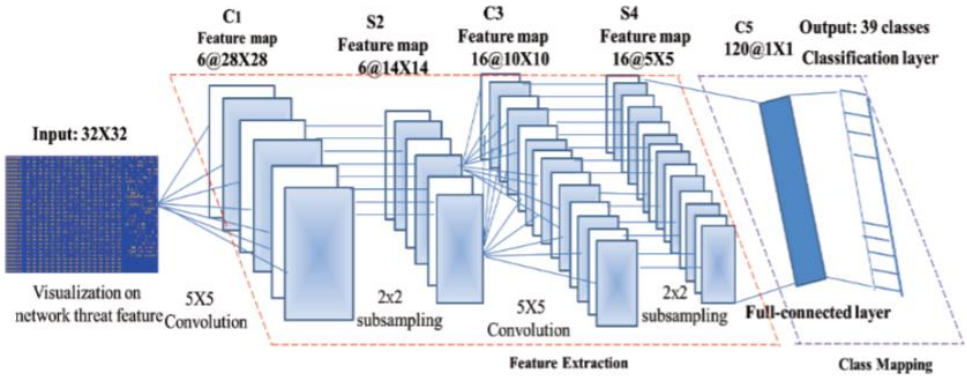


Figure 2. The revised LeNet-5 model for network threat classification

Model description:

Decision tree theory can be used as a feature reduction and increasing the speed of algorithm for NIS. The model consiss of following sub phases:

1) feature extraction: In this phase, training data were obtained from dtabase and the model was trainded

2) model learningphase: In model learning phase, the present study revised the LeNet-5 model designed by LeCun et al. in 1998

3) model verification phase: A cross validation approach was used

Datasets analyzed in the paper with the performance results:

The dataset used for the analysis for threat type is KDD cup’99 dataset with more than 10,000 records.

Performance results:

The model prediction accuracy is 99.65 % and the average accuracy is 95.41%.

Any comparison done with the previous techniques to specify that the proposed method is superior:

The previously proposed methods use to give abnormal outputs for some inputs.

Experimental results have shown that the classification error decreased as the size of the testing dataset increased.

Paper Title 2: An Improved Convolutional Neural Network model for Intrusion Detection in Networks

Authors: **Year of publish: 2019**

Riaz Ullah Khan, Xiaosong Zhang, Mamoun Alazab,

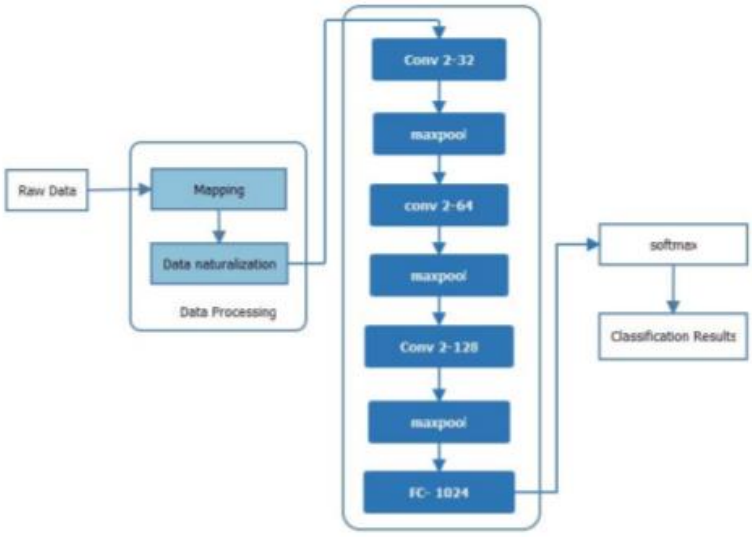
Rajesh Kumar

Technique/algorithm used and why it was chosen (motivation):

There are a lot many types of algorithm available in real world application like SVM etc. But every algorithm has their own advantages or disadvantages. So as to decrease the disadvantages, Peddabachigari proposed a hybrid intrusion detection model based on deep learning and verified that the model is more efficient than traditional machine learning methods.

Architecture/ model/pseudocode developed:

Model description:



Proposed Model for Intrusion Detection

MODEL DESIGN:

1) Architecture of Convolutional Neural Network: The architecture is composed of input , convulatonal layer, pooled layer, fully connected layer and output layer.

Assuming that the input feature of the convolutional neural network is X, and the feature map of the i-th layer is , the convolution process can be expressed as:

$$M_i = f(M_{i-1} \otimes w_i + b_i)$$

2) Intrusion Detection Model Framework: Framework mainly consists of three steps:

a) Data preprocessing: Complete data is converted to structured form. Any missing data is filled

b) Training and Feature Extraction: Now CNN model is used for training purpose.

c) Softmax classifier is used to classify and get the classification results .

Datasets analyzed in the paper with the performance results:

The dataset used here is KDD99 data set. The data set obtained segregates network intrusion into 5 departments ie Normal, DOS, R2L, U2R and probing.

Performance results:

The experimental results how that this model improve the accuracy of himan intrusion detection and improve performance of human invading detection system. The accuracy abotained here is around 99.23%

Any comparison done with the previous techniques to specify that the proposed method is superior:

The previously used SVM gives around 98.20% accuracy, DBN gives 98.59% accuracy and the new improved CNN gives 99.23 % accuracy

Paper Title 3: Enhanced Network Intrusion Detection using Deep Convolutional Neural Networks.

Authors: **Year of publish : 2018**

Sheraz Naseer, Dr. Yasir Saleem

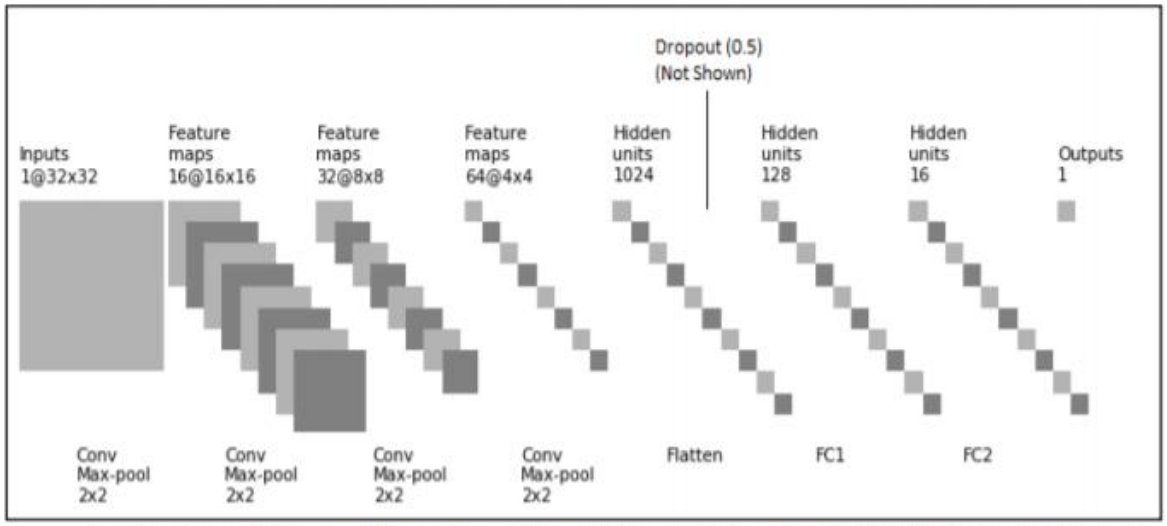
Technique/algorithm used and why it was chosen (motivation):

DCNN is used for IDS approach here.

This is used because earlier a four layer DNN with 100 units was used and this system gave 99% accuracy. But this approach is flawed, as given sufficient training, classifiers can be over-fitted to achieve such high rates.

Architecture/ model/pseudocode developed:

Proposed IDS approach uses a DCNN with an input layer, 3 pairs of conv-subsample layers, 3 fully connected layers and an output layer with one sigmoid unit.



Architecture of Proposed Deep Convolutional Neural Network (DCNN) for Intrusion Detection

Hyper-parameter selection method:

In process of deep learning, hyper parameters include such 'higher-level' properties of the model which cannot be learned from training set but have profound impact on learning capacity and accuracy of the model. Some hyper-parameters include learning rate of model, non-linearity, choice of objective function, regularization, parameter update method (optimizer), initial weight initializations, mini-batch size of input and number of training epochs to name a few. A learning algorithm $\mathcal{AA}$ maps $XXttttttttttt$from $GGxx$ to ff through optimization of hyper-parameters $\lambda\lambda$. The problem of determining good values for $\lambda\lambda$ is called Hyper-parameter optimization.

$$\lambda^{(*)} = argmin_{\lambda \in \Lambda} \mathcal{E} \sim G_x \left[\mathcal{L}(x; \mathcal{A}_\lambda(X^{train})) \right]$$

Datasets analyzed in the paper with the performance results:

Dataset used here is NSLKDD which has 41 features like its predecessor KDDCUP99.). Proposed DCNN model was trained using GPU on NSLKDD training dataset and evaluation of the same was performed on NSLKDDTest+ and NSLKDDTest21 datasets.

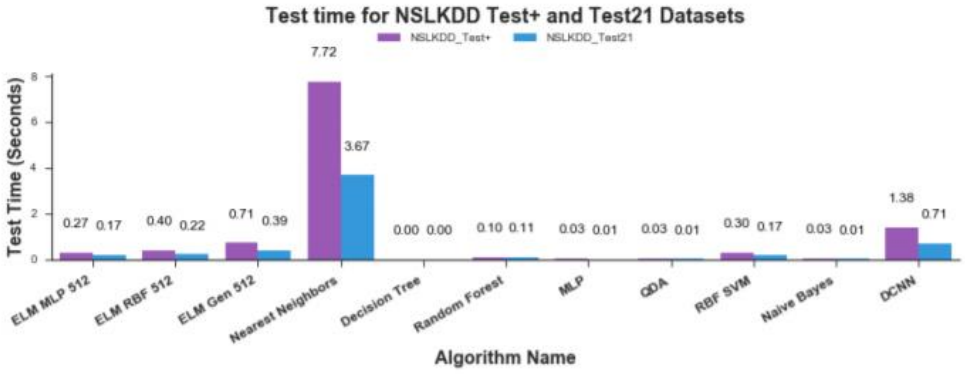
Performance results:

Accuracy, precision-recall-curve and mean Average precision. Proposed model achieved classification accuracy of 85.22 % and 69.56% for NSLKDDTest+ and NSLKDDTest21 respectively

Any comparison done with the previous techniques to specify that the proposed method is superior:

For evaluation of test Datasets, k-NN proved to be the most expensive algorithm and took approximately 8 and 4 seconds for NSLKDDTest+ and NSLKDDTest21 datasets respectively. DCNN took 1.39 and 0.71 seconds for evaluating above-mentioned test Datasets. The fastest evaluation was performed by Decision Tree algorithm, which took 5 and 3 milliseconds for evaluating test

datasets. Evaluation times for classification algorithms used in this study are shown below:



Paper Title 4: An Intrusion Detection Model based on a Convolutional Neural Network

Authors: **Year of publish: 2019**

Jiyeon Kim, Yulim Shin

Technique/algorithm used and why it was chosen (motivation):

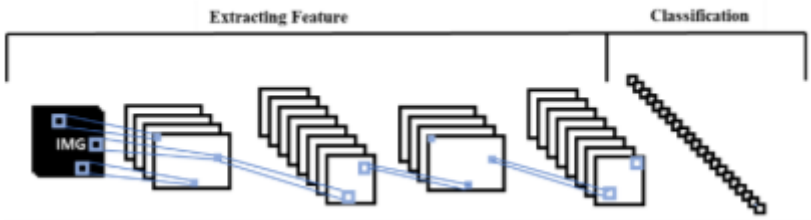
Here algorithm used is deep-learning techniques and a convolutional neural network (CNN) model for CSE-CIC-IDS 2018.

It was chosen so as to compare its working accuracy wit other algorithms like RNN etc and then improve performance of the model used here.

Architecture/ model/pseudocode developed:

Model description:

CNN model: A CNN model consists of convolutional layers, max-pooling layers, and a fully connected layer. Proper CNN model can be formed by adjusting the layers along modelling parameters like kernel size etc.



	Convolution Layer 1	Pooling Layer 1	Convolution Layer 2	Pooling Layer 2	Fully-connected Layer
Num. of kernels	16	-	32	-	-
Size of kernel	3*3	-	3*3	-	-
Size of pooling	-	2*2	-	2*2	-
Dropout	-	0.7	-	0.7	0.5

Our CNN model and parameters.

Datasets analyzed in the paper with the performance results:

CSE-CIC-IDS2018(CIC-2018) is a dataset which is used because it contains network traffic and system logs. It contains sub dataset of around 10 days with 16 different types of attack. Also it contains around 80 different types of features.

Performance results:

Results show that CNN model detects attacks in CIC-2018 with high accuracy, we still need to figure out a way of improving the accuracy of each attack. For instance, the accuracy of SD-3 is 0.9677. According to the confusion matrix of SD-3, however, the accuracies of 'DoS-GoldenEye' and 'DoSSlowloris' are 0.66 and 0.47 while the accuracy of 'benign' is 0.99.

Any comparison done with the previous techniques to specify that the proposed method is superior:

The accuracies of SD-2, SD-3, SD-5, and SD-9 with CNN are about 10% to 60% higher than that of using RNN.

Paper Title 5: An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks.

Authors: **Year of publish: 2019**

YIHAN XIAO , CHENG XING, TAINING ZHANG,

ZHONGKAI ZHAO

Technique/algorithm used and why it was chosen (motivation):

Here the model used is a network intrusion detection model based on a convolutional neural network–IDS (CNN–IDS).

It is chosen because Traditional intrusion detection algorithms typically employ mining association rules to identify intrusion behaviours. However, they fail to fully extract the characteristic information of user behaviours and encounter various problems, such as high false alarm rate (FAR), poor generalization capability, and poor timeliness.

Architecture/ model/pseudocode developed:

Model description:

The overall framework of the model presented in figure 3 consists of three steps:

1) Data preprocessing and data type conversion: The symbolic characteristic attributes in KDD datasets are digitized and normalized to obtain standardized datasets. After the standardized datasets undergo dimensionality reduction.
2) Concrete structure of the CNN intrusion detection model: Feature extraction is done here. Again classification is done using Softmax classifier.

3) Model train: Model training and reverse fine-tuning to improve the performance of the model is done. Back propagation (BP) algorithm fine-tunes the parameters of the network model.

Datasets analyzed in the paper with the performance results:

The dataset used here is KDDcup99 dataset which consists of 41 features, 38 of which are digital and 3 symbolic features.

Performance results:

The model used here, CNN-IDS model, AC,DR,FAR can reach 94%, 93% and 0.5%. Thus accuracy is 94%.

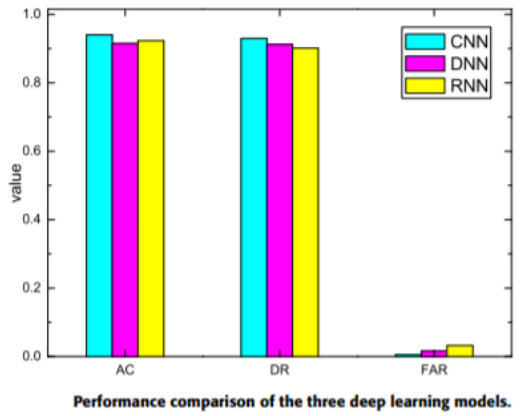
Any comparison done with the previous techniques to specify that the proposed method is superior:

Comparison is done between DNN,RNN and CNN which we used on the same dataset AE(100). The DNN sued three layer fully connected neural network. RNN first used dimensional transformation to divide 10x10 matrix into 10 time points and then input into RNN unit for training.

Performance Comparison :

Comparison time of three models.

time / model	training at every epoch	test
CNN–IDS	20s	11s
DNN	26s	15s
RNN	82s	124s



Performance comparison of the three deep learning models.

Comparison of each technique with other techniques (Tabular form):

<table>
<tr><th>S.No.</th><th>Title with year</th><th>Algorithm/Model used</th><th>Comparison with other algorithm
(Algorithm) (Accuracy %)</th></tr>
<tr><td>1</td><td>Using convolutional neural networks to network intrusion detection for cyber threats. 2018.</td><td>ID3 decision tree theory is used as a scheme of feature reduction for speeding up the learning of normal and intrusive pattern for NIDS.</td><td>The prediction accuracy of threat detection was higher with the CNN-based classifier than with the existing LeNet-5 model .
Accuracy without Back-Propagation ----------95

Accuracy with Back-Propagation----------99.65</td></tr>
<tr><td>2</td><td>An Improved Convolutional Neural Network Model for Intrusion Detection in Networks. 2019.</td><td>A hybrid intrusion detection model based on deep learning.</td><td>SVM----------98.20
DBN----------98.59
Improved CNN----------99.23</td></tr>
<tr><td>3</td><td>Enhanced Network Intrusion Detection using Deep Convolutional Neural Networks. 2018.</td><td>Proposed IDS approach uses a DCNN with an input layer, 3 pairs of conv-subsample layers, 3 fully connected layers and an output layer with one sigmoid unit.</td><td>
<table>
<tr><th>Classifier</th><th>Test+ (%)</th><th>Test21 (%)</th></tr>
<tr><td>DCNN(Proposed)</td><td>85.22</td><td>69.56</td></tr>
<tr><td>J48 [16]</td><td>81.05</td><td>63.97</td></tr>
<tr><td>Naïve Bays [16]</td><td>76.56</td><td>55.77</td></tr>
<tr><td>Random Forest[16]</td><td>80.67</td><td>63.25</td></tr>
<tr><td>MLP [16]</td><td>77.41</td><td>57.34</td></tr>
<tr><td>SVM [16]</td><td>69.52</td><td>42.29</td></tr>
</table>
<table>
<tr><th>Classifier</th><th>Test+ (%)</th><th>Test21 (%)</th></tr>
<tr><td>Random Tree [16]</td><td>82.02</td><td>66.16</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>NNRw1 [22]</td><td>82.41</td><td>67.06</td></tr>
<tr><td>NNRw2 [22]</td><td>84.12</td><td>68.82</td></tr>
<tr><td>Deep Autoencoder (AE) [25]</td><td>83.34</td><td>Not Reported</td></tr>
<tr><td>Denoising AE [41]</td><td>88.65</td><td>Not Reported</td></tr>
</table>
</td></tr>
<tr><td>4</td><td>An Intrusion Detection Model based on a Convolutional Neural Network. 2019.</td><td>Deploy two convolutional layers and the two maxpooling layers behind each convolutional layer.</td><td>CNN (especially-------10-60% more than SD-2, SD-3, SD-5 that of RNN. and SD-9)

RNN----------Between 81 to 98</td></tr>
</table>

5	An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks. 2019.	Enhanced way to use Softmax classifier.	**CNN-IDS----------92.5** **Logistic Regression----------88.2** **Decision Tree ----------91.26** **Random Forest ----------91.56** **SVM ----------92.25** **AdaBoost ----------89.68** **Naïve Bayes ----------82.67**

References:

[1]: Wen-Hui Lin1, Hsiao-Chung Lin , Ping Wang, Bao-Hua Wu, Jeng-Ying Tsai. Using convolutional neural networks to network intrusion detection for cyber threats. 2018.

(https://ieeexplore.ieee.org/document/8394474)

[2]: Riaz Ullah Khan, Xiaosong Zhang, Mamoun Alazab, Rajesh Kumar. An Improved Convolutional Neural Network Model for Intrusion Detection in Networks. 2019.

(https://ieeexplore.ieee.org/document/8854549)

[3]: Sheraz Naseer, Dr. Yasir Saleem. Enhanced Network Intrusion Detection using Deep Convolutional Neural Networks. 2018.

(https://www.researchgate.net/publication/329467386_Enhanced_Network_Intrusion_Detection_using_Deep_Convolutional_Neural_Networks)

[4]: Jiyeon Kim, Yulim Shin. An Intrusion Detection Model based on a Convolutional Neural Network. 2019.

(https://www.researchgate.net/publication/338455935_An_Intrusion_Detection_Model_based_on_a_Convolutional_Neural_Network)

[5]: YIHAN XIAO , CHENG XING, TAINING ZHANG, ZHONGKAI ZHAO. An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks.2019.

(https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8666014)

Yuvraj Kumar
(18BIT0276)

Student's Signature