**Introduction:**

Convolutional neural network in deep learning is a systematic detection model for arranging network flow and tracking down expected kinds of intrusion and it is a deep feed forward artificial neural network. CNN contains three parts: an input layer, a result layer and a hidden layer. The convolutional layer applies various convolutional filters or channels (kernels) to the image for each sub-region, the layer plays out a bunch of numerical tasks or mathematical operations with explicit spatial degree and step values, creating a solitary incentive for the output. The initiation capability then, at that point, plays out a non-straight activity by utilizing a few numerical Operations. The pooling layer performs a dimensionality decrease on the information to diminish the dimensionality of the feature image, holding the greatest value and disposing the rest. At last, the completely connected layer characterizes the highlights where every node in the layer is associated with every node in the past layer. [1]

Model of Intrusion Detection system positioned on Convolutional Neural Networks:

Diagram

Description automatically generated

Fig 1: LeNet-5 Model for Network intrusion detection [3]

A Precised workflow from the suspicious flows of network for classification of the behaviour using the revised LeNet-5 model as shown in the figure above. There are three subphases [3]

1. Feature Extraction Phase

2. Model Training phase

3. Model Verification Phase

1. **Feature extraction phase:**

In feature extraction stage, the training sample data were acquired from two information source:

i) The archive dataset for behavioural highlights download from KDD'Cup99 (http://kdd.ics.uci.edu/data sets/kddcup99/kddcup99.html).

ii) The network flows which are suspicious hooked by NHSNC to draw out the new behavioural features.

Step 1.1 Model picking up utilizing KDD'Cup99 dataset

Initial, a progression of examinations was performed to explore the CNN-based classifier viability utilizing a benchmark KDD'Cup99 dataset, where the learning results were respected as a premise of model boundaries including weight network, batch\_size, batchs\_per\_epoch, ages and accuracy of arrangement.

Step 1.2 New behaviour highlights separated from recent flow of the network stream. [3]

**2. Model Training phase**:

The current study revised the model of LeNet-5 designed by LeCun et al. in 1998 [4] and they have illustrated in below Fig 2. The model which they have proposed also involves the algorithm for gradient-descent optimization Algorithm to precisely adjust the model parameters for future reinforce learning using error- derivatives of back propagation with the studying rate for all layers. [3]

Diagram

Description automatically generated

Fig 2: LeNet-5 Revised model for threat classification in Network [3]

It has involved in 6 layers Convolutional Layer C1, pooling layerS2, Convolutional layer C3, Pooling layer S4, fully connected layer and at last the classification layer.

**3. Model Validation Phase:**

A cross-validation scheme has been adopted to evaluate the predicted accuracy of Convolutional Neural Network model to get the better of over-training problem by using various n-folds of the cross-validation scheme. For Instance, K=8 means that 70 percent of the dataset collected was used in the learning experiment.

In this phase the system gives us the benefit of rapid-responding for classification of threat through the utilisation of the weights of neural networks by using the trained Convolutional Neural Networks in this phase.[3]

**Experimental Outturn:**

Step 1: Feature Extraction Phase -- To inspect the model productivity or efficiency, the principal model consolidates an revised LeNet-5 model with KDD'Cup99 dataset where LeNet-5 model is made out of at least one convolutional layers with FCLs common to artificial neural Networks where model arranges input images into 40 classes.

In the experiment, Feature extraction included a pre-process counting two stages:

I) Feature Reduction ii) Convert elements or features to a image network.

**Step 1.1**- Feature reduction -- The elements of the NSL-KDD 1999 Dataset were outclassed as per the score doled out by information gain (IG) measure. To further develop grouping speed, a bunch of diminished highlights was consistently chosen utilizing information gain (IG) plot where the quantity of component chose. A set of diminished highlights chose from top 32 positioned highlights utilizing IG approach are recorded.

**Step 1.2**- Convert features to a image matrix --- This step is to preprocess the analysis of experimental data includes The conversion of the network packets and then convert features to an image matrix and then at the end the input images are normalised to a size of 32\*32 pixels . This study selects 80% training data set as random and the remaining 20% is for testing purpose in which both data sets contain 39 sub-categories of attack samples to avoid separation bias.

**Step 2** - Model Learning phase --- In this phase they have conducted two cases of purposes which are different to test the effectiveness of CNN based classifier.

Case 1: Large amounts of samples are used to classify the major types of intrusion threats

To begin with investigation was lead and used more than 100000 samples of captured data in training process.

In the below table the data which has taken that is CNN has trained to detect the network intrusion using the trained weights of networks. The prediction accuracy for this model is 99.65%.[3]

Table

Description automatically generated

Table 1. Analysis for the type of threats in KDD cup’99dataset [3]

Case 2 : Classification of attacks for 39 sub categories

In the experiment CNNs are utilized for grouping data into 39 sub-classifications as indicated by distinct ways of behaving. In KDD cup'99 dataset, the quantity of test or sample for a particular danger type is by and large under 500 it should be equitably and arbitrarily recreated to 500 samples to keep away from too couple of information that produces the learning bias contrasted with huge examples of the assault type. Table 2 shows that the relating exactness (%) by utilizing the cross-validation technique (k = 4,8,10). For Case II, the normal exactness is 95.41 %. [3]

Table

Description automatically generated

Table 2. Accuracy combined using various n-fold values [3]

Step 3- The results show that the error has been decreased for the classification as the size of testing dataset increased. For different sizes of sample data, the prediction accuracy of threat classification increases with an increase in N value. The prediction accuracy for intrusion detection increased to 99.65% when N≥10,000. [3]

**Conclusion:**

This paper presents an intrusion detection model considering a CNNs-based classifier for upgrading the accuracy of model. Critically, the proposed approach modifies the LeNet-5 model with the versatile delta improvement algorithm to finetune the model boundaries and limit an order mistake by utilizing error derivatives of back-propagation and speedy reaction to intrusion detection utilizing Tensor stream or flow. The proposed technique works on the exactness of intrusion detection for threat characterization by utilizing improved conduct highlights from prepared CNNs. Generally, the proposed approach can upgrade the accuracy or precision for network intrusion detection. [3]

References:

1. Kaiyan He 2021 J. Phys.: Conf. Ser. **1865** 042097

2.https://www.geeksforgeeks.org/convolutional-neural-network-cnn-in-machine-learning/#:~:text=A%20convolutional%20neural%20network%2C%20or,or%20even%20eyes%20and%20faces.

3. W. -H. Lin, H. -C. Lin, P. Wang, B. -H. Wu and J. -Y. Tsai, "Using convolutional neural networks to network intrusion detection for cyber threats," 2018 IEEE International Conference on Applied System Invention (ICASI), 2018, pp. 1107-1110, doi: 10.1109/ICASI.2018.8394474.

4. https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/