**Back Propagation Neural Networks –**

In recent years, the network environment is becoming more and more complex, and the attack methods are becoming more and more diverse. Several computers are hacked because they do not consider the necessary precautions to protect against network attacks. Numerous businesses and organisations run a far higher risk of financial loss by failing to secure their systems.

The primary purpose of an intrusion detection system is to safeguard resources from danger. It analyses and predicts the behaviours of users, and then these behaviours will be considered an attack or normal behaviour. It’s a belief that a user leaves a 'print' when using the system; a neural network can be used to learn this print and identify each attack.

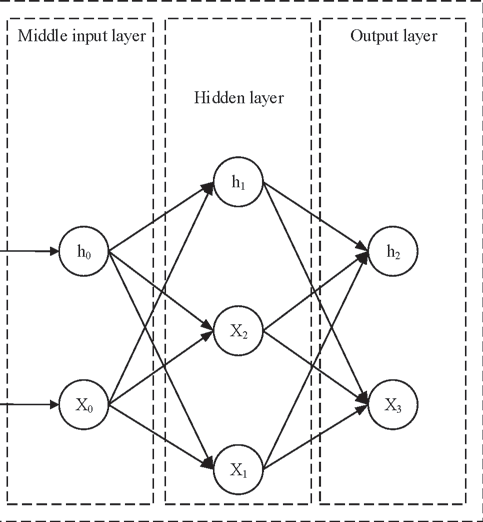
Back Propagation Neural Network is a kind of feedforward neural network which has been applied in many fields. It has a complete theoretical system and learning mechanism. Its topological structure consists of a three-layer structure. The three-layer structures are: hidden layer, input layer and output layer.

**BPNN-**

 The BPNN is based on the method of error back propagation, through multiple iterations of learning, to achieve the target accuracy. The algorithm is implemented by providing a collection of inputs and a set of desired outputs. The transformation from input to output is determined by the weights associated with the interconnections between processing elements; by modifying these interconnections, the network can adapt to the desired output (target).

BPNN has multiple layers. Each layer consists of one or more than one interconnected node with some ‘activation function’. The left-most layer is known as the ‘Input layer’ and the right-most layer is known as the ‘Output layer’. Between these two layers there may be one or more than one hidden layer. Patterns are presented as input to the network via the input layer, which in turn communicates to the hidden layers where the actual processing is done through a set of weighted connections.

 The network starts with a set of fresh patterns as input data and a set of predefined weights for each connection. It works through a forward calculation from input layer to output layer through hidden layers followed by a backward calculation from output to input layer for error rectification by adjusting the old weights in the connections. Every set of forward and backward operations is termed as a single ‘Epoch’. For every epoch, a fresh set of patterns is given to the network as input. The network is trained in this way with a training set for a certain number of epochs. The network is able to recognise the unknown pattern after the training phase in accordance with its training.

****

**Single hidden layer in BP-NN Technology**

**Working with Back Propagation -**

1. First, apply the inputs to the network and work out the output (the target); this initial output could be anything, as the initial weights were random numbers.
2. We got the inputs and respective weights, as it is a feed-forward network. The neuron will send to the next neuron, i.e., the hidden layer neuron. As it is a fully connected network, each node will receive input from all the nodes of the input layer.  The calculation gives an output which is completely different to what we have given initially (the target), since all the weights are random.
3. In this step, we need to compute the error value occurring with respect to target output and feed-forward computation values –

Error = Output (1-Output) (Target – Output)

Due to the Sigmoid Function, the "Output (1-Output)" term is required in the equation.

1. After the above operation, we need to start backwards. To update the weights based on the error value.
2. Calculating the errors for the hidden layer neurons. Unlike the output layer, we can’t calculate these directly, so we propagate back from the output layer. This is accomplished by taking the errors from the output nodes and putting them through the weights again to obtain the hidden layer errors. The weights are then altered mathematically using this inaccuracy to make the error lower. In other words, each neuron's output will be closer to its intended goal (this part is called the reverse pass). Up until the inaccuracy is barely perceptible, the procedure is performed repeatedly.

**Experimental Result –**

The proposed IDS is experimented using dataset KDD Cup99. The dataset has Denial of service attack (DoS), Probe attack, Remote to Local attack(R2L), User to Root attack(U2R) are the different attacks. Initially, pre-processing is carried out. The data is divided into training and testing at this stage. Then they applied the Back propagation neural network algorithm to the dataset. The performance measure is used to evaluate each attack computed based on the detection rate and false alarm rate. The detection rate is calculated and indicates the percentage of detected attack among all attack data.

**References-**

1. <https://ieeexplore-ieee-org.lib-ezproxy.concordia.ca/document/6508202>
2. <https://ieeexplore.ieee.org/document/9342479>
3. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.65.6790&rep=rep1&type=pdf>
4. S. Haykin, “Neural Networks – A Comprehensive Foundation”, 2nd Edition, Pearson.
5. [https://www.researchgate.net/publication/349077282\_A\_Study\_on\_Backpropagation\_in\_Artificial\_Neural\_Networks](https://www.researchgate.net/publication/349077282_A_Study_on_Backpropagation_in_Artificial_Neural_Networks )