Exploration of a Deep Learning Techniques

Edris Safari

Bellevue University, Nebraska U.S.A.

Abstract

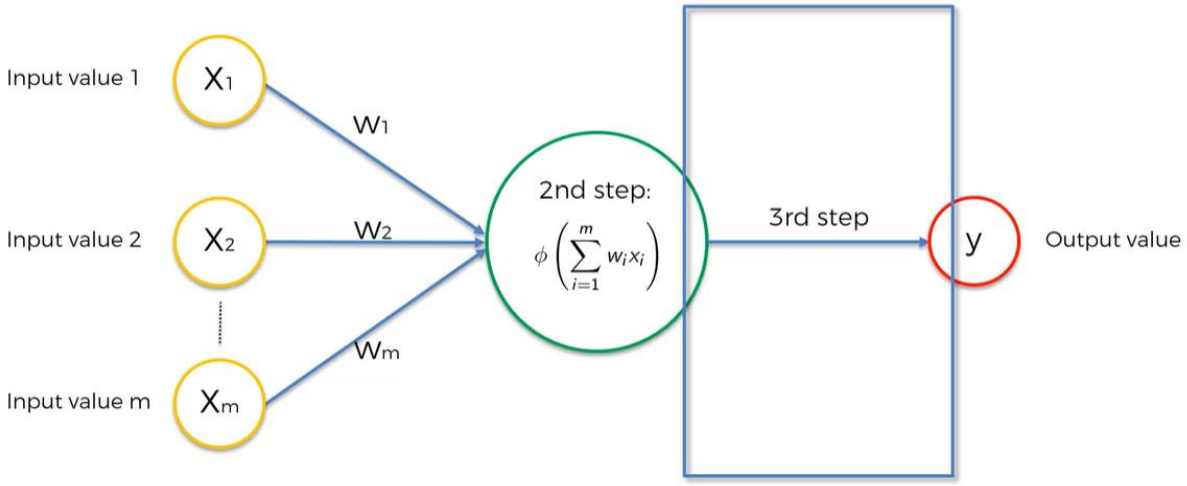
Deep learning is based on neural networks and neural networks are modeled after the way neurons work in the human brain. As the brain processes data, the neurons in the human brain connect to each other in an interwoven fashion through millions of other neurons. This happens dynamically meaning that the result of the process involves a subset of the neurons in at least the region of the brain. This is modeled by a node representing the neuron and a link to other nodes, and from those nodes to others making a network. The output of the network is the desired result. The data that the network processes could be pixel color, comments gathered from 10,000 participants, etc. This paper expounds further the concept and application of deep learning. It will describe the techniques and methodologies used to manage and control neural networks to achieve a result.

Keywords:

Exploration of a Deep Learning Techniques

Deep Learning

In their simplest form a neural network would have 1 or more input nodes connected by a line to a single node and from this single node a single line to another node. The input nodes are in the "input layer", the 2nd node in the "hidden layer" and the last node in the "output" layer. It is the hidden layer that does the work to produce the output that goes in the output layer. The work that is done in the hidden layer node is called an activation function. The parameters of this functions are the sum of the weighted values of the input parameters. In the picture below (from Udemy- Deep Learning A-Z), X1 to Xm are the inputs, W1 to Wm are the weights for each X parameter and the activation function in the hidden layer.



A neural network with multiple nodes in the hidden layer and multiple hidden layers is called a Deep Network that facilitates deep learning. In all forms, the weight and activation functions prevail in the architecture and design of the neural network. The algorithms that drive the network can be supervised, semi-supervised, or unsupervised. The concepts of forward and backward propagation also prevail in the architecture and design of neural networks. These are

comparing the outcome with a true value using loss function (forward propagation), and from output layer back toward the input layer to determine how much each parameter contributed to the error between predicted value and the true value. This iteration can be repeated several times. Each iteration is called an epoch and they contribute to the adjustment in the weight that is applied to the input of each node. As for activation function, there are several. Most popular ones are:

- Threshold Function: The output of a node with threshold function can be either zero or one. If the input is less than zero, the output is zero, if the input is greater than zero, the output is 1. They are typically used in the output layer
- Sigmoid Function: The output of a node with sigmoid function follows the sigmoid curve that show the probability of the outcome to be 1-forexample .89 or 89% probability that it is true. This function is also a good fit for the output layer.
- Rectifier Function: Also knows as Rectifier Linear Unit (ReLu) outputs zero if the input is less than or equal to zero and the value of input if it is greater than zero.
- Hyperbolic tangent(tanh): This is the as sigmoid function except the output can range from -1 to +1 rather than 0 to 1.

The rectifier function is the most popular one to use in multi-layer networks (multiple hidden layers) such as convolutional neural networks, artificial neural networks and recurrent neural networks.

The steps to create and implement a neural network is rather the same as machine learning. We create training set, and a test set. We train the algorithm using the training dataset and test it on the test dataset. Using sklearn and Keras, we can implement these steps seamlessly.We can specify the number of nodes in the input layer(number of features in each observation).

And the activation function to use for each layer. For example, the code below creates the input layer with 16 nodes and 10 feature values per observation as the inputs. Then a 16 node hidden layer node is create. Both layers use the ReLu activation function. The 3rd layer is the output layer which in the case uses sigmoid activation function because the output is a binary classifier.

```
# Add fully connected layer with a ReLU activation function
network.add(layers.Dense(units=16, activation="relu", input_shape=(10,)))

# Add fully connected layer with a ReLU activation function
network.add(layers.Dense(units=16, activation="relu"))

# Add fully connected layer with a sigmoid activation function
network.add(layers.Dense(units=1, activation="sigmoid"))
```

Conclusions

Neural networks are used when dealing massive amount of data that requires massive processing power to process. Techniques in using Neural Networks combined with the advents in computing technology have given way to application not possible only a decade ago. With Graphical Processing Units (GPU), we can distribute the processing of large networks to speed up the process. This is critical in self driving cars with vison. The deep learning algorithm in this case has to be able to recognize and detect ANY object that it might hit or be hot by in real time(see YOLO Object Detection for more information on this topic).

References

Neurons: A Curious Collection of Shapes and Sizes - https://www.brainfacts.org/archives/2012/neurons-a-curious-collection-of-shapes-and-sizes

Deep Learning A-Z - https://www.udemy.com/course/machinelearning/learn/lecture/6760384#overview

A Gentle Introduction to the Rectified Linear Unit (ReLU) - https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

PREDICTING CUSTOMER CHURN IN BANKING INDUSTRY USING NEURAL NETWORKS - https://www.indecs.eu/2016/indecs2016-pp116-124.pdf

CNN vs. RNN vs. ANN – Analyzing 3 Types of Neural Networks in Deep Learning - https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/

YOLO Object Detection (TensorFlow tutorial) - https://www.youtube.com/watch?v=4eIBisqx9_g&t=1079s