Synopsis on

Neural Network Architecture, Components & Algorithms

By

Renuka Takle (CT18005)

Rishikesh Rahangdale (CT18049)

Pranav Marjive (CT18027)

Kunta Bhargav Sai (CT18063)

Kotoju Naveen (CT18115)

Srushthi Mohod (CT18007)

Under the humble guidance of

Dr.Vilas Mahatme (HOD CTD)

2019-2020

DEPARTMENT OF COMPUTER TECHNOLOGY

KAVIKULGURU INSTITUTE OF TECHNOLOGY & SCIENCE

RAMTEK – 441 106

PROBLEM STATEMENT :

Study & Implementation of Neural Network Architecture, Components & Algorithms.

PROBLEM OBJECTIVES :

Deep Learning focuses on five core Neural Networks, including:

- **Multi-Layer Perceptron**

  The field of artificial neural networks is often just called neural networks or multi-layer perceptrons after perhaps the most useful type of neural network. A perceptron is a single neuron model that was a precursor to larger neural networks.

- **Radical Basis Network**

  In the field of mathematical modelling, a radial basis function network is an artificial neural network that uses radial basis functions as activation functions.

- **Convolutional Neural Networks**

  A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

- **Recurrent Neural Network**

  A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior.

INTRODUCTION :

Neural Networks are complex structures made of artificial neurons that can take in multiple inputs to produce a single output. This is the primary job of a Neural Network – to transform input into a meaningful output. Usually, a Neural Network consists of an input and output layer with one or multiple hidden layers within.

In a Neural Network, all the neurons influence each other, and hence, they are all connected. The network can acknowledge and observe every aspect of the dataset at hand and how the different parts of data may or may not relate to each other. This is how Neural Networks are capable of finding extremely complex patterns in vast volumes of data.

The layer or layers hidden between the input and output layer is known as the hidden layer. It is called the hidden layer since it is always hidden from the external world. The main computation

of a Neural Network takes place in the hidden layers. So, the hidden layer takes all the inputs from the input layer and performs the necessary calculation to generate a result. This result is then forwarded to the output layer so that the user can view the result of the computation.

In our tea-making example, when we mix all the ingredients, the formulation changes its state and colour on heating. The ingredients represent the hidden layers. Here heating represents the activation process that finally delivers the result – tea. In a Neural Network, the learning (or training) process is initiated by dividing the data into three different sets:

Training dataset – This dataset allows the Neural Network to understand the weights between nodes.

Validation dataset – This dataset is used for fine-tuning the performance of the Neural Network.

Test dataset – This dataset is used to determine the accuracy and margin of error of the Neural Network.

Once the data is segmented into these three parts, Neural Network algorithms are applied to them for training the Neural Network. The procedure used for facilitating the training process in a Neural Network is known as the optimization, and the algorithm used is called the optimizer. There are different types of optimization algorithms, each with their unique characteristics and aspects such as memory requirements, numerical precision, and processing speed.

LITERATURE SURVEY :

- **Python :**

Python is high level, general purpose programming language created by Guido Van Rossum. Python provides code readability, code will have fewer lines, and it is only possible for python because of large standard libraries.

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited too many tasks. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary precision decimals, manipulating regular expressions, and unit testing.

- **Multi-Layer Perceptrons :**

The field of artificial neural networks is often just called neural networks or multi-layer perceptrons after perhaps the most useful type of neural network. A perceptron is a single neuron model that was a precursor to larger neural networks.

It is a field that investigates how simple models of biological brains can be used to solve difficult computational tasks like the predictive modeling tasks we see in machine learning. The goal is not to create realistic models of the brain, but instead to develop robust algorithms and data structures that we can use to model difficult problems.

The power of neural networks comes from their ability to learn the representation in your training data and how to best relate it to the output variable that you want to predict. In this sense neural networks learn a mapping. Mathematically, they are capable of learning any mapping function and have been proven to be a universal approximation algorithm.

The predictive capability of neural networks comes from the hierarchical or multilayered structure of the networks. The data structure can pick out (learn to represent) features at different scales or resolutions and combine them into higher-order features. For example from lines, to collections of lines to shapes.

Neurons :

The building block for neural networks are artificial neurons.

These are simple computational units that have weighted input signals and produce an output signal using an activation function.

Neuron Weights :

You may be familiar with linear regression, in which case the weights on the inputs are very much like the coefficients used in a regression equation.

Like linear regression, each neuron also has a bias which can be thought of as an input that always has the value 1.0 and it too must be weighted.

For example, a neuron may have two inputs in which case it requires three weights. One for each input and one for the bias.

Weights are often initialized to small random values, such as values in the range 0 to 0.3, although more complex initialization schemes can be used.

Like linear regression, larger weights indicate increased complexity and fragility. It is desirable to keep weights in the network small and regularization techniques can be used.

Activation :

The weighted inputs are summed and passed through an activation function, sometimes called a transfer function.

An activation function is a simple mapping of summed weighted input to the output of the neuron. It is called an activation function because it governs the threshold at which the neuron is activated and strength of the output signal.

Historically simple step activation functions were used where if the summed input was above a threshold, for example 0.5, then the neuron would output a value of 1.0, otherwise it would output a 0.0.

Traditionally non-linear activation functions are used. This allows the network to combine the inputs in more complex ways and in turn provide a richer capability in the functions they can model. Non-linear functions like the logistic also called the sigmoid function were used that output a value between 0 and 1 with an s-shaped distribution, and the hyperbolic tangent function also called tanh that outputs the same distribution over the range -1 to +1.

More recently the rectifier activation function has been shown to provide better results.

- ***Data Preparation :***

You must first prepare your data for training on a neural network.

Data must be numerical, for example real values. If you have categorical data, such as a sex attribute with the values "male" and "female", you can convert it to a real-valued representation called a one hot encoding. This is where one new column is added for each class value (two columns in the case of sex of male and female) and a 0 or 1 is added for each row depending on the class value for that row.

This same one hot encoding can be used on the output variable in classification problems with more than one class. This would create a binary vector from a single column that would be easy to directly compare to the output of the neuron in the network's output layer, that as described above, would output one value for each class.

Neural networks require the input to be scaled in a consistent way. You can rescale it to the range between 0 and 1 called normalization. Another popular technique is to standardize it so that the distribution of each column has the mean of zero and the standard deviation of 1.

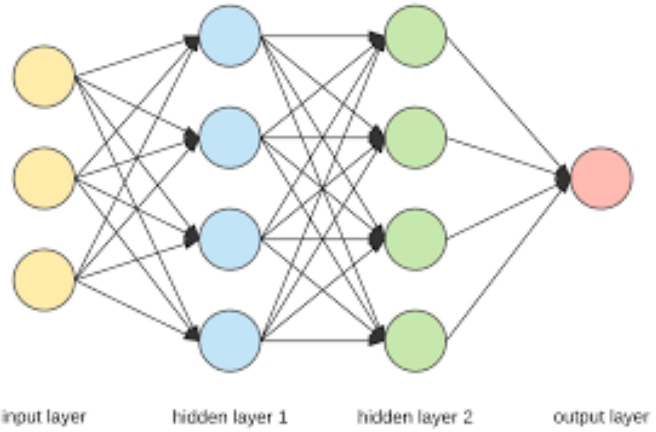
Scaling also applies to image pixel data. Data such as words can be converted to integers, such as the popularity rank of the word in the dataset and other encoding techniques.

PROPOSED APPROACH AND SYSTEM ARCHITECTURE

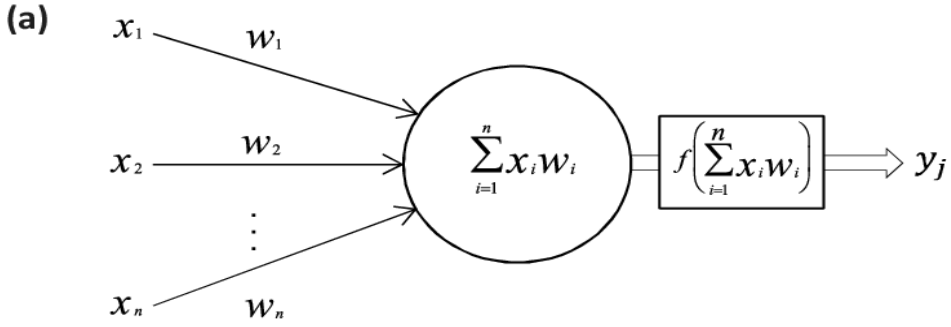
The original goal of the neural network approach was to create a computational system that could solve problems like a human brain. However, over time, researchers shifted their focus to using neural networks to match specific tasks, leading to deviations from a strictly biological approach. Since then, neural networks have supported diverse tasks, including computer vision, speech recognition, machine translation, social network filtering, playing board and video games, and medical diagnosis.

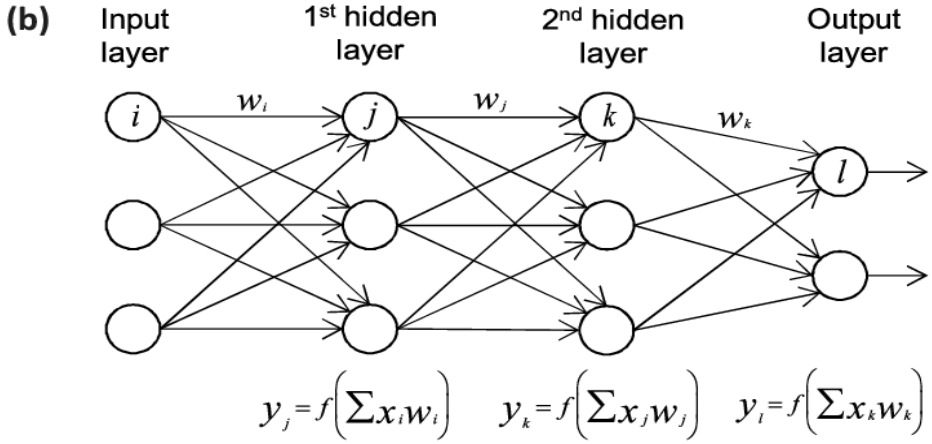
- ***HOW NEURAL NETWORKS WORK :***

A simple neural network includes an input layer, an output (or target) layer and, in between, a hidden layer. The layers are connected via node and these connections form a "network" – the neural network of interconnected nodes.



A node is patterned after neuron in a human brain. Similar in behavior to neurons, nodes are activated when there is sufficient stimuli or input. This activation spreads throughout the network, creating a response to the stimuli (output). The connections between these artificial neurons act as simple synapses, enabling signals to ne transmitted from one to another. Signals across layers as they travel from the first input to the last output layer – and get processed along the way.





When posed with a request or problem to solve, the neurons run mathematical calculations to figure out if there's enough information to pass on the information to the next neuron. Put more simply, they read all the data and figure out where the strongest relationships exist. In the simplest type of network, date inputs received are added up, and if the sum is more than a certain threshold value, the neuron "fires" and activates the neurons it's connected to.

ANN is an information paradigm that is inspired by the way of biological nervous system

Configured for specific application.

i) Data Classification

ii) Pattern Recognition

MODEL :

i) Interconnections
ii) Activation Function
iii) Learning Rules

I = i0w0 + i1w1+B

Where i0,i1 are inputs and w0,w1 are wights

B is BIAS.

Output = f(I)

F is Activation function.

Where such summations are done and output is generated from inputs.

Plan of Implementation :

1. Selection of Topic : done
2. Selection of Domain : done
3. Gathering Information : done
4. Study Python basics : 4/4/2021-14/4/2021
5. Studying Architecture of Neural Computing: 15/4/2021-20/4/2021
6. Studying different algorithms : 21/4/2021-28/4/2021
7. Implementing ,Testing algorithms – 29/4/2021-7/5/2021

Tools and Libraries used :

1. Python AI
2. Anaconda and Miniconda (idle)
3. Neurolab(library)

References :

1. https://www.google.com/search?q=neural+network+computer+architecture&oq=neural+network&aqs=chrome.0.69i59j69i57j69i59l3j69i60l3.4569j0j7&sourceid=chrome&ie=UTF-8
2. Efficient process of Deep Neutral Networks(Synthesis Lecture on computer architecture)-import 30june 2020
3. Fundamentals of Neural Network book by - Laurence V. Fausseet