Neural Networks

Neural networks, a special machine learning model, are made to recognise patterns and solve complex problems. They work by processing data in layers just like the human brain. These networks have turned out to be key in many areas such as image understanding or natural language handling because they can learn from data and make good predictions. This report is about the main structure of neural networks, different types present in them and how we train these models.

The structure of a neural network includes neurons, layers, and activation functions. Neurons are the basic units that make up neural networks; they imitate human brain's neurons. Every neuron takes in input, does some processing on it and gives out an output. A neuron's input usually comes from adding up the results of previous layer neurons, all multiplied by certain weights. After this, we apply an activation function to the added results for introducing non-linearity in neural network that can handle complicated patterns.

Neural networks have layers for input, hidden and output. Input layer is where data comes in, with every neuron representing a feature of the input data. Hidden layers are located between the input and output layer. In these layers, actual computation takes place along with pattern recognition. A network might contain a single hidden layer or many, each with different numbers of neurons. The last layer in the network is called output layer where it makes predictions. The number of neurons in this final part matches up with how many classes or values we want to predict from our model.

Activation functions are responsible for the non-linearity in neural networks. They allow the network to learn and perform complicated tasks. Some common activation functions are Sigmoid, which gives output values between 0 and 1 and is generally used for binary classification; ReLU (Rectified Linear Unit), when input is positive it gives this as output but if not then it outputs zero – this function has become popular because of its straightforwardness and effectiveness; Tanh, similar to sigmoid but instead of giving outputs from 0 to 1 it gives values between -1 and 1 making it good for data that is centered around zero.

Different neural networks are designed for different types of data and tasks. A simple kind called Feedforward Neural Network (FNN) has information moving only in one direction: from the input layer through all hidden layers to reach the output layer. This is useful for jobs like recognizing images or solving basic regression problems. FNNs, although they work well in modeling static data, have difficulties with sequential information.

Convolutional Neural Networks (CNNs) are made for managing grid-like data, like images. These CNNs use convolutional layers that learn spatial hierarchies of features in an automatic and adaptive manner – this property makes them very popular for recognizing images and videos as well as analyzing medical pictures or doing some natural language processing tasks where knowing about spaces is crucial.

Recurrent Neural Networks (RNNs) are distinct from Feedforward Neural Networks (FNNs) due to their connections, which form directed cycles. These recurrent connections permit the network to sustain a memory of previous inputs, making RNNs effective for handling sequence data such as time series prediction, speech recognition and natural language processing. Variants like Long Short-Term Memory (LSTM) networks tackle the difficulty of gradients going away in normal RNNs by letting the network learn lasting dependencies.

The process of training a neural network is about changing the weights in connections between neurons so that the error in predictions becomes as small as possible. This starts with forward propagation, where input data moves through the network, layer by layer, creating forecasts. The predicted results are matched up against real targets to compute loss or mistake; this quantifies how close predictions are from desired outcomes.

Backpropagation comes next, which involves finding the gradient of the loss function for each weight in our network. This shows how much a certain weight contributes to total error. The error travels from output layer towards input layer, and gradients are computed for every layer. After the gradients have been calculated, methods of optimization such as Stochastic Gradient Descent (SGD) or Adam are employed for updating weights. These algorithms decide on the magnitude and direction (learning rate) of weight changes, progressively reducing the loss function. Correct setting up of these algorithms is very important so that network can arrive at an optimal solution.

Neural networks, in simple terms, are a type of model that imitates the way human brain works. They have brought about big changes to artificial intelligence because they can process complicated information and solve problems across many different areas. Knowing the basic structure, various types of neural networks and how training happens is important for creating useful models. As research on neural networks progresses, it's predicted that they will become more complex and bring advancements to fields like self-governing systems, medical care domains among others.

Bibliography

AWS. (2023). *What is a Neural Network? AI and ML Guide - AWS*. Amazon Web Services, Inc. https://aws.amazon.com/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a

Staff, C. (2024). *Neural Network Examples, Applications, and Use Cases*. Coursera. https://www.coursera.org/articles/neural-network-example#:~:text=Neural%20network%20examples%3A%20Technology&text=As%20a%20framework%2C%20it%20powers

Geeksforgeeks. (2019, January 17). *Neural Networks | A beginners guide*. GeeksforGeeks. https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/

https://www.cloudflare.com/learning/ai/what-is-neural-network/ (no apa referenced)

Chen, J. (2022, September 21). *What is a Neural Network?* Investopedia. https://www.investopedia.com/terms/n/neuralnetwork.asp

Used 3blue1brown on youtube