Alzheimer’s Detection with Convolutional Neural Networks

Technical Research Document – Minor Artificial Intelligence for Society

Sverre van Gompel  
15-10-2021



# Introduction

The medical field is currently experiencing an AI revolution. Many hospitals are conducting their own research on implementing artificial intelligence solutions within their processes. For example, [ETZ in Tilburg recently applied AI to detect fractures on x-ray photo’s](https://www.etz.nl/Over-ETZ/Nieuws/2021/09/AI-toepassing-ziet-breuken). AI has the potential to provide substantial improvements to existing processes within the medical field, as well as the possibility to innovate new solutions.

A possible use case for AI is assisting radiologists in the diagnostic process. An AI could analyse scans and highlight any irregularities it detects. By combining the expertise of radiologists with an AI, misdiagnoses could potentially be prevented, and Alzheimer’s disease could be detected earlier, improving overall outcomes for patients. This leads to the following question:

How can neural networks be used to detect various stages of Alzheimer’s disease?

# Contents

[Introduction 2](#_Toc83807082)

[Contents 2](#_Toc83807083)

[1. What are neural networks? 3](#_Toc83807084)

[2. How a neural network works 3](#_Toc83807085)

[3. What types of neural networks exist? 3](#_Toc83807086)

[3.1. Perceptron (P) 3](#_Toc83807087)

[3.2. Feed forward (FF) 4](#_Toc83807088)

[3.3. Deep feed forward (DFF) 4](#_Toc83807089)

[3.4. Recurrent neural network (RNN) 5](#_Toc83807090)

[3.5. Convolutional Neural Network (CNN) 5](#_Toc83807091)

[4. How does a Convolutional Neural Network function? 5](#_Toc83807092)

[Convolutional layer 5](#_Toc83807093)

[Feature detector 5](#_Toc83807094)

[5. What are General Adversarial Networks (GANs)? 6](#_Toc83807095)

[6. Can GANs provide better performance than CNNs in this project? 6](#_Toc83807096)

[References 6](#_Toc83807097)

# What are neural networks?

To put it simply, neural networks – also known as artificial neural networks (ANNs) or simulated neural networks (SNNs) – are a method of machine learning. Their name and structure are inspired by the human brain, resembling the way that biological neurons signal to one another. ANNs are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer (IBM Cloud Education, 2020).

What sets neural networks apart from other machine learning methods is their ability to learn from their own errors; allowing them to ingest data in its raw form and extract features by themselves. This opens the doors to many different applications for neural networks, from learning to play a game to detecting fractures on x-ray images.

# How does a neural network function?

## Nodes

Each connection between nodes in a network has an associated weight. This weight is used to calculate the weighted sum of all the inputs a node receives. Then, the bias is added to the weighted sum to compute the output signal of the node. This can be written as the following equation, where ŷ represents the output signal, *w* represents the connection weight, *b* represents the bias, and *a* represents the input value:

The sigmoid is only calculated if the weighted sum + bias is higher than the threshold, after which the node is activated. Otherwise, the node is not activated, and no signal will be sent out (3Blue1Brown, But what is a neural network? | Chapter 1, Deep learning, 2017).

## Loss and gradient descent

When initialising a network, random weights and values for each node and connection are generated. This, unsurprisingly, leads to completely inaccurate and nonsensical outputs. A *loss* function is then defined – a way of telling the network what the output should have been. In mathematical terms, this function adds up the square of the differences between the outputs and the correct values. Again, ŷ represents the output signal, and *y* represents the actual value:

Note that this function does not need to be squared, it could also be the absolute value or anything else. The square is commonly used however, because it always results in a positive value. This *loss* is smaller when the network is closer to the actual value, and higher when it is far off (3Blue1Brown, Gradient descent, how neural networks learn | Chapter 2, Deep learning, 2017).

Then, the derivative can be used to update the weights and biases by decreasing or increasing. The derivative tells us the direction (either decrease or increase). This is known as gradient descent.

## Backpropagation

In backpropagation – instead of working from left to right – the network is updated from right to left. After the loss is calculated from the output the network provided (for multiple training examples to prevent overfitting, more on this later), the weights and biases of each node in the output layer can be updated. Then, the layers to the left can be updated sequentially. However, each node in these layers can have multiple nodes to the right ‘telling’ the node how to update the weights, as shown in the example below:

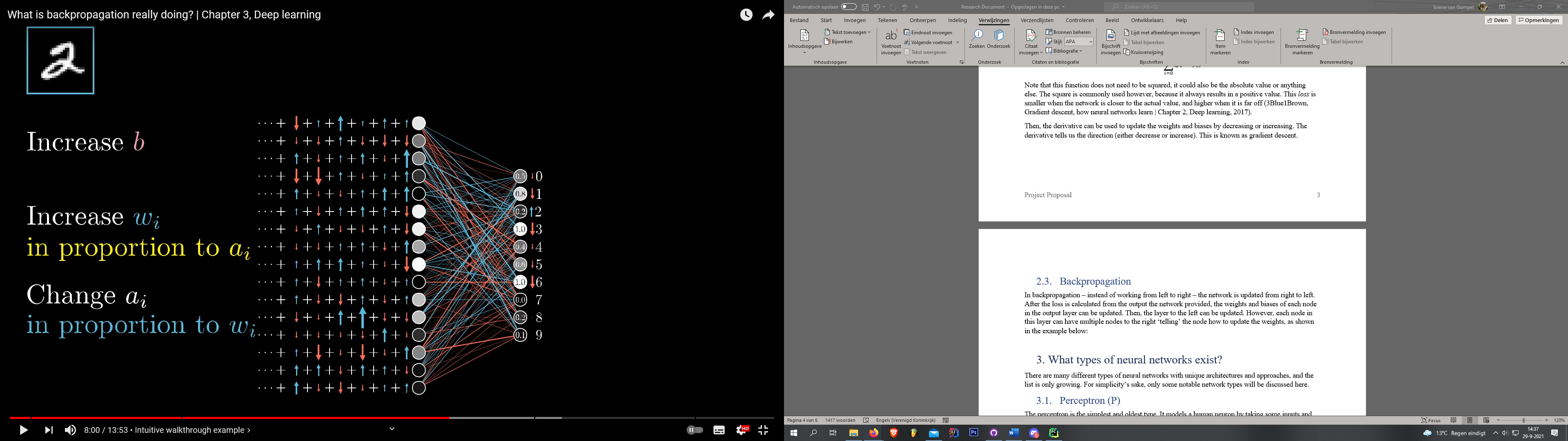


Figure 1: Updating weights and biases. (Source: 3Blue1Brown)

In this example, each node on the right side is connected to every node on the left side. Each of those nodes ‘wants’ to update the nodes its connected to, indicated by the red and blue arrows and lines, representing negative and positive increases respectively. Since the nodes on the left are connected to multiple nodes providing a value to update weights, those values must be summed up to calculate the final updated weight for each node. This process is then repeated throughout the rest of the network (3Blue1Brown, What is backpropagation really doing? | Chapter 3, Deep learning, 2017).

# What types of neural networks exist?

There are many different types of neural networks with unique architectures and approaches, and the list is only growing. For simplicity’s sake, only some notable network types will be discussed here.

## Perceptron (P)

The perceptron is the simplest and oldest type. It models a human neuron by taking some inputs and calculating the weighted sum (all the inputs multiplied by their respective weights, summed up). It then applies the activation function to create an output (Tch, 2017). *Figure 2* shows a simple diagram of how a perceptron works.

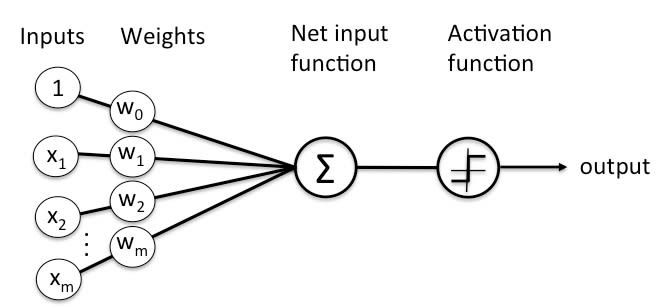


Figure 2: Diagram of a perceptron. (Source: simplilearn.com)

## Feed forward (FF)

Feed forward networks are also quite old, originating from the 50s. This network consists of multiple perceptrons split into an input layer, one hidden layer, and an output layer. Each node is connected to every node in the next layer. Activation always flows from the input layer to the output layer, never backwards (Tch, 2017). *Figure 3* shows the structure of such a network.

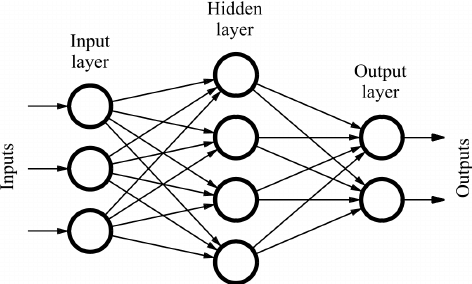


Figure 3: Diagram of a feed forward neural network. (Source: deepai.org)

## Deep feed forward (DFF)

DFF networks are in essence just feed forward networks, but with more than one hidden layer. The key difference between DFF and regular FF is the amount of computing power required; in DFF networks, the additional layers lead to exponential growth in training times. Because of this, DFF networks only started to be used in the early 2000s, after better approaches to train these networks were developed (Tch, 2017).

## Recurrent neural network (RNN)

A recurrent neural network (RNN) is a type of artificial neural network which uses sequential data or time series data, by being able to store the state of information of previous inputs to generate the next output. These deep learning algorithms are commonly used for ordinal or temporal problems, such as language translation, natural language processing (NLP), speech recognition, and image captioning; they are incorporated into popular applications such as Siri, voice search, and Google Translate (IBM Cloud Education, 2020).

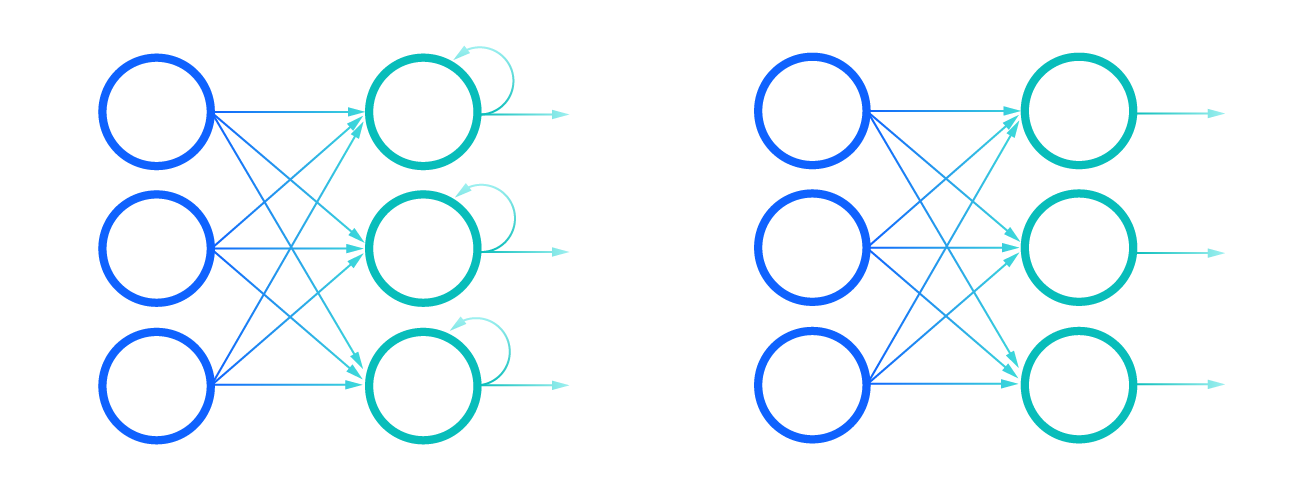


Figure 4: Comparison of Recurrent Neural Networks (left) and Feedforward Neural Networks (right). (Source: ibm.com)

## Convolutional Neural Network (CNN)

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They consist of three main types of layers: the convolutional, pooling, and fully-connected layer. CNNs are used to power image recognition and other computer recognition tasks (IBM Cloud Education, 2020). As the objective of this project is to achieve image classification, CNNs may be a great choice of network.

# How does a Convolutional Neural Network function?

As mentioned in the previous chapter, the consist of three main layers: the convolutional, pooling, and fully-connected layer.

### Convolutional layer

This layer is the core building block of CNNs, it’s where most of the computation occurs. It requires a few components: input data, a filter, and a feature map. Assume that input data is a coloured image, which is made up of a matrix of pixels in 3D. This means the input will have three dimensions – a height, width, and depth – corresponding to RGB in an image. There is also a feature detector, known as a kernel or filter, which moves across the receptive fields of the image, checking if a feature is present. This is known as convolution (IBM Cloud Education, 2020).

### Feature detector

# What are General Adversarial Networks (GANs)?

# Can GANs provide better performance than CNNs in this project?

# References

3Blue1Brown. (2017, October 5). *But what is a neural network? | Chapter 1, Deep learning*. Retrieved from https://www.youtube.com/watch?v=aircAruvnKk

IBM Cloud Education. (2020, October 20). *Convolutional Neural Networks.* Retrieved from ibm.com: https://www.ibm.com/cloud/learn/convolutional-neural-networks

IBM Cloud Education. (2020, August 17). *Neural Networks*. Retrieved from ibm.com: https://www.ibm.com/cloud/learn/neural-networks

IBM Cloud Education. (2020, September 14). *What are recurrent neural networks?* Retrieved from ibm.com: https://www.ibm.com/cloud/learn/recurrent-neural-networks#toc-what-are-r-btVB33l5

Tch, A. (2017, August 4). *The mostly complete chart of Neural Networks, explained*. Retrieved from towardsdatascience.com: https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464