How Do Neural Networks Work?

In everyday applications, we see complex neural networks and powerful machine learning models solving simple tasks, thereby facilitating the users' experiences in the application. Google Photos, for example, provides the user with a list of faces appearing in their collection. Here, an AI image recognition model is at work. This machine learning model searches through the user's photos and searches for unique faces and creates a list of all instances of this face in the user's photos. Google Photos also has a keyword search feature, where a user can enter a keyword into the search bar that describes the photo they are looking for, and a machine learning model analyzes all photos and selects certain ones based on the keyword entered by the user.

But how do these machine learning models analyze the photos?

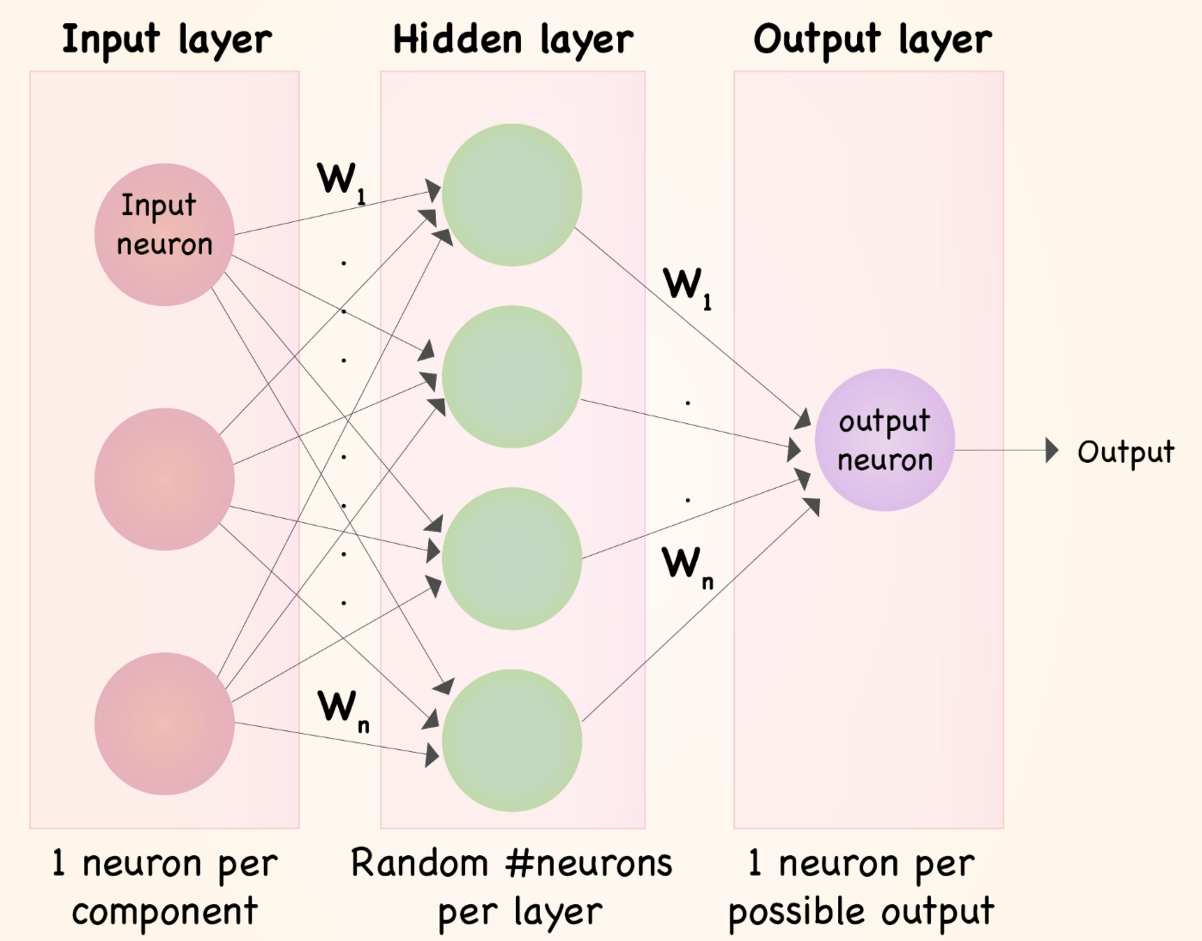
Neural networks.

What is a Neural Network?

"Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another." - IBM Cloud Education

According to an article published by the MIT News Office, neural networks were first proposed in 1944 by Warren McCullough and Walter Pitts, two University of Chicago researchers who moved to MIT in 1952 as founding members of what's sometimes called the first cognitive science department. Neural nets were a major area of research in both neuroscience and computer science until 1969, when, according to computer science lore, they were killed off by the MIT mathematicians Marvin Minsky and Seymour Papert, who a year later would become co-directors of the new MIT Artificial Intelligence Laboratory.

An article written by Chris Woodford, on ExplainThatStuff, explains how neural networks has several artificial neurons called units arranged in a series of layers each of which connects to the layers on either side. **Input units**, receive information from the outside world that the network will attempt to learn about, recognize, and process. **Output units** sit on the opposite side of the network and send signals about how the network should respond to the information received. **Hidden units** rest between output and input units. Most neural networks are fully connected. This means that each output unit and each hidden unit is connected to every unit in the layers on either side. These connections between separate units are represented by a number called a weight, which can be positive or negative. If a weight is positive, one unit excites another, whereas if the weight is negative, one unit inhibits another. The higher the weight, the more influence one unit has on another.

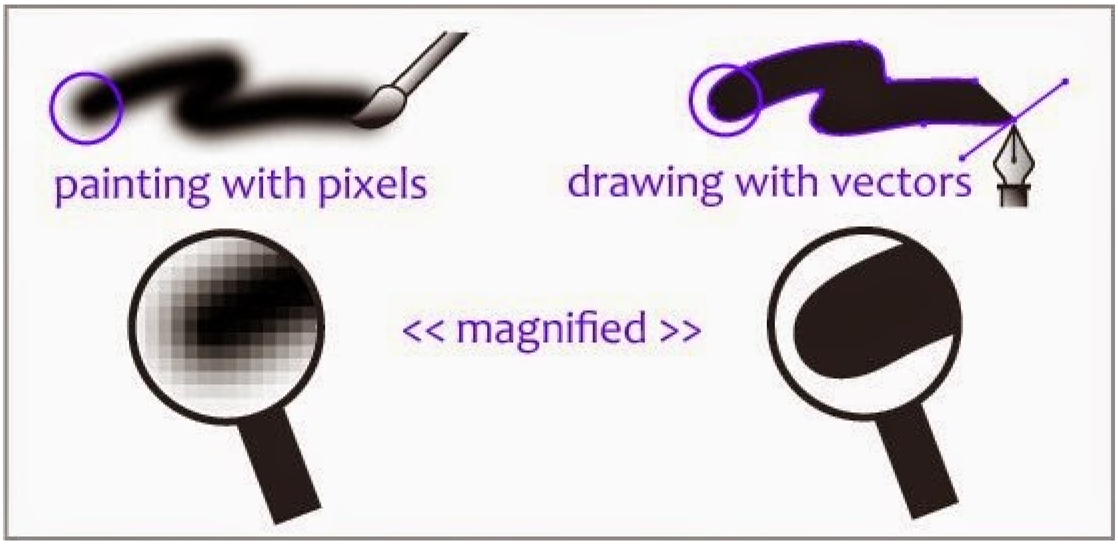


There can be one or more hidden layers resting between input and output units in a neural network. Inputs are fed in from the left into the input units (red), activate the hidden units (green), and make outputs that are fed out from the output unit (purple) on the right. The weights (W) of connections are gradually adjusted as the neural network learns.

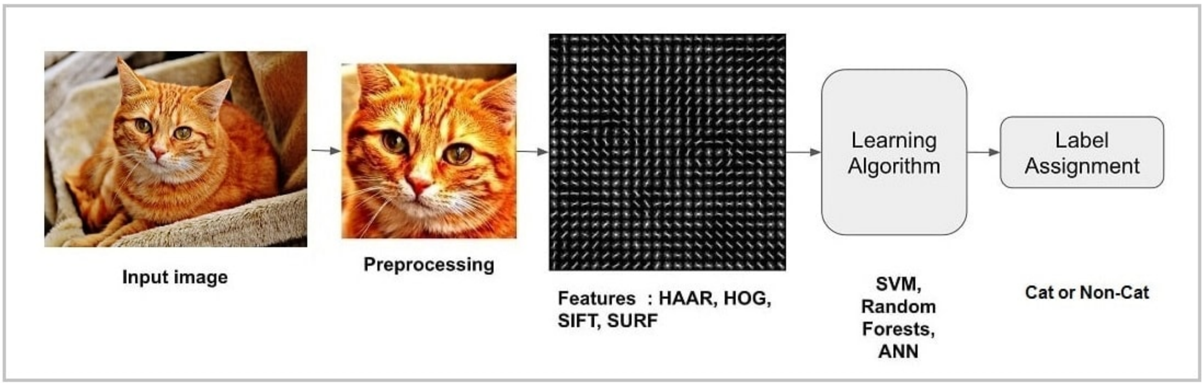
Although a simple neural network could consist of just three layers, as seen in the image above, it could also comprise many different layers, giving it a richer and more complex structure, which would result in a deep neural network (DNN). While a DNN does tackle tougher problems, it needs much more training because of its increased complexity and intricacies.

How does a Neural Network read images?

AI image recognition tries to mimic human eyes. An article written by Maruti Techlabs, explains that the human eye perceives an image as a set of signals which are processed by the visual cortex in the brain, which results in a vivid experience of the scene. The computer, however, perceives an image as a raster image or a vector image. Raster images are a sequence of pixels with numerical values for colors, while vector images are a set of color-annotated polygons.



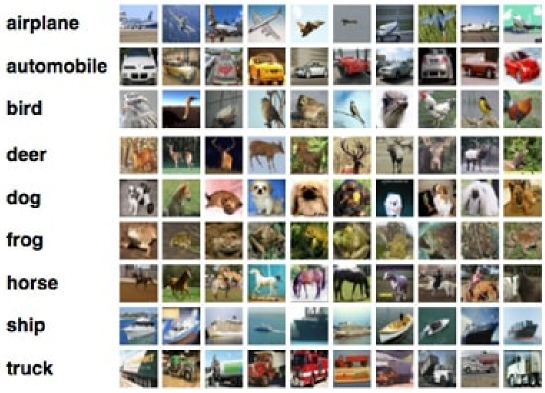
"*To analyze images the geometric encoding is transformed into constructs depicting physical features and objects. These constructs can then be logically analyzed by the computer. Organizing data involves classification and feature extraction. The first step in image classification is to simplify the image by extracting important information and leaving out the rest. For example, in the below image if you want to extract cat from the background you will notice a significant variation in RGB pixel values.*"



The image can be simplified by running an edge detector on the image. The circular shape on the face and eyes can easily be discerned in these edge images, so it is possible to conclude that edge detection retains the essential information while disregarding the non-essential.

Before a classification model can do its job, it needs to be trained. The general principle in machine learning algorithms as points in higher dimensional space. It then tries to find planes or surfaces that separate this higher dimensional space in a way that all examples from a particular case are on one side of the plane.

Cat Images

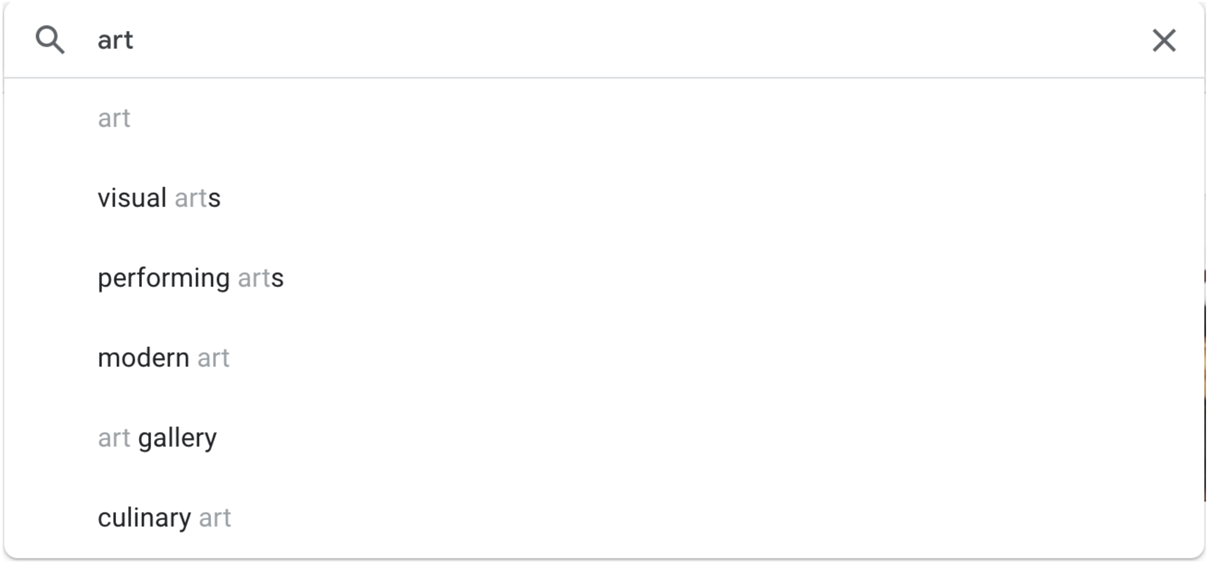


Non-Cat Images

Training data is different from test data, which also means we remove duplicates (or near duplicates) between them. This data is fed into the model to recognize images. We have to find the image of a cat in our database of known images which has the closest measurements to our test image. All we need to do is train a classifier that can take the measurements from a new test image and tells us about the closest match with a cat. Running this classifier takes milliseconds. The result of the classifier is the ‘Cat’ or ‘Non-cat’.

Training data is different from test data. This data is fed into the model to recognize images. Based on the cat example, the image of a cat has to be found in the database of known images of cats, as seen above. The classifier needs to be trained to take measurements from a test image and send out the closest match with a cat. The result of this classifier is 'Cat' or 'Non-cat.' Referring back to the Google Photos example, the face classifier uses a database of what faces look like and compares it to the photos in a user's collection. It then groups them according to similarities between faces extracted.

The keyword search works in the same way. The classifier analyzes photos based on those in its database and tries to correlate them. For example, if 'art' is typed in, the classifier scans all photos to decide which is the most relevant to 'art' based on the data it has in its database that has been labeled with 'art'.



Ultimately, it's important to remember that a neural network, physically, is nothing like the brain. It's a complex algorithm comprising a bunch of clever math and tons of equations.