Chapter 00| Basics of Neural Networks

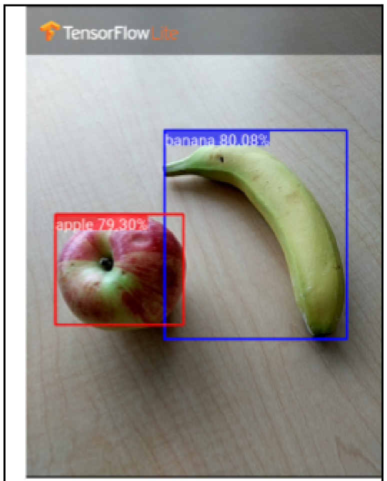
~~~~~~

Overview of AI and its Parts

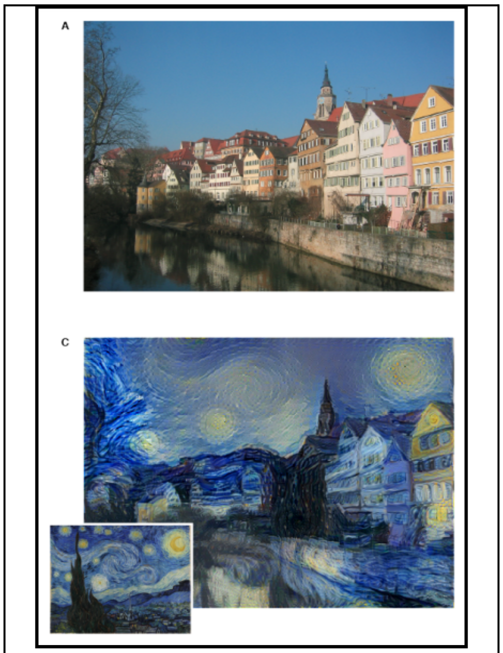
To build some more background information on AI, we need to understand where AI is used today. Below is a list of many different current world applications of AI which are categorized by problem type.

Computer Vision

- Augmented Reality (AR)
- Fake Image Generation (GANs)
- Object Detection (example from Tensorflow on the right)
- Deep Fakes (creation of **fake** images based on real images, example below on the right)
- Neural Style Transfer (merging two images to generate a new image, example from Figure 2 of this paper below on the left)



Tensorflow object detection



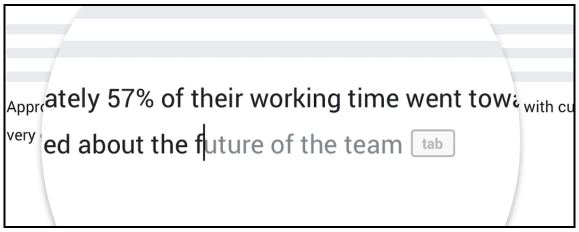
Neural Style Transfer

Example of Deepfake imaging

Look! Computers will become the next van Gogh!
~~~~~~

Recommender Systems

- Netflix
- Youtube
- Google Docs Auto-Complete (example below from Thurrott):



Appr ately 57% of their working time went tow with cu
very ed about the future of the team

Audio

- Song/Voice Generation
- Smart Speakers (example below on the left)
- Audio Processing Voice Changers (example below on the right from Murf AI)

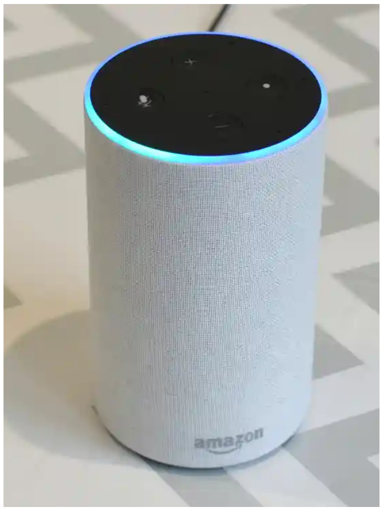
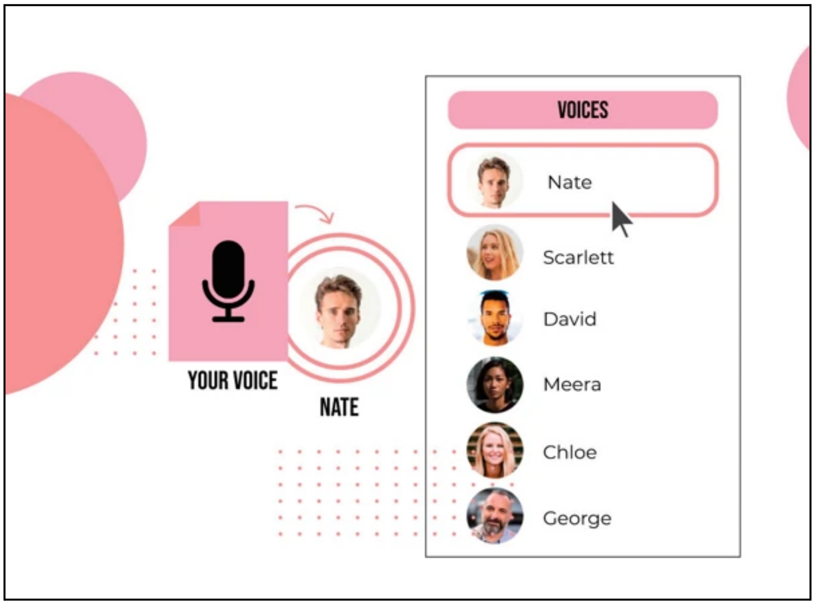


Image from the Guardian



Note that the list above is **not exhaustive** of all the current applications of AI. In the future AI may also be used in many other applications due to its potential.

Most AI applications fall into **machine learning** which consists of many different types of algorithms that try to get “smarter” as they go through a **training phase**. This training phase means that the algorithm is **not explicitly programmed** so AI becomes its own branch inside of the field of computer science.

There are 3 different learning methods used in machine learning: supervised learning, unsupervised learning, and reinforcement learning.

Supervised Learning

Step 1: Algorithm is given a set of input data
Step 2: Algorithm predicts answers to the data
Step 3: **“Show”** the algorithm the correct answers
Step 4: **“Ask”** the algorithm to make more accurate predictions
Step 5: Repeat until the algorithm is **“smart/artificially intelligent”**

You might argue that this repetition of predicting and showing the algorithm the correct answers is a waste of time. The algorithm should see the correct answers once, memorize them and then, boom, 100% accuracy. However this means that the algorithm cannot predict on other data that it has not been **“shown”** the correct answers to. As a result, the algorithm/machine has to **learn**. Hence, the name for this field of study: **machine learning**.

Since this book is only an introduction to AI, we are going to focus on supervised learning problems as these are easier to work with for beginners over unsupervised or reinforcement learning problems.

Unsupervised Learning

Step 1: Algorithm is given a set of input data
Step 2: Algorithm predicts answers to the data
Step 3: Calculate the **“error”** of the predictions **without** any correct answers
Step 4: **“Ask”** the algorithm to make **“better”** predictions
Step 5: Repeat until the algorithm is **“smart/artificially intelligent”**

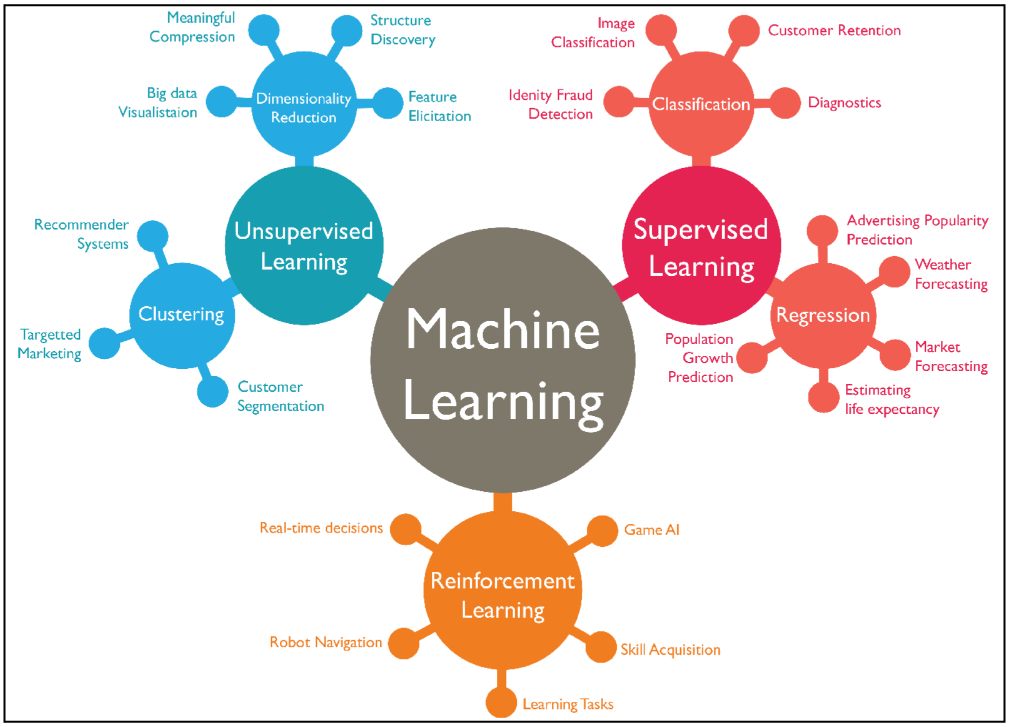
Unsupervised learning might seem **counterintuitive**. How can a machine learn without “seeing” any correct answers? This is because for some problems, **there are no correct answers**. An example of this is in recommender systems. There is no correct answer to which video/movie to recommend to someone, since it is subjective to the user.

For recommender systems, the machine tries to get to **“know”** the user (or “learn” what the user likes to watch). The algorithm’s job is then to keep the user entertained for as long as possible. While the algorithm is learning, the algorithm tries to get better at its job (keeping the user entertained for even longer).

Reinforcement Learning

This type of learning is very complex and varies depending on the actual problem at hand, so we will not explain its procedure to keep the book at an introductory level. The most prominent example of reinforcement learning is in games, such as in Go, where Deepmind's AlphaGo beat Lee Sedol, a professional Go Player (DeepMind).

The graphic below from Word Stream does a good job summarizing the breakdown of machine learning into its components and subcomponents:



Since we are focusing on supervised learning, in this book we will be working on the regression and classification problems of supervised learning.

Regression is when the model is trying to predict a continuous value, such as stock/house prices, temperature/humidity/rainfall, age, etc.

Classification is when the model is trying to sort the inputted information into classes/categories. **Binary classification** is when there are only two classes/categories and is used when classifying tumors being benign (noncancerous) or malignant (cancerous), confirming the face belongs to the owner of the phone, or other two-sided questions. **Multi-class Classification** is when there are more than two classes and can be used in object recognition (whether the image is a dog, cat or person), determining the type of news article, etc.

Currently, many state-of-the-art AI algorithms are based on one group of algorithms in machine learning. These algorithms are inspired by the **biological brain** (a neural network) of humans because researchers believed that using the structure of the biological brain will make computers as smart as humans. These algorithms, called **artificial neural networks,** are what we are going to learn in this book.

When artificial neural networks get complex, they are called **deep neural networks** and their learning phase is known as **deep learning**. However, this book is not going to teach deep neural networks because they rely heavily on the understanding of the fundamental concepts of artificial neural networks. Deep neural networks also use a large amount of a computer's resources (CPU/GPU and RAM) which is not ideal for beginners.

~~~~~~

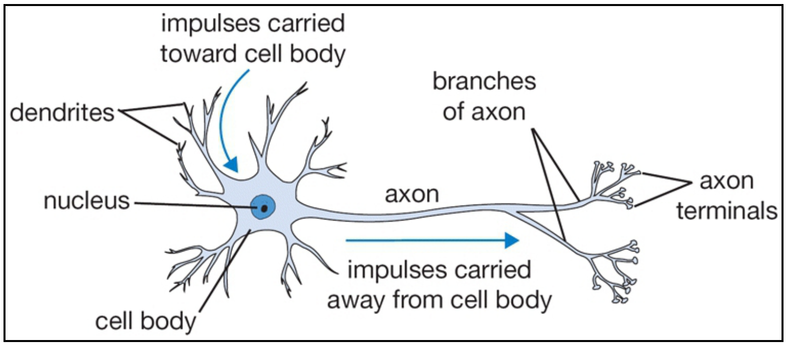
Connecting Biological Neural Networks to Artificial Neural Networks

Now, let's try to understand how people modeled an artificial neural network from a biological one.

A **biological** brain is made up of many **neurons** that are connected together to form a **neural network**. This should be easy for those who have taken a psychology class, but what is a neuron?

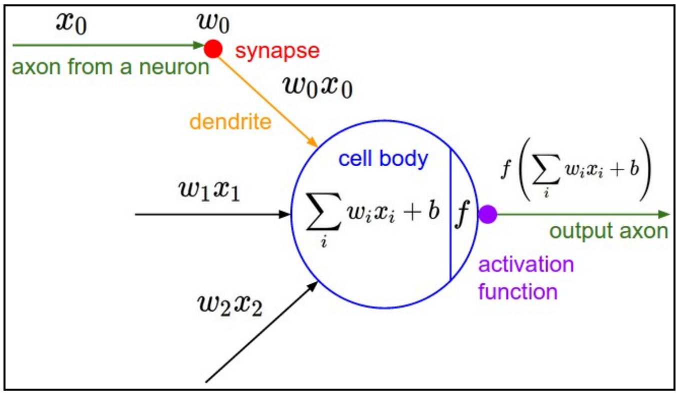
Source Image: NIH - National Eye Institute

Below is an image of a **biological** neuron from Stanford University's CS231n Course Notes.


impulses carried toward cell body
dendrites
branches of axon
nucleus
axon
axon terminals
impulses carried away from cell body
cell body
~~~~~~

A biological neuron, shown in the image above, consists of dendrites that receive information from (the outputs of) other neurons and the cell body which does some “thinking” within. Its outputs are carried away from the cell body to axon terminals as input for the next neurons.

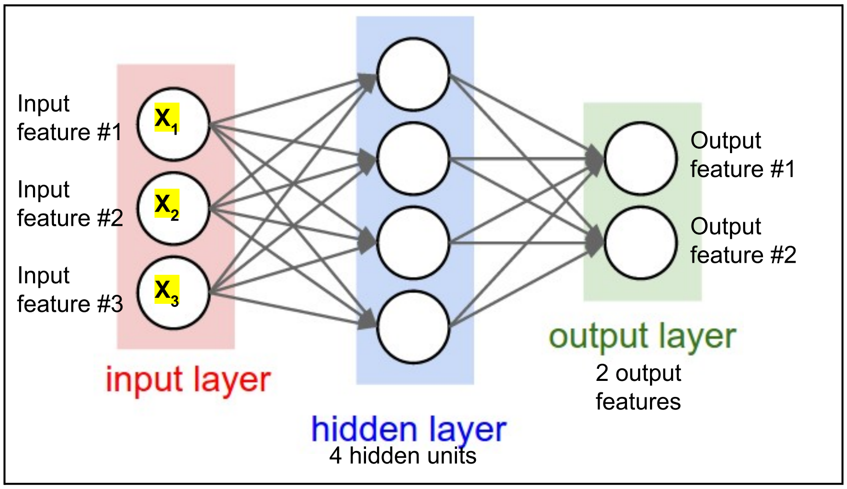
Now how are the dendrites, cell body, and axon terminals mapped artificially? Below is an image of the artificial neuron from Stanford University's CS231n Course Notes.



Don’t try to understand everything that you see in the image.

The key takeaway is how the components of a biological neuron are mapped to an artificial neuron.

Now by mapping multiple connected biological neurons to artificial ones, to get an artificial neural network, we get a **model** (original image from Stanford University's CS231n Course Notes):



The circles in the input layers represent **input features**. From a biological point of view, they can be thought of as the human senses (sight, hearing, touch, taste, and smell).

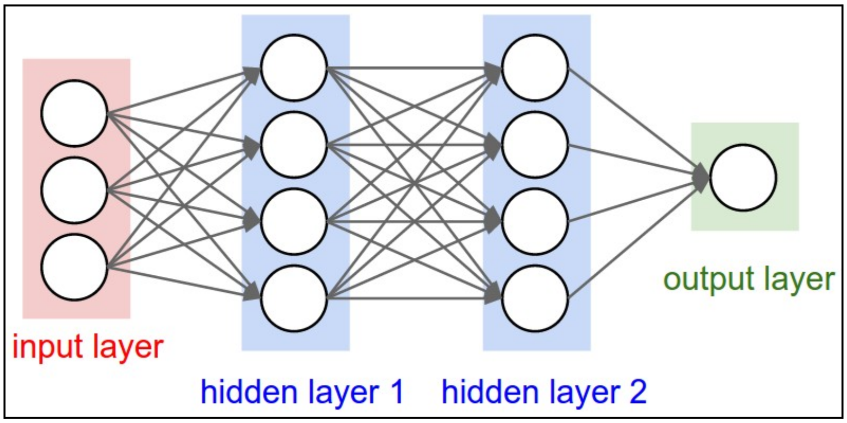
However, this artificial neural network only has 3 senses.

Virtually, these input features are represented by the variables, X_1, X_2, X_3. Because they are represented by variables, these input features are NOT neurons. In the last section of this chapter, we will make the idea of input features more concrete, but just know that the number of input features and what each input feature describes varies depending on the problem type.

Next, we have a hidden layer (the middle layer between input and output layers) that contains 4 hidden units. Each circle represents a neuron, so there are 4 neurons in the hidden layer (1 neuron per hidden unit). As you can see, each input feature passes through the dendrites of the neurons in the hidden layer. Then, each of their outputs move through the output axon to the dendrites of the neurons in the output layer.

The output layer of this model has 2 output features (output neurons) and whatever they output are not passed anywhere. As a result, **the output of the output layer is the prediction of the model** (artificial neural network). We call this a two-layered model with 4 hidden units and 2 output neurons. Instead of a three-layered model, we only count the layers that contain neurons. Remember the circles of the input layer are not neurons but variables in code.

Looking at another example of a model (image from Stanford University's CS231n Course Notes):



This model has 3 input features, 2 hidden layers, each with 4 neurons/units, and only one output feature. At this point, it might be confusing as to why hidden layers are needed in artificial neural networks, where an output layer should be enough, since the output layer contains neurons that can predict. As a result, our first example of a model in action will be a model without a hidden layer, and the motivation for hidden layers will become clear.

~~~~~~
~~~~~~

Steps for Artificial Neural Network

1. **Define Model Architecture** (Start with a "dumb" brain) – Technically, we have done this when looking at the artifical neural network. However, we have no understanding of what happens inside the neuron body (how the neuron calculates its outputs). Also we have not looked at any code for a truly artificial (virtual) neural network. When we define the model, think of all the neurons in the brain as **"dumb"** and therefore the brain is **"dumb"**. Think of a newborn baby's brain: because it has not yet learned anything, it is "dumb", just like a newly defined model.

2. **Training/Learning Phase** (Make the brain become "smart") – This is the step where messy math equations come in to make the brain **"smarter"**. Unfortunately, there's really no better way to teach an artificial neural network than to use calculus to guide the model in the right direction. Don't worry, because instead of calculus equations we'll be seeing algebraic equations (that should hopefully hit closer to home).

3. **Other** (Fixes to the limitations or problem of the current model) – We are not going to be dealing with this step in the beginning since there isn't much to fix when we haven't seen where the model has **limitations/problems**. Think of this step not as changing the method of training but a way to improve the outcome of training to make the brain even **"smarter/better"**.

~~~~~~

Basic Application of an Artificial Neural to Help Concepts "Sink In" Faster

To have a better understanding on what input/outputs features are, let's say you have taken the SAT and on the math section you score a 500. Let's also say that in your opinion, this is not good enough and so you decide to retake the SAT. This time, however, you will practice before the 2nd test to improve your score. Let's assume you set a high bar for yourself and want a score of 800 next time and there are 90 days left until testing. How often should you practice and how long should you practice each time?
~~~~~~

To answer this question, you decide to do some research and find out that there are official SAT practices in Khan Academy. However, this does not answer your question so you continue researching and find that “Students who used Official SAT Practice for six or more hours and followed at least one best practice scored 39 points higher than students who did not use Official SAT Practice” (College Board).

Well, you need to score 300 points higher, so you cannot only practice for six hours, plus you do not know how to split practice in between 90 days. As a result, you’re in dire need of a personal plan. The only way to get such a plan would be to use a tool that can predict for you. Look no further than to an AI that will tell you **how often and how long** you should practice each time to bring your SAT math score from 500 to 800 points in 90 days!

In this example, the input features to your model can be your current SAT section score (500), the score you want on your next SAT section score (800), and the number of days left until your next SAT (90).

The output features are the number of minutes to practice (length), and the number of days to practice those minutes (frequency). For example, if output feature #1 is 30 minutes and output feature #2 is 1 day, then you would practice 30 minutes per day. If the output features were 40 minutes and 2 days, then you would practice 40 minutes in total in the span of 2 days (40 minutes per 2 days). How many minutes to do on day one and day two is up to you.

Therefore, this example has three input features (current SAT section score, score you want on your next SAT section score, number of days left until next SAT) and two output features (time in minutes, and frequency in days).

As we learn about what happens in the cell body to create outputs of neurons, try to relate the inputs they receive to individual input features and the outputs to individual output features. This way, the process of understanding and remembering the new concepts and their details will become smoother.