



Deep Learning

Table of contents:

[What is a Deep Learning ?](#)
[Why use Machine Learning or Deep Learning ?](#)
[What Deep Learning is good for ?](#)
[What Deep Learning is not good for ? \(typically\)](#)
[Machine Learning vs. Deep Learning](#)
[Machine Learning and Deep Learning algorithms](#)
[What are the neural networks ?](#)
[Neural Networks](#)
[Anatomy of Neural Network](#)
[Types of Learning](#)
[What is Deep Learning actually used for ?](#)
[What is TensorFlow ?](#)
[Why using TensorFlow ?](#)
[What is GPU and TPU ?](#)
[What is a Tensor ?](#)

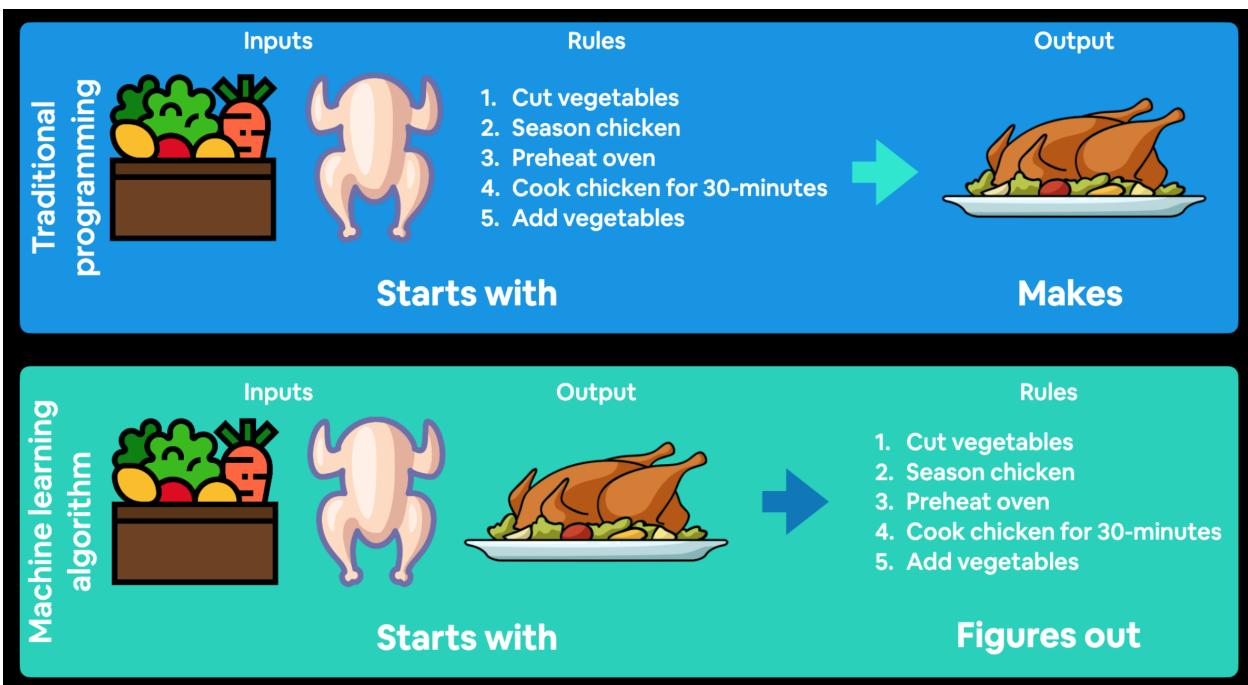
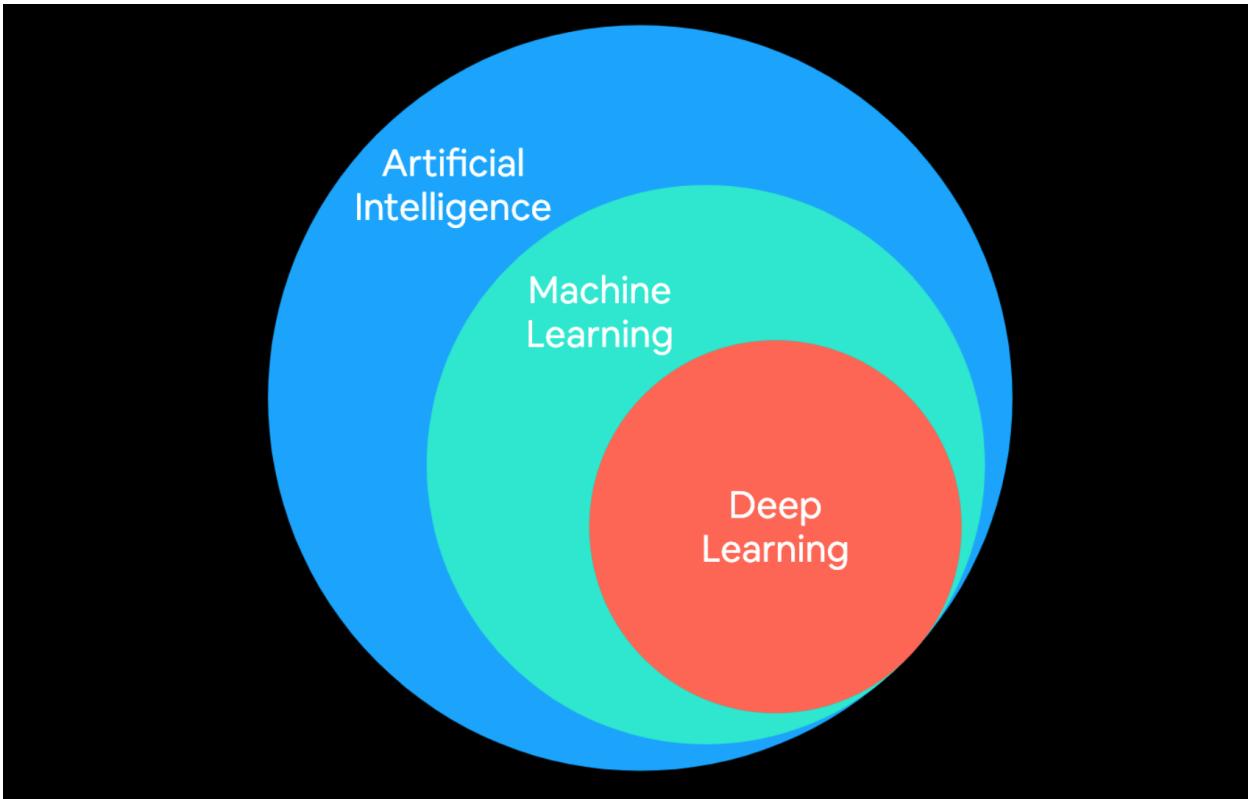
What is a Deep Learning ?



Wikipedia Info:

Deep learning is part of a broader family of machine learning methods based on artificial neural networks with representation learning.

Machine Learning is turning things (data) into numbers and finding patterns in those numbers.



- In **traditional programming**, we specify the rules: getting input, do some steps, get the result (output)

- In **Machine Learning**, we don't specify the rules, instead, we provide the input and ideal output to the Machine Learning algorithm. The algorithm will try to learn (come up with the rules), will try to figure out the ideal rules to get from the input to the output.

Why use Machine Learning or Deep Learning ?

For a complex problem (self-driving), can we think of all the rules?

We can use ML for literally anything as long as we can convert it into numbers and program it to find patterns. It could be anything any input or output.

(maybe not very simple...)

“If you can build a **simple rule-based system
that doesn’t require machine learning, do
that.”**

— A wise software engineer... (actually rule 1 of [Google’s Machine Learning Handbook](#))

Rules of Machine Learning: | ML Universal Guides | Google Developers

This document is intended to help those with a basic knowledge of machine learning get the benefit of Google's best practices in machine learning. It presents a style for machine learning, similar to the Google C++ Style Guide and other popular guides to

➡ <https://developers.google.com/machine-learning/guides/rules-of-ml?hl=es>

Google Developers

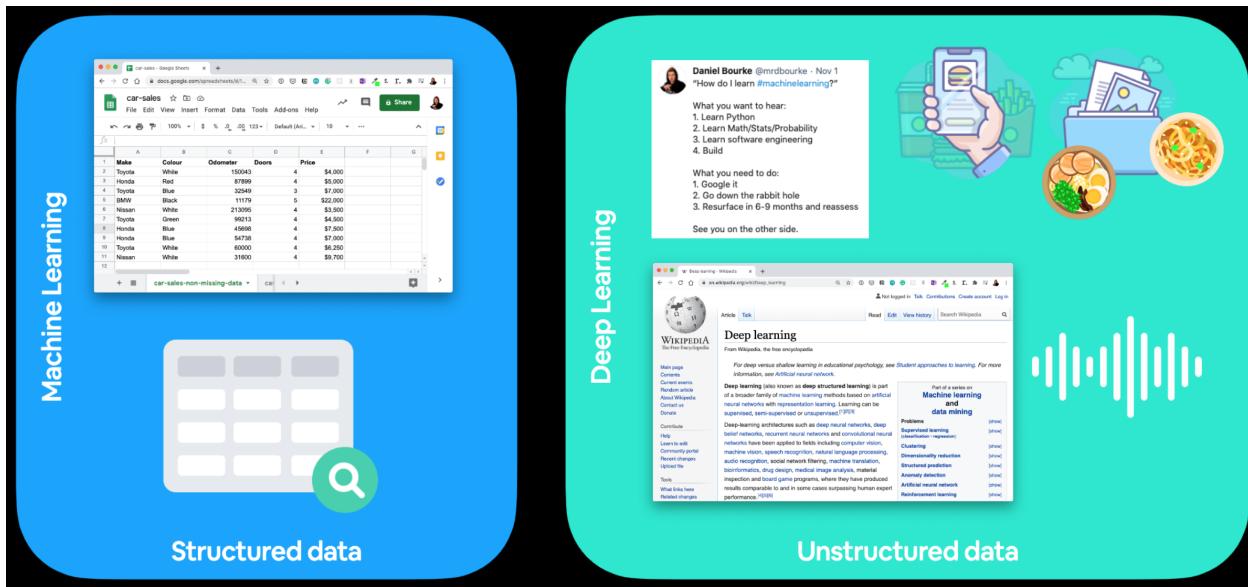
What Deep Learning is good for ?

- **Problems with long lists of rules** - when the traditional approach fails, machine learning/deep learning may help.
- **Continually changing environments** - deep learning can adapt ('learn') to new scenarios
- **Discovering insights within a large collections of data** - can you imagine trying to hard-craft the rules for what 101 different kinds of food look like ?

What Deep Learning is not good for ? (typically)

- **When we need explainability** - the patterns learned by deep learning model are typically uninterpretable by a human
- **When the traditional approach is a better option** - if we can accomplish what we need with a simple rule-based system
- **When errors are unacceptable** - since the outputs of deep learning aren't always predictable
- **When you don't have much data** - deep learning models usually require a fairly large amount of data to produce great results

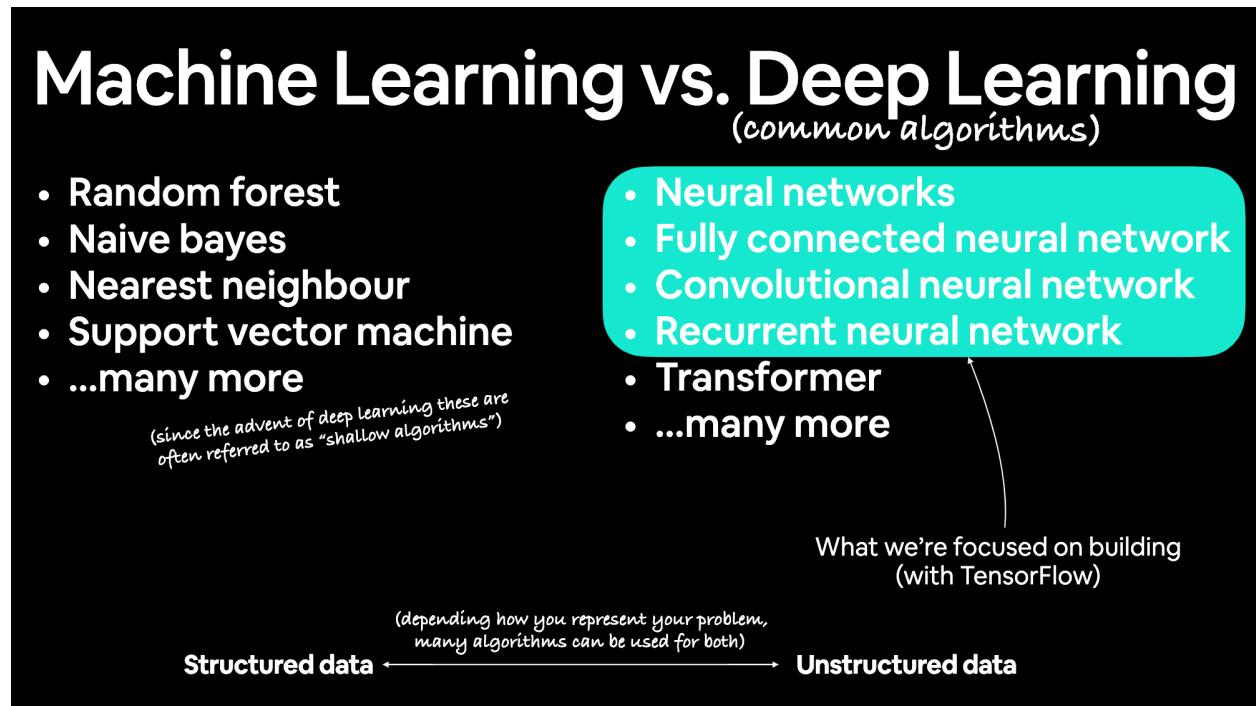
Machine Learning vs. Deep Learning



Machine Learning algorithms have typically performed best on **structured data** (tabular data)

Deep Learning typically performs best on unstructured data such as Natural Language Text, Images, Sounds

Machine Learning and Deep Learning algorithms



What are the neural networks ?

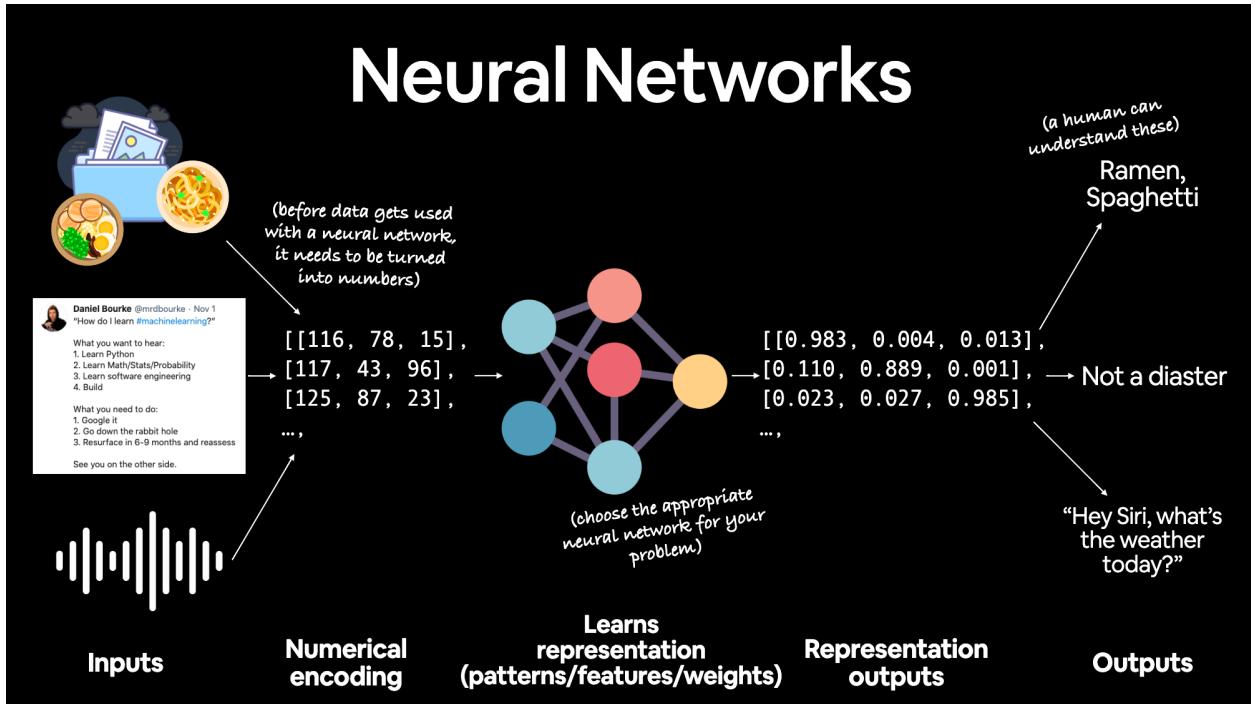


Wikipedia info:

A **neural network** is a network or circuit of neurons, or in a modern sense, an artificial neural network, composed of artificial neurons or nodes.

Neural Networks

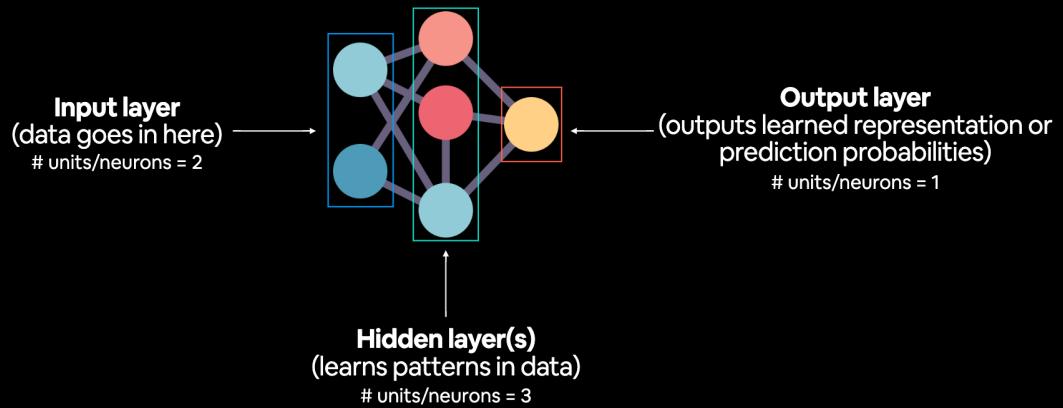
Neural Networks



- Inputs: Images, Text, Sounds
- We need to turn the input data into numbers, the process is called **numerical encoding**.
- Then, we feed the resulted numbers from numerical encoding to our neural network.
- The neural network will learn representation (patterns/features/weights). Depending on what problem we are working on, we have to choose an appropriate neural network.
- The neural network is going to create representation outputs. The numbers have been transformed. They represent the patterns that our neural network model has defined (found), patterns in numerical representation form.
- Finally, it's up to us to take there representation outputs that our neural network models has learned about our data, and convert it to human-understandable output.

Anatomy of Neural Network

Anatomy of Neural Networks



Note: “patterns” is an arbitrary term, you’ll often hear “embedding”, “weights”, “feature representation”, “feature vectors” all referring to similar things.

Types of Learning

Types of Learning



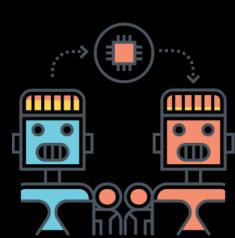
Supervised Learning



Semi-supervised Learning



Unsupervised Learning

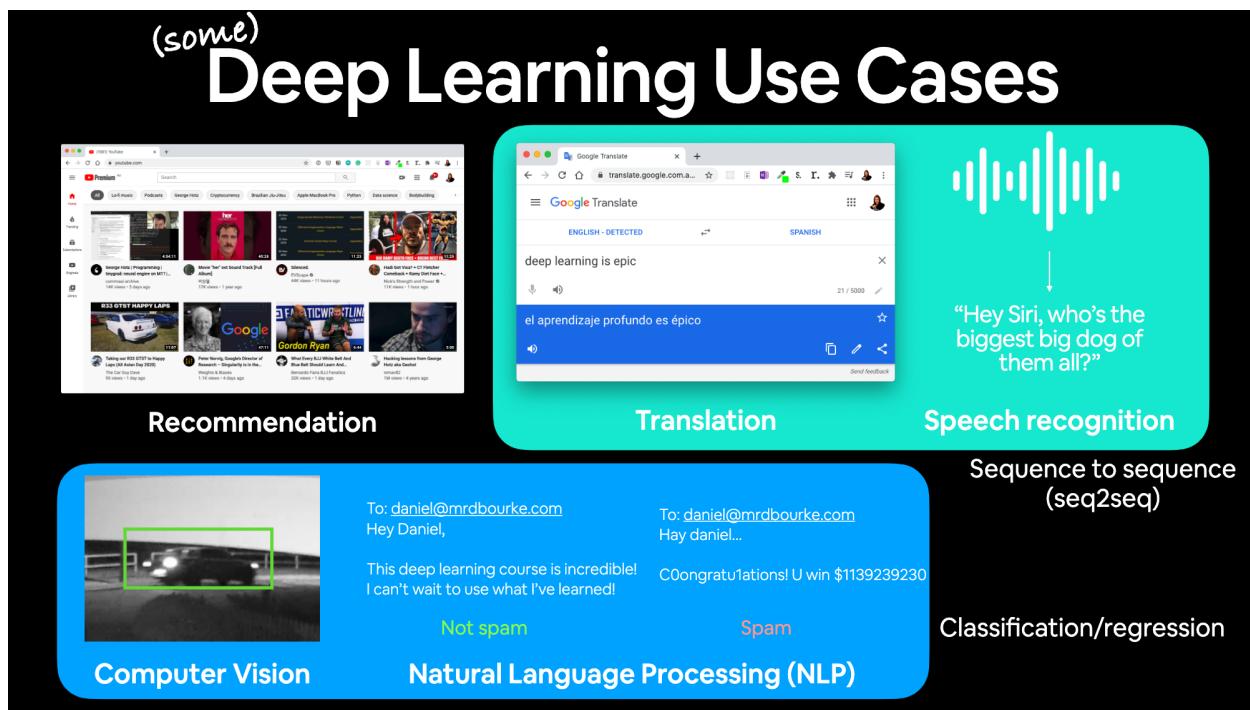


Transfer Learning

We'll be writing code to do these

- Supervised Learning often involves having data and labels. Image Food identification: data - Image of the pizza, label - pizza.
- Semi-supervised Learning has some data, or have data like in Supervised Learning but only has some labels. For example, we have 10,000 images of the food and we know only 1000 labels of those images. We train the neural network model on the data that has a label, then we use it to predict the label for rest of the data (that don't have a label).
- Unsupervised Learning - we only have a data, but we don't have labels. To find patterns in the data
- Transfer Learning - taking what one Deep Learning model has learned on some set of data, and then use it on the problem on another set of data. For example food identification, we can take the deep learning model has learned about the world, and use it in our problem (identifying food images)

What is Deep Learning actually used for ?



DeepMind - What if solving one problem could unlock solutions to thousands more?

Artificial intelligence could be one of humanity's most useful inventions. We research and build safe AI systems that learn how to solve problems and advance scientific discovery for all.

 <https://deepmind.com/>



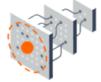
What is TensorFlow ?

What is TensorFlow?

- End-to-end platform for machine learning
- Write fast deep learning code in Python/other accessible languages (able to run on a GPU/TPU)
- Able to access many pre-built deep learning models (TensorFlow Hub)
- Whole stack: preprocess data, model data, deploy model in your application
- Originally designed and used in-house by Google (now open-source)

Why using TensorFlow ?

Why TensorFlow?



Easy model building

Build and train ML models easily using intuitive high-level APIs like Keras with eager execution, which makes for immediate model iteration and easy debugging.



Robust ML production anywhere

Easily train and deploy models in the cloud, on-prem, in the browser, or on-device no matter what language you use.



Powerful experimentation for research

A simple and flexible architecture to take new ideas from concept to code, to state-of-the-art models, and to publication faster.

Source: TensorFlow.org

What we'll be doing (lots of)

and a little bit of this...

TensorFlow

An end-to-end open source machine learning platform for everyone. Discover TensorFlow's flexible ecosystem of tools, libraries and community resources.

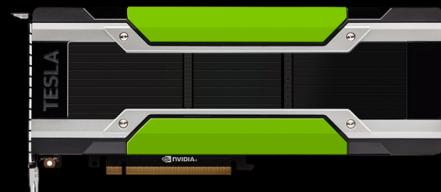
 <https://www.tensorflow.org/>



TensorFlow

What is GPU and TPU ?

What is a GPU/TPU?



GPU (Graphics Processing Unit)



TPU (Tensor Processing Unit)

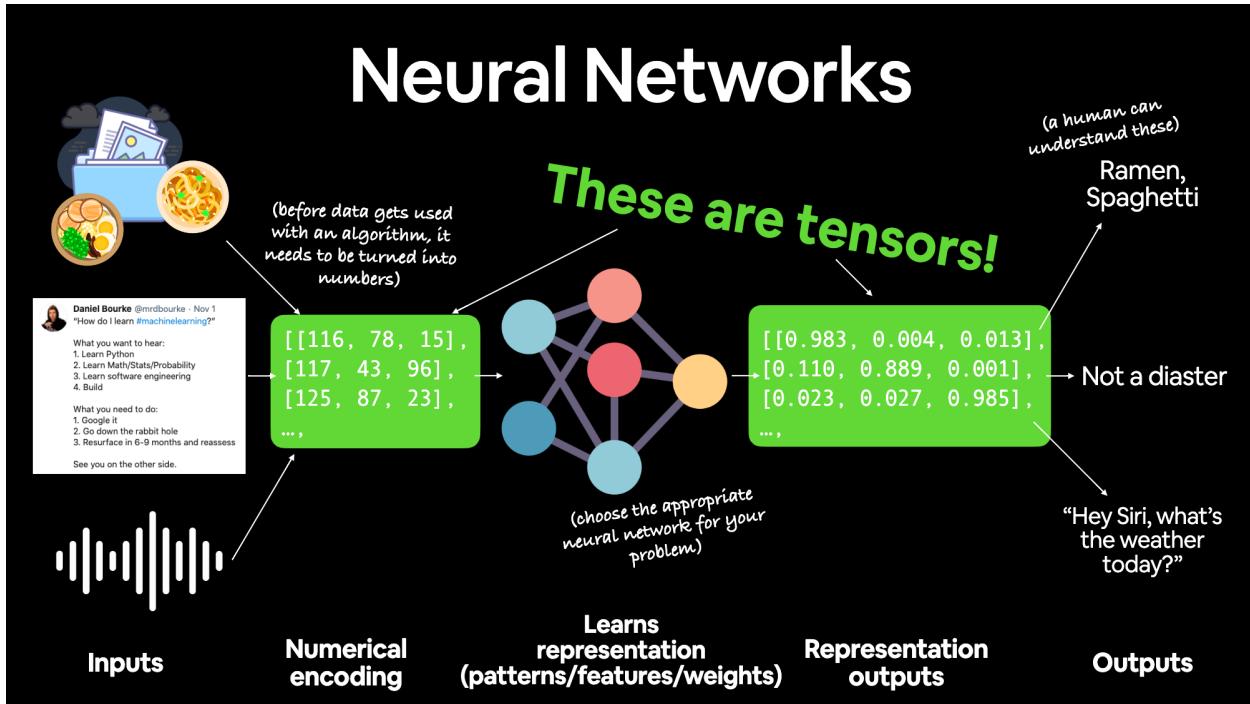


Wikipedia info:

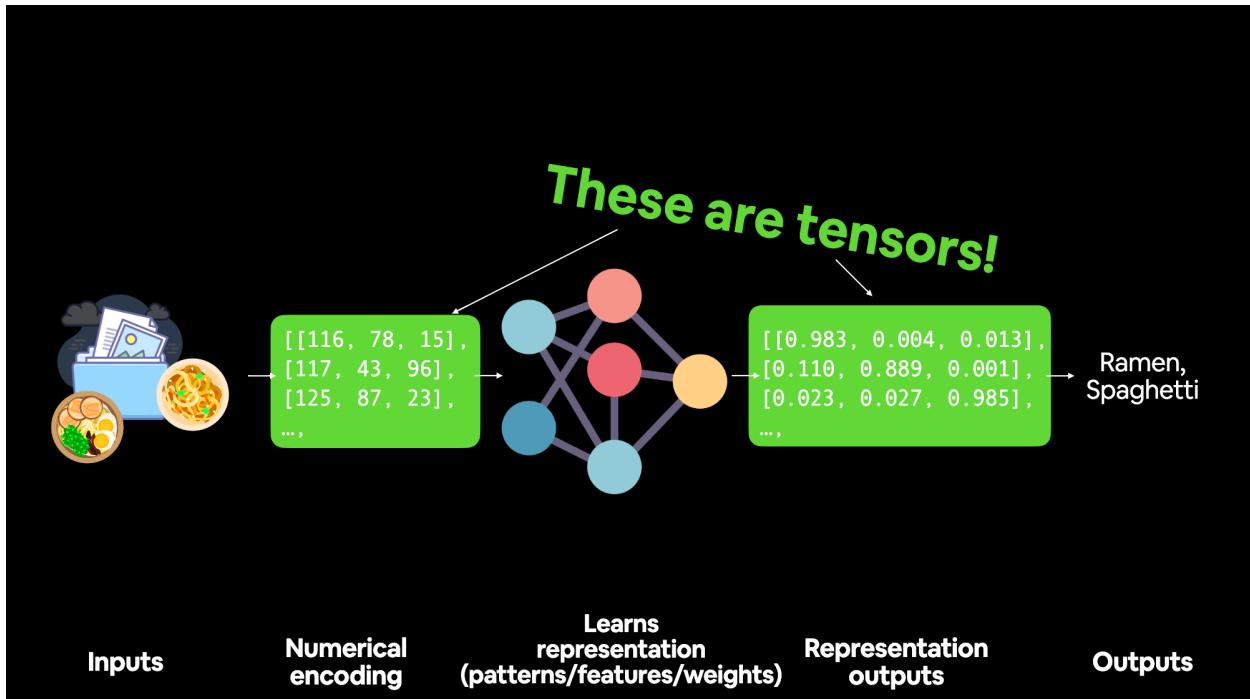
Tensor Processing Unit (TPU) is an AI accelerator application-specific integrated circuit (ASIC) developed by Google specifically for neural network machine learning, particularly using Google's own TensorFlow software.

What is a Tensor ?

Neural Networks



Tensor - Flow



Tensor - is a some numerical way to represent information.

<https://www.youtube.com/watch?v=f5liqUk0ZTw>