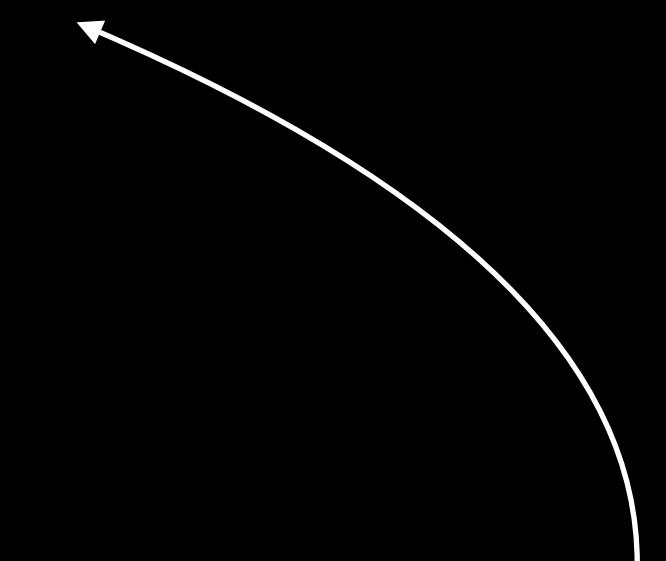




Deep Learning with TensorFlow

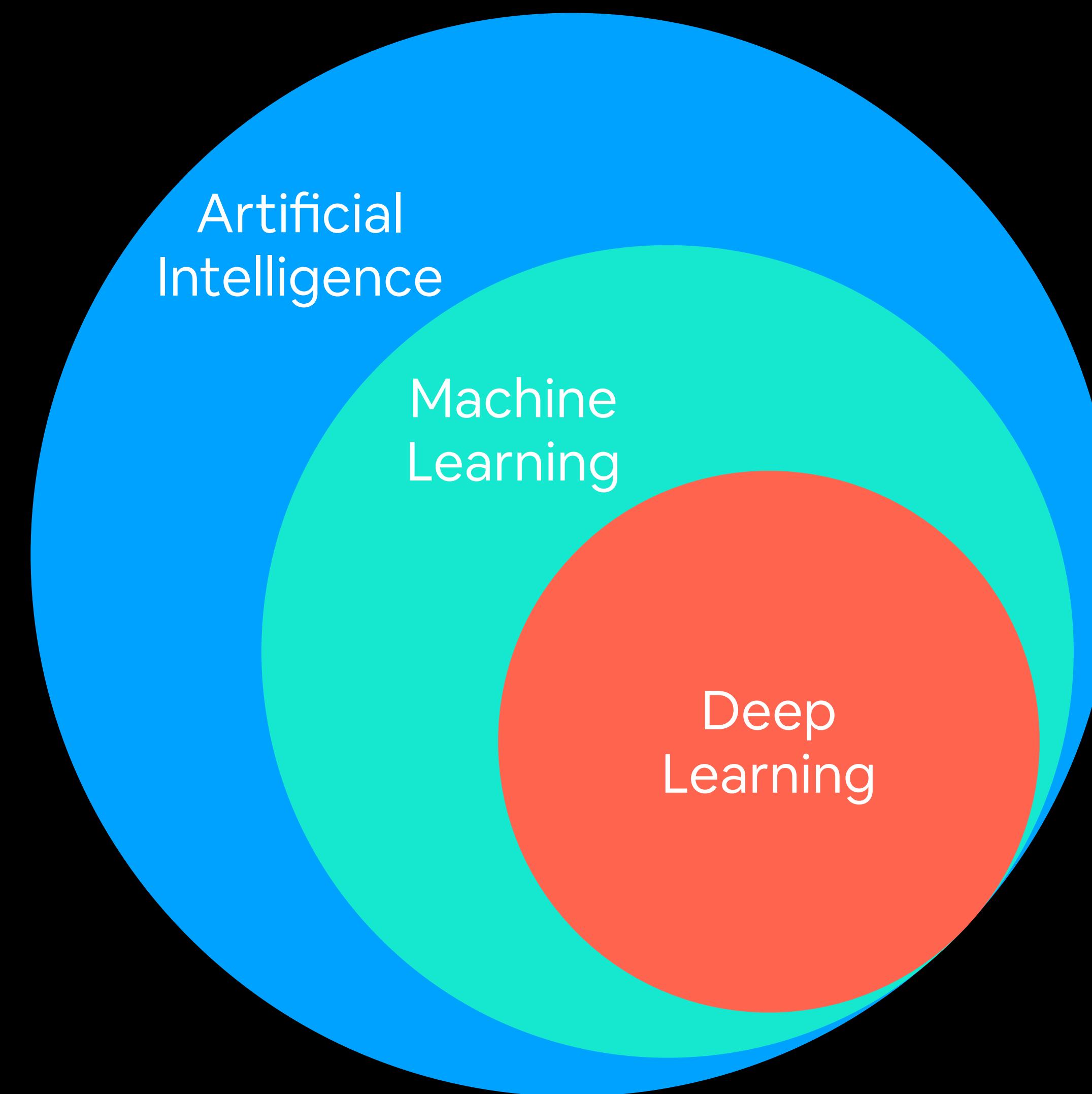
“What is deep learning?”

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



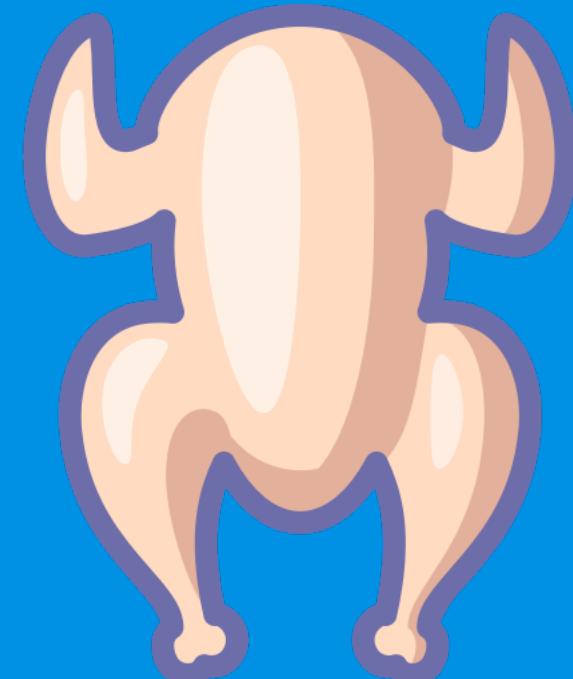
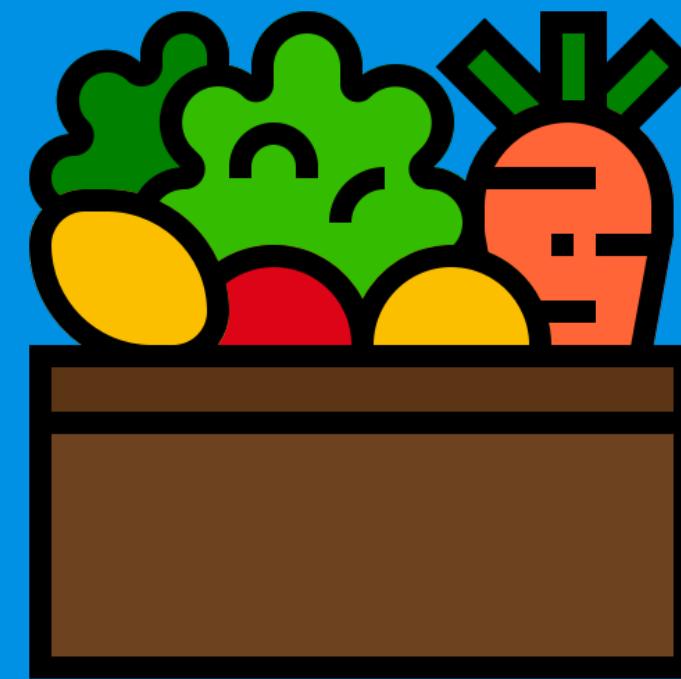
The computer does this part.
How?
Code & math.
We're going to be writing the code.

Machine Learning vs. Deep Learning



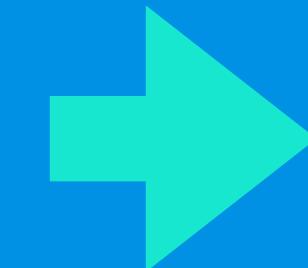
Traditional programming

Inputs

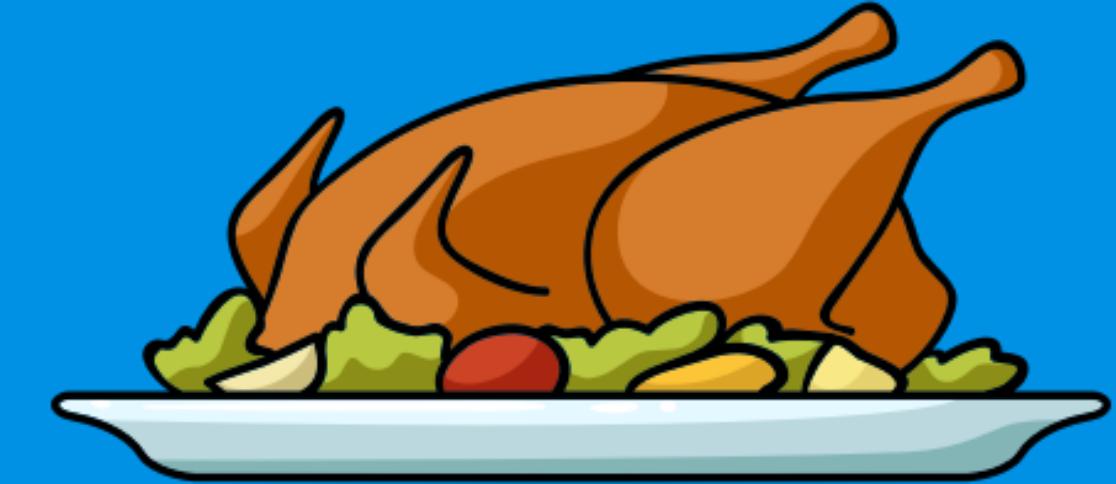


Rules

1. Cut vegetables
2. Season chicken
3. Preheat oven
4. Cook chicken for 30-minutes
5. Add vegetables



Output

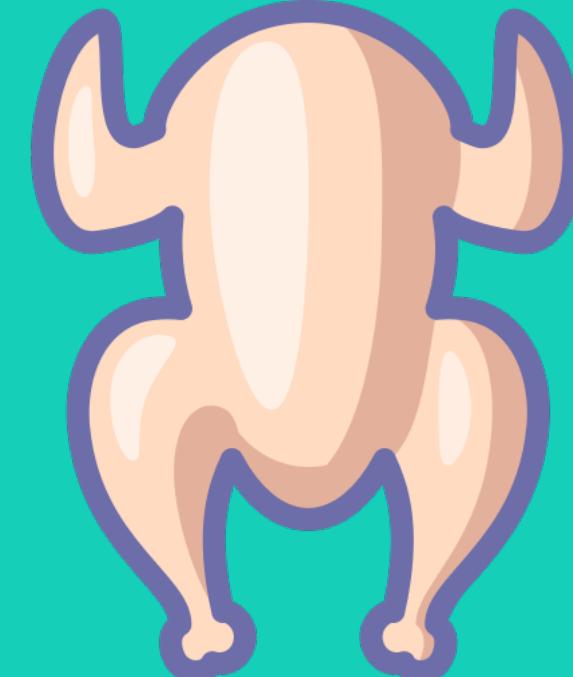
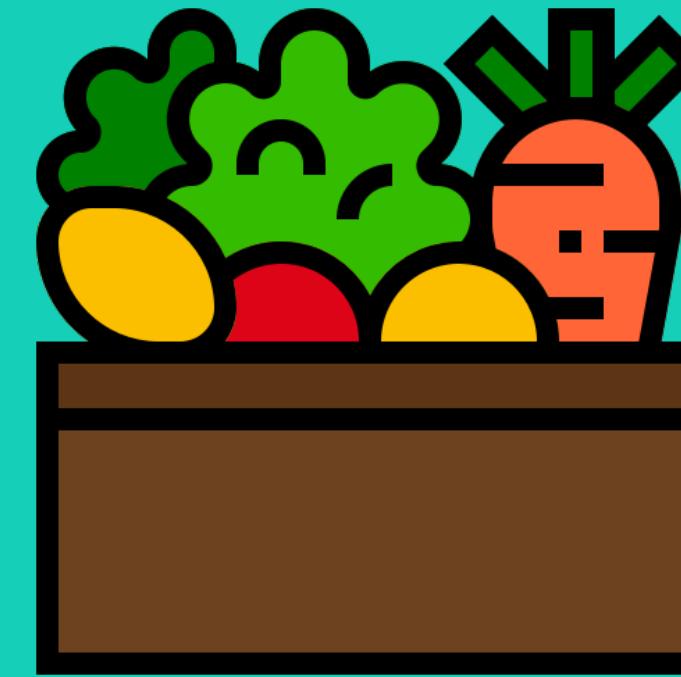


Starts with

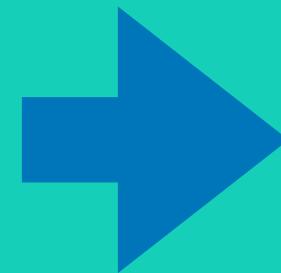
Makes

Machine learning algorithm

Inputs



Output



Rules

1. Cut vegetables
2. Season chicken
3. Preheat oven
4. Cook chicken for 30-minutes
5. Add vegetables

Starts with

Figures out

“Why use machine learning (or
deep learning)?”

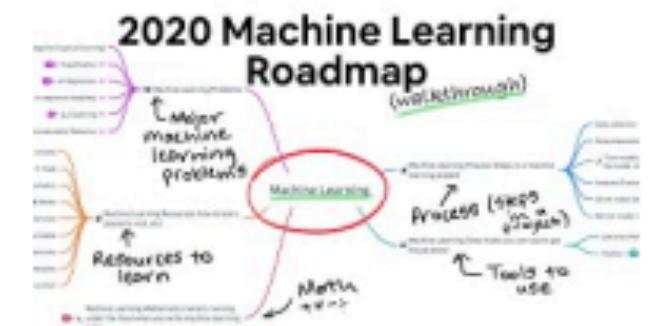
Good reason: ~~Why not?~~

Better reason: For a complex
problem, can you think of all the rules?
(probably not)



YouTube

Yashaswi Kulshreshtha commented on your video



2020 Machine Learning Roadmap



Yashaswi Kulshreshtha

I think you can use ML for literally anything as long as you can convert it into numbers and program it to find patterns. Literally it could be anything any input or output from the universe

Source: [2020 Machine Learning Roadmap video.](#)

(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)

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 large collections of data**—can you imagine trying to hand-craft rules for what 101 different kinds of food look like?

What deep learning is not good for (typically)



- **When you need explainability**—the patterns learned by a deep learning model are typically uninterpretable by a human.
- **When the traditional approach is a better option** — if you can accomplish what you need with a simple rule-based system.
- **When errors are unacceptable** — since the outputs of deep learning model 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.

(though we'll see how to get great results without huge amounts of data)

Machine Learning vs. Deep Learning

Machine Learning

Make	Colour	Odometer	Doors	Price
Toyota	White	150043	4	\$4,000
Honda	Red	87899	4	\$5,000
Toyota	Blue	32549	3	\$7,000
BMW	Black	11179	5	\$22,000
Nissan	White	213095	4	\$3,500
Toyota	Green	99213	4	\$4,500
Honda	Blue	45698	4	\$7,500
Honda	Blue	54738	4	\$7,000
Toyota	White	60000	4	\$6,250
Nissan	White	31600	4	\$9,700



Structured data

Deep Learning

WIKIPEDIA The Free Encyclopedia

Deep learning

From Wikipedia, the free encyclopedia

For deep versus shallow learning in educational psychology, see [Student approaches to learning](#). For more information, see [Artificial neural network](#).

Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.^{[1][2][3]}

Deep-learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, machine vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance.^{[4][5][6]}

Part of a series on

Machine learning and data mining

Problems

- Supervised learning (classification · regression)
- Clustering
- Dimensionality reduction
- Structured prediction
- Anomaly detection
- Artificial neural network
- Reinforcement learning

Daniel Bourke @mrdbourke · Nov 1
"How do I learn #machinelearning?"

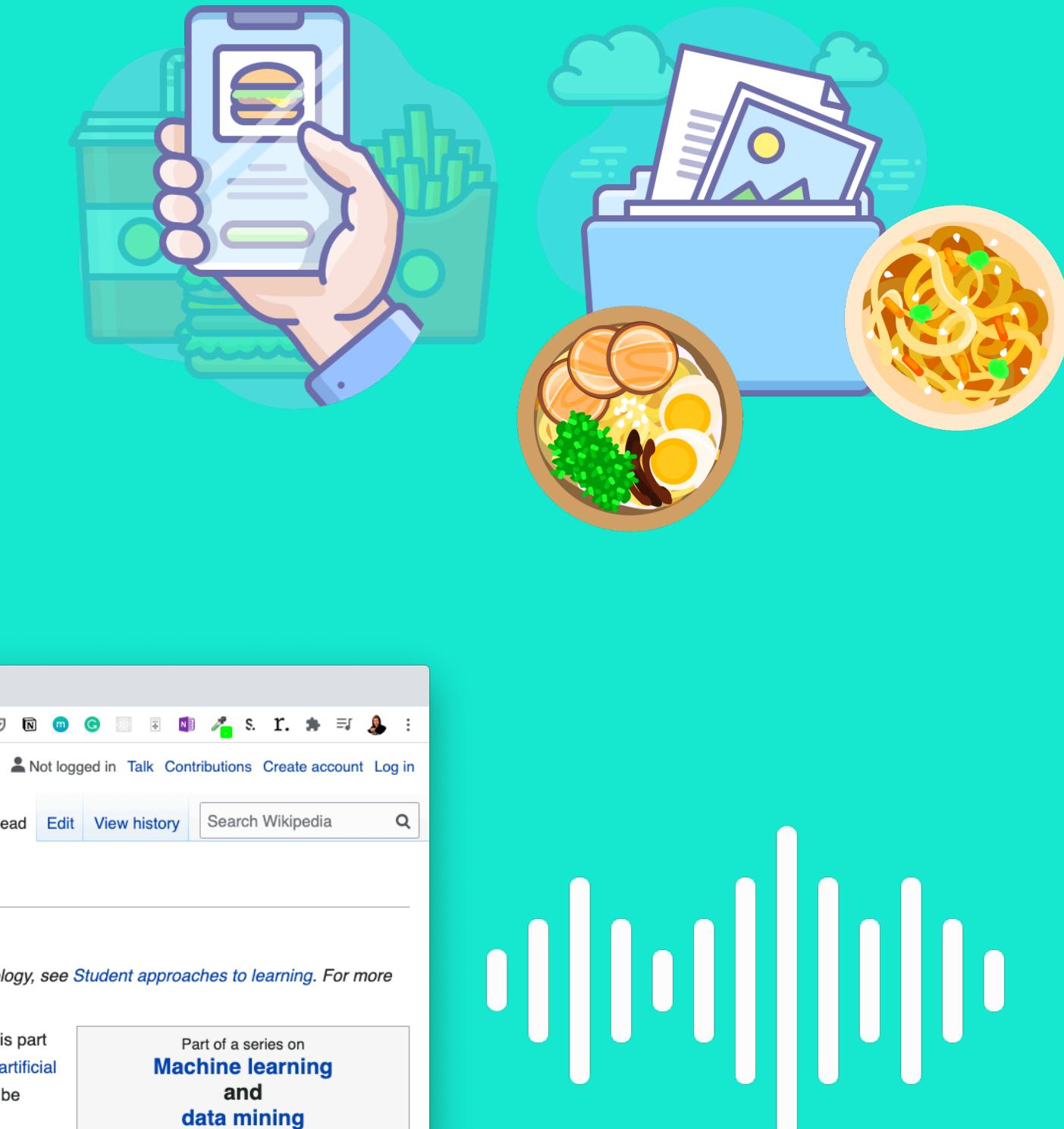
What you want to hear:

1. Learn Python
2. Learn Math/Stats/Probability
3. Learn software engineering
4. Build

What you need to do:

1. Google it
2. Go down the rabbit hole
3. Resurface in 6-9 months and reassess

See you on the other side.



Unstructured data

Machine Learning vs. Deep Learning

(common algorithms)

- Random forest
- Naive bayes
- Nearest neighbour
- Support vector machine
- ...many more

(since the advent of deep learning these are often referred to as "shallow algorithms")

- Neural networks
- Fully connected neural network
- Convolutional neural network
- Recurrent neural network
- Transformer
- ...many more

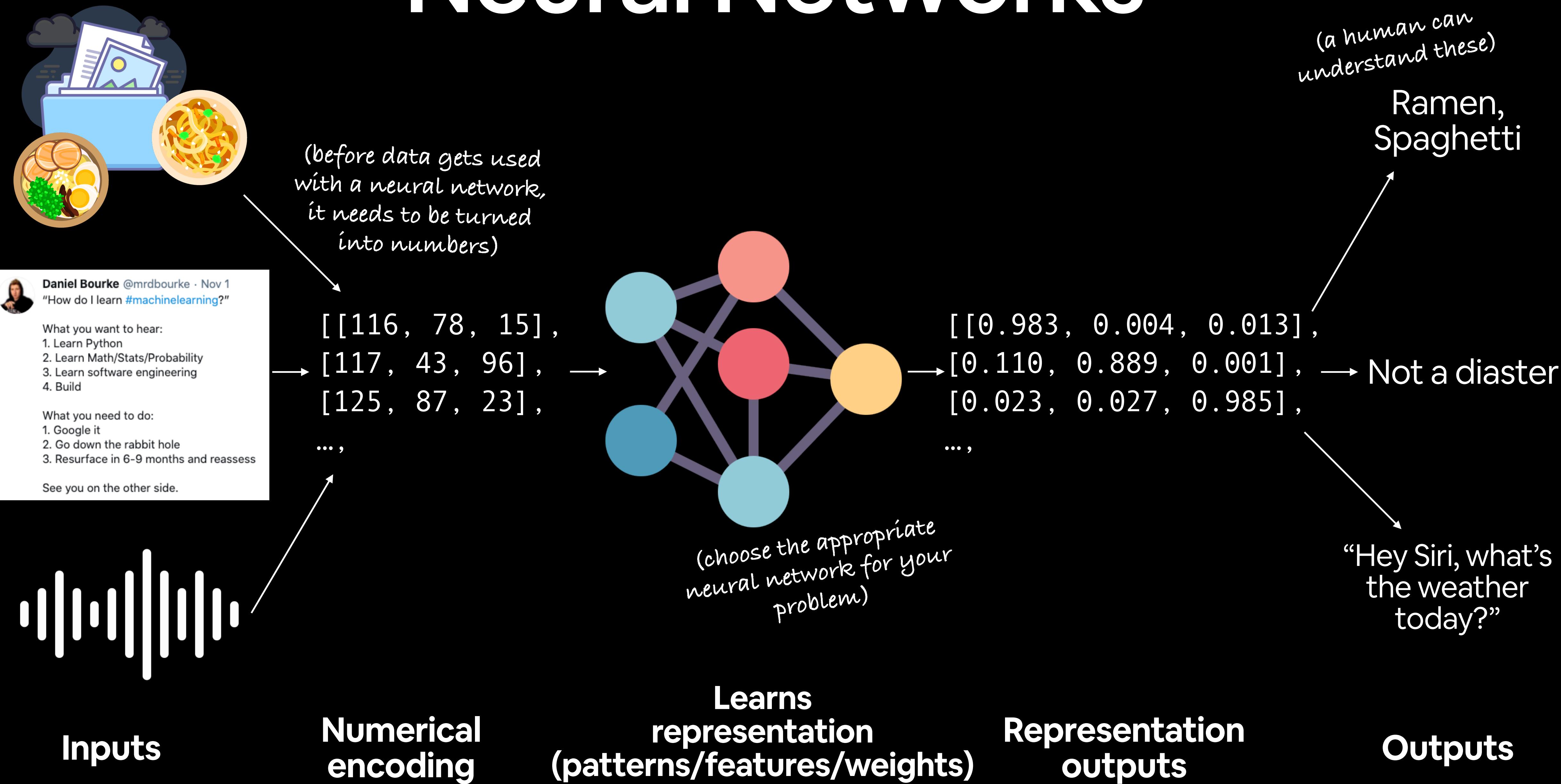
What we're focused on building
(with TensorFlow)

Structured data ← → Unstructured data

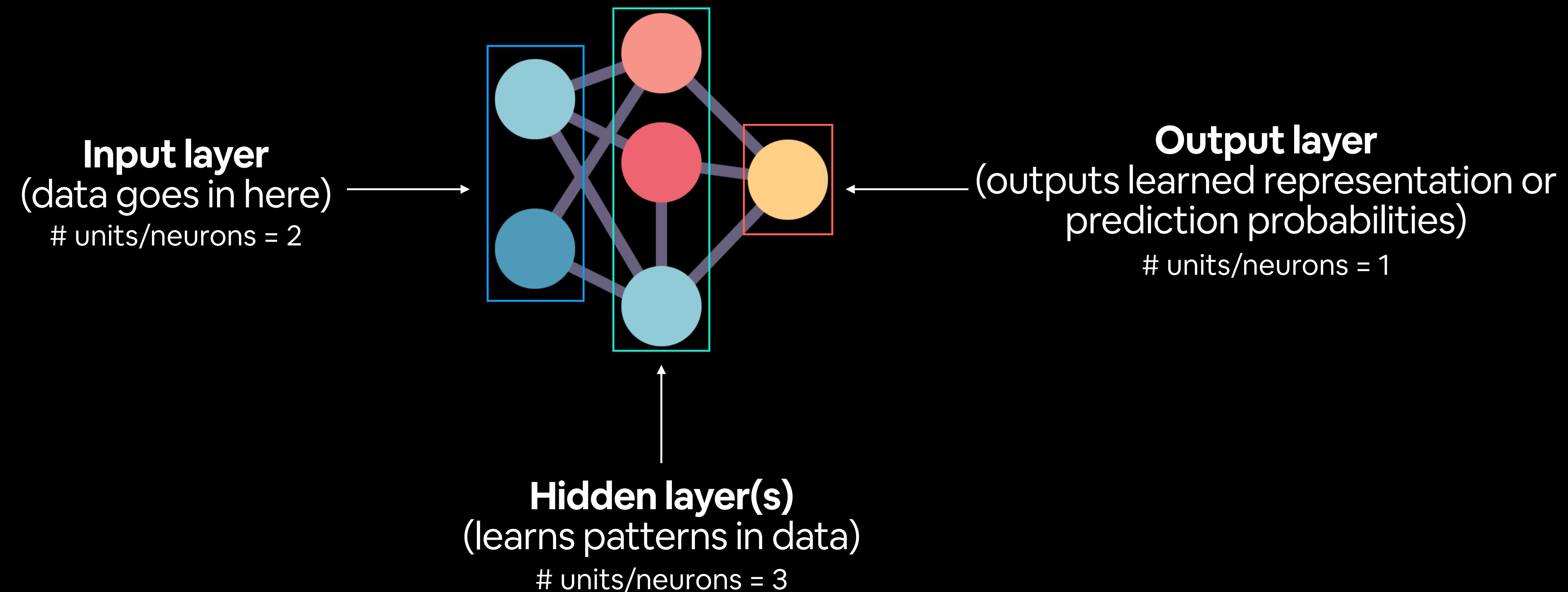
(depending how you represent your problem,
many algorithms can be used for both)

“What are neural networks?”

Neural Networks



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



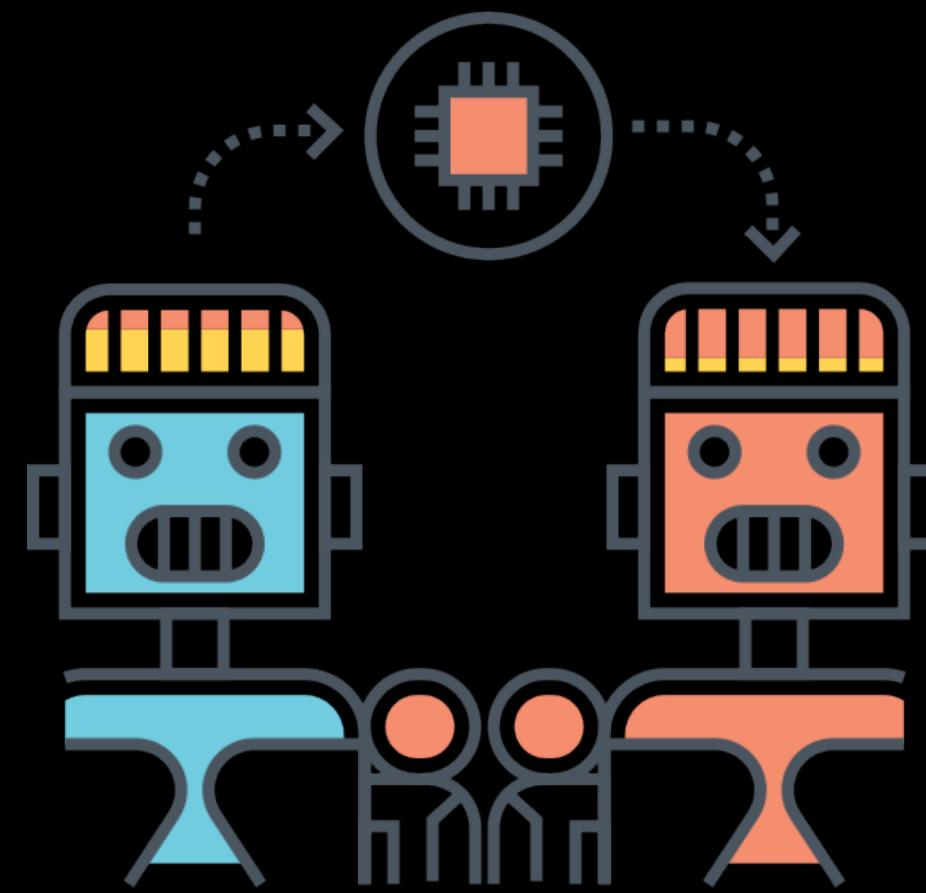
Supervised
Learning



Semi-supervised
Learning



Unsupervised
Learning



Transfer
Learning

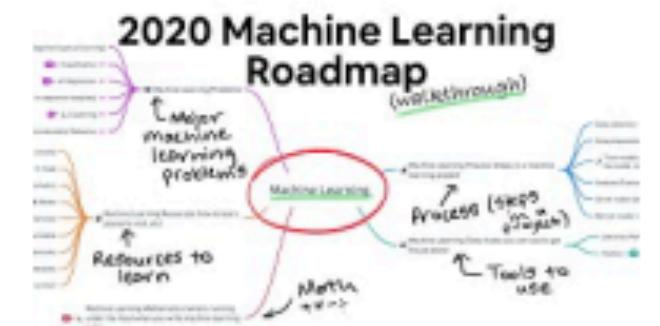
We'll be writing code to do these

“What is deep learning actually
used for?”



YouTube

Yashaswi Kulshreshtha commented on your video



2020 Machine Learning Roadmap

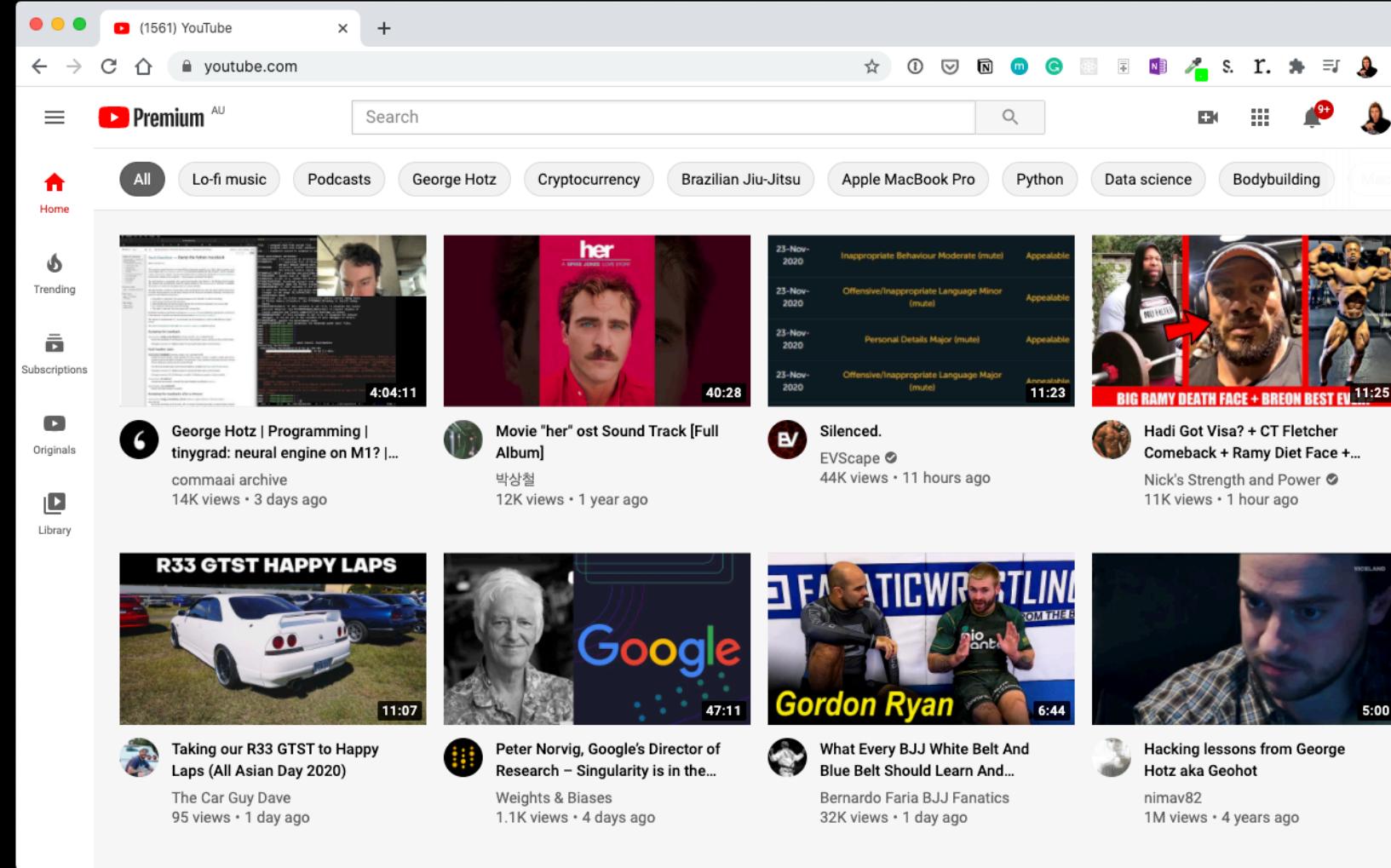


Yashaswi Kulshreshtha

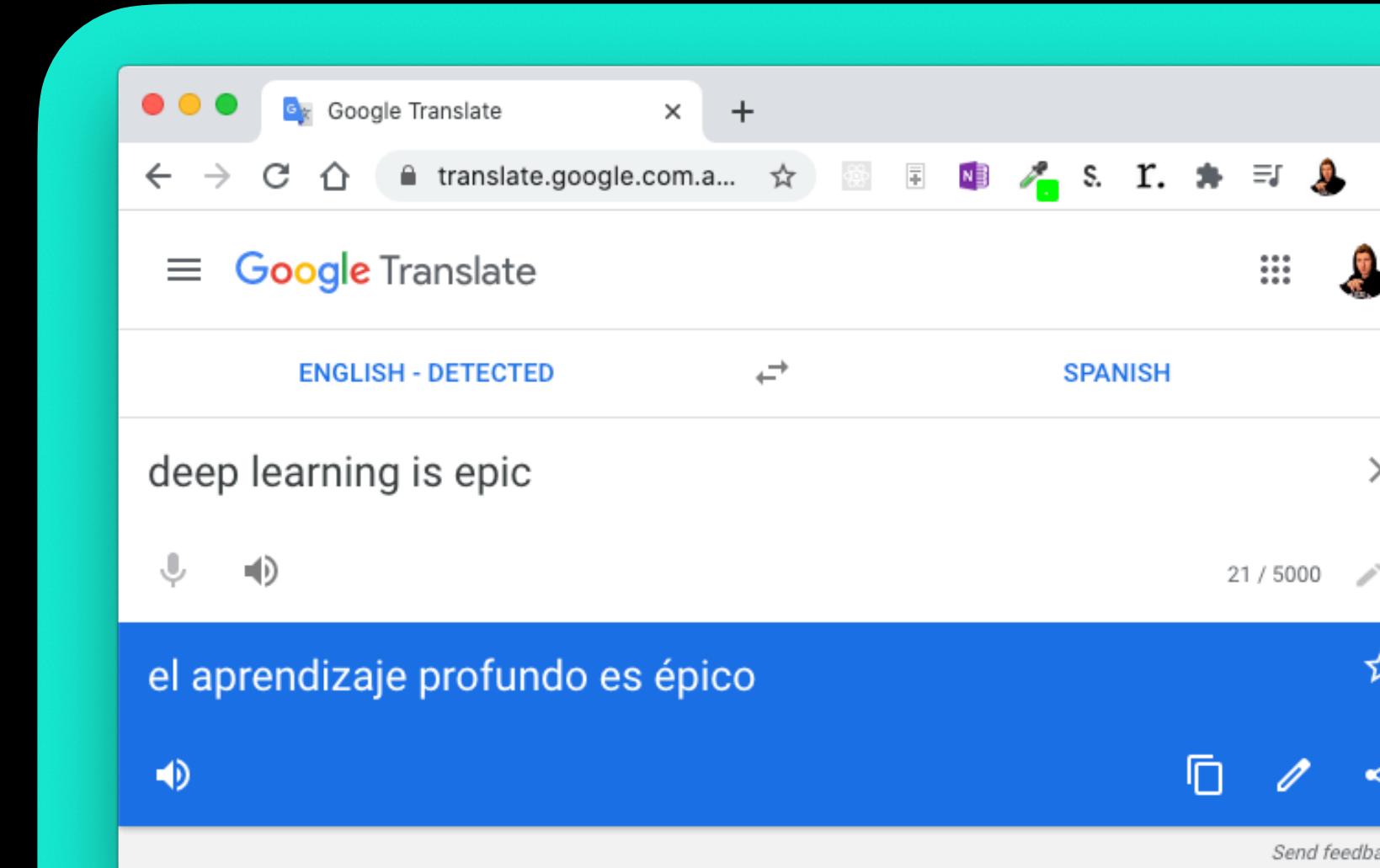
I think you can use ML for literally anything as long as you can convert it into numbers and program it to find patterns. Literally it could be anything any input or output from the universe

Source: [2020 Machine Learning Roadmap video](#).

(some) Deep Learning Use Cases



Recommendation



Translation



"Hey Siri, who's the biggest big dog of them all?"

Speech recognition



Computer Vision

To: daniel@mrdourke.com
Hey Daniel,

This deep learning course is incredible!
I can't wait to use what I've learned!

Not spam

To: daniel@mrdourke.com
Hay daniel...

Congratulations! U win \$1139239230

Spam

Sequence to sequence (seq2seq)

Classification/regression

Natural Language Processing (NLP)



TensorFlow

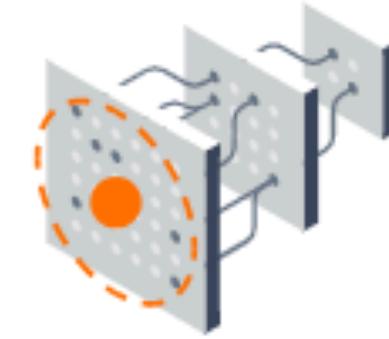
“What is TensorFlow?”



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 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

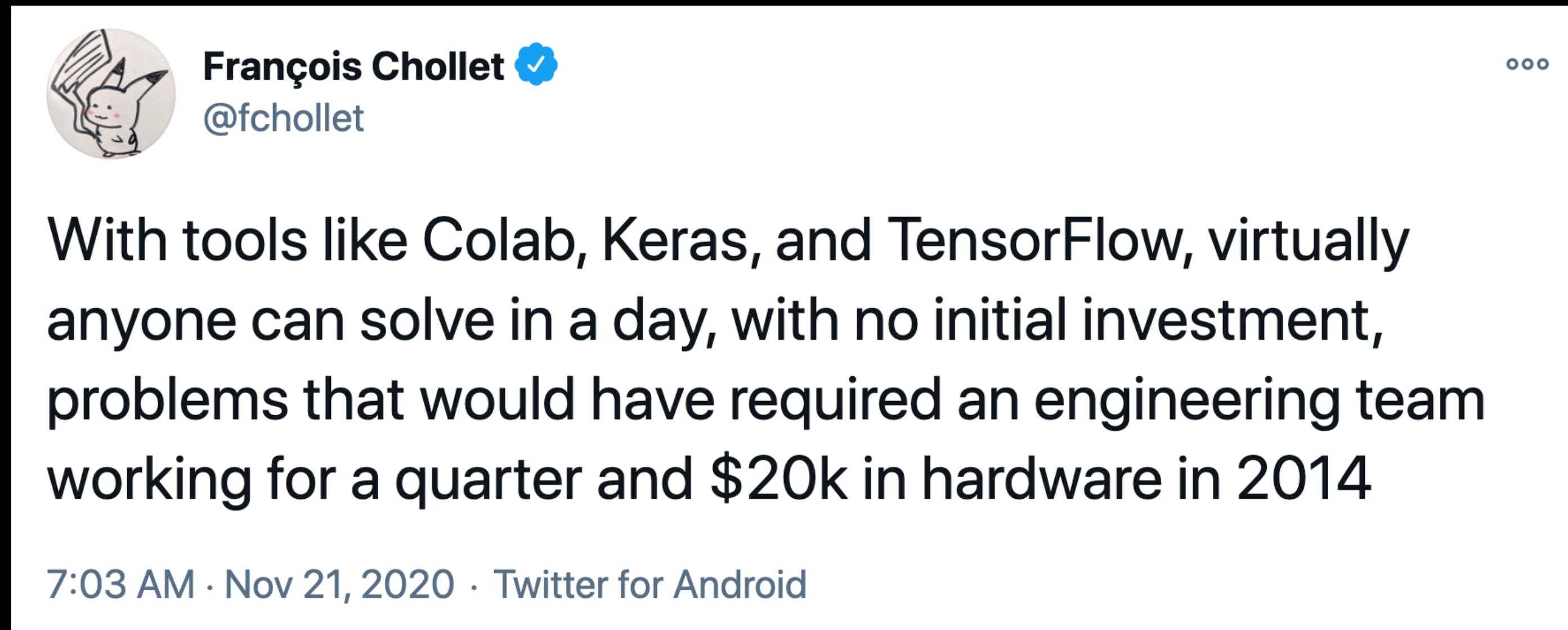
A simple and flexible architecture to take new ideas from concept to code, to state-of-the-art models, and to publication faster.

Source: TensorFlow.org

What we'll be doing (lots of)

and a little bit of this...

Why TensorFlow?



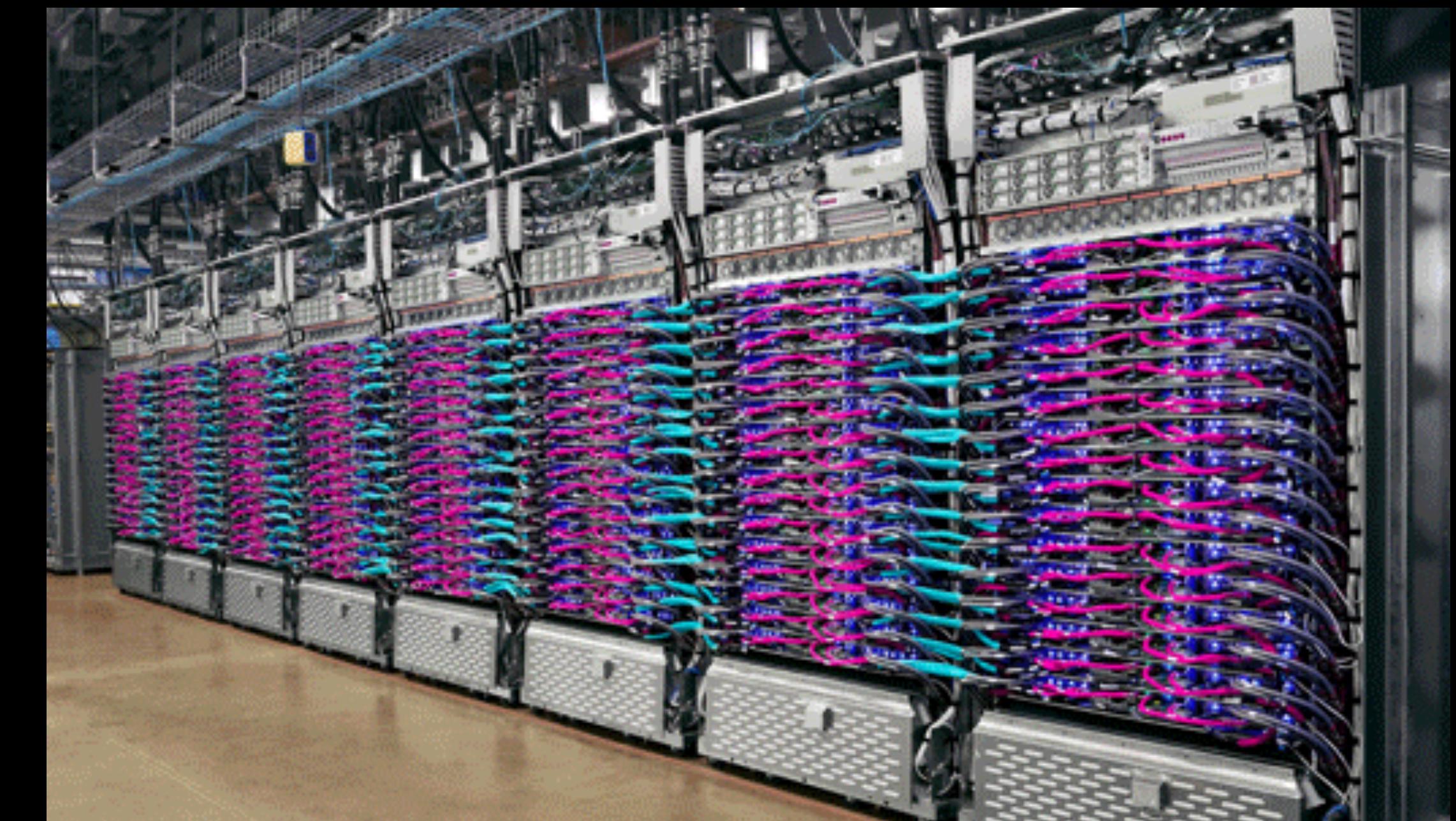
A screenshot of a Twitter post from François Chollet (@fchollet). The post features a profile picture of a cartoon character holding a book. The text reads: "With tools like Colab, Keras, and TensorFlow, virtually anyone can solve in a day, with no initial investment, problems that would have required an engineering team working for a quarter and \$20k in hardware in 2014". The timestamp at the bottom left is "7:03 AM · Nov 21, 2020 · Twitter for Android".

Source: [@fchollet Twitter](#)

What is a GPU/TPU?



GPU (Graphics Processing Unit)



TPU (Tensor Processing Unit)

“What is a tensor?”

Neural Networks

These are tensors!



Inputs

Numerical encoding

(before data gets used with an algorithm, it needs to be turned into numbers)

```
[[116, 78, 15],  
 [117, 43, 96],  
 [125, 87, 23],  
 ... ,
```



(choose the appropriate neural network for your problem)

Learns representation
(patterns/features/weights)

Representation outputs

```
[[0.983, 0.004, 0.013],  
 [0.110, 0.889, 0.001],  
 [0.023, 0.027, 0.985],  
 ... ,
```

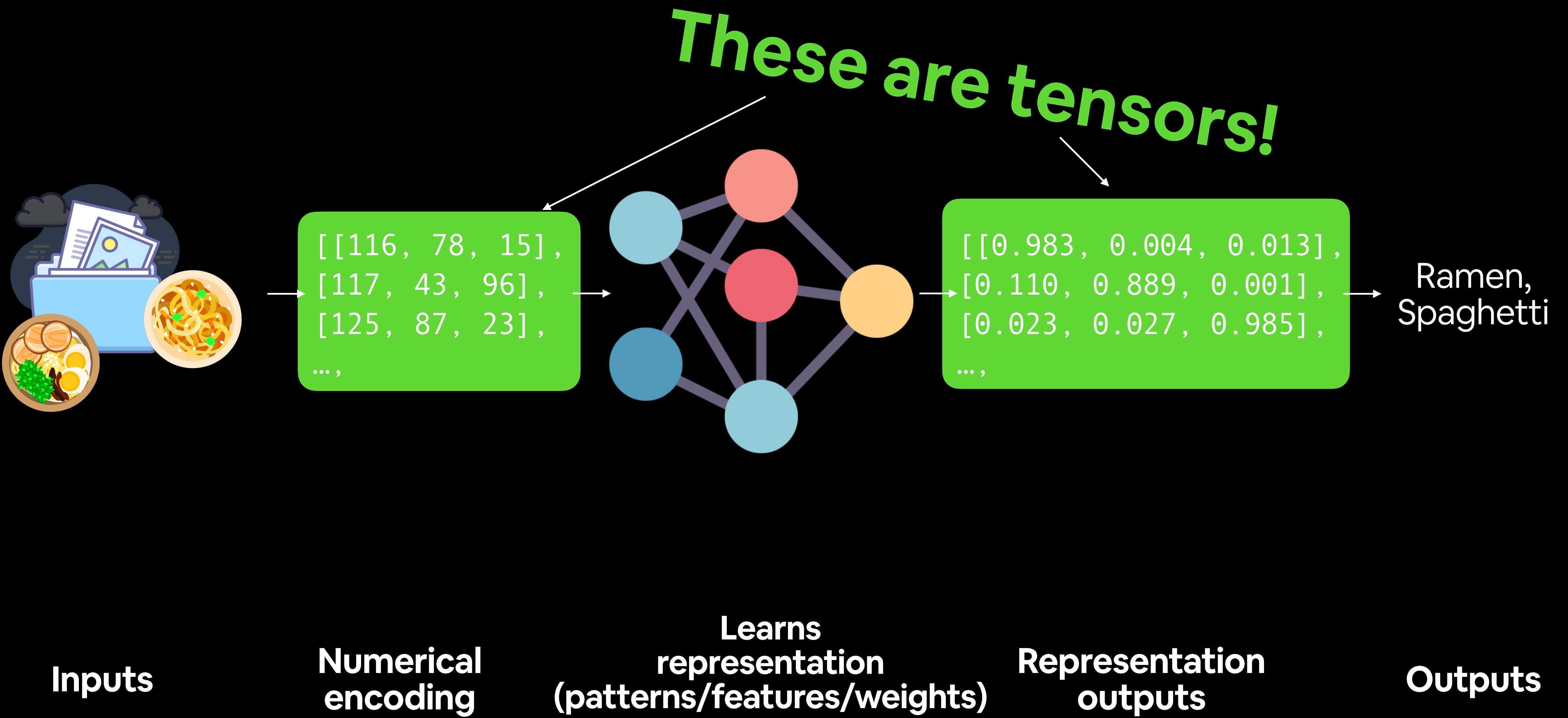
(a human can understand these)

Ramen,
Spaghetti

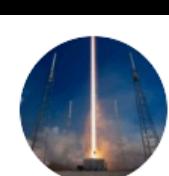
Not a disaster

"Hey Siri, what's the weather today?"

Outputs



“What are we going to
cover?”



Elon Musk

@elonmusk

...

Deus ex machine learning

**LEARNING ML,DL
FROM UNIVERSITY**

ONLINE COURSES

FROM YOUTUBE

FROM ARTICLES

FROM MEMES



8:07 AM · Nov 18, 2020 · Twitter for iPhone

14.9K Retweets 2.3K Quote Tweets 188.4K Likes

Source: @elonmusk Twitter

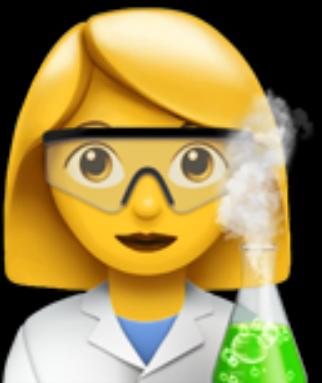
What we're going to cover

(broadly)

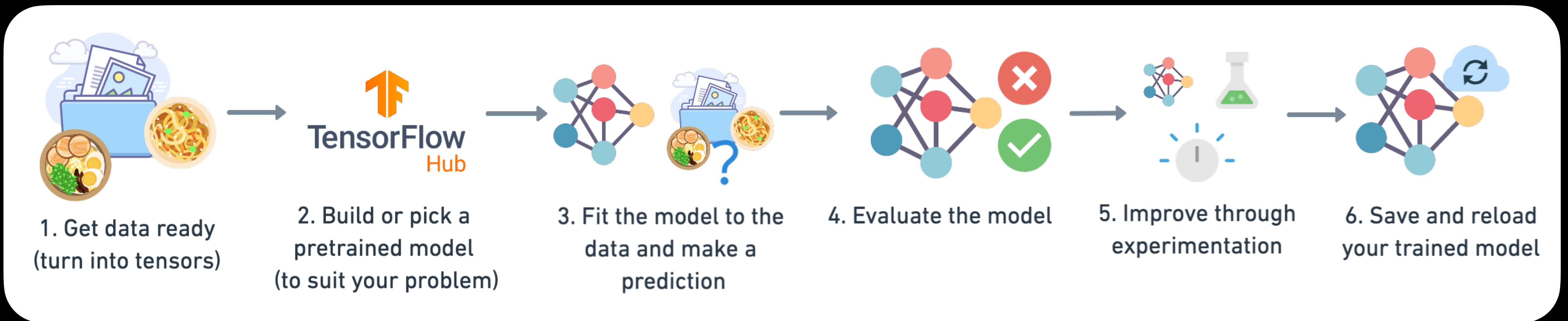
- TensorFlow basics & fundamentals
- Preprocessing data (getting it into tensors)
- Building and using pretrained deep learning models
- Fitting a model to the data (learning patterns)
- Making predictions with a model (using patterns)
- Evaluating model predictions
- Saving and loading models
- Using a trained model to make predictions on custom data

(we'll be cooking up lots of code!)

How:



What we're going to cover



A TensorFlow workflow

“How should I approach
this course?”

How to approach this course

- Write code (lots of it, follow along, let's make mistakes together)
 - Motto #1: “If in doubt, run the code”
- Explore & experiment
 - Motto #2: “Experiment, experiment, experiment”
 - Motto #3: “Visualize, visualize, visualize” (recreate things in ways you can understand them)
- Ask questions (including the “dumb” ones) 🤔
- Do the exercises (try them yourself before looking at the solutions) 🔧
 - This course doesn't cover everything, if you want to learn more on something, look it up
- Share your work
- ❌ Avoid:
 - Overthinking the process
 - The “I can't learn it” mentality (that's bullsh*t)

Let's code!

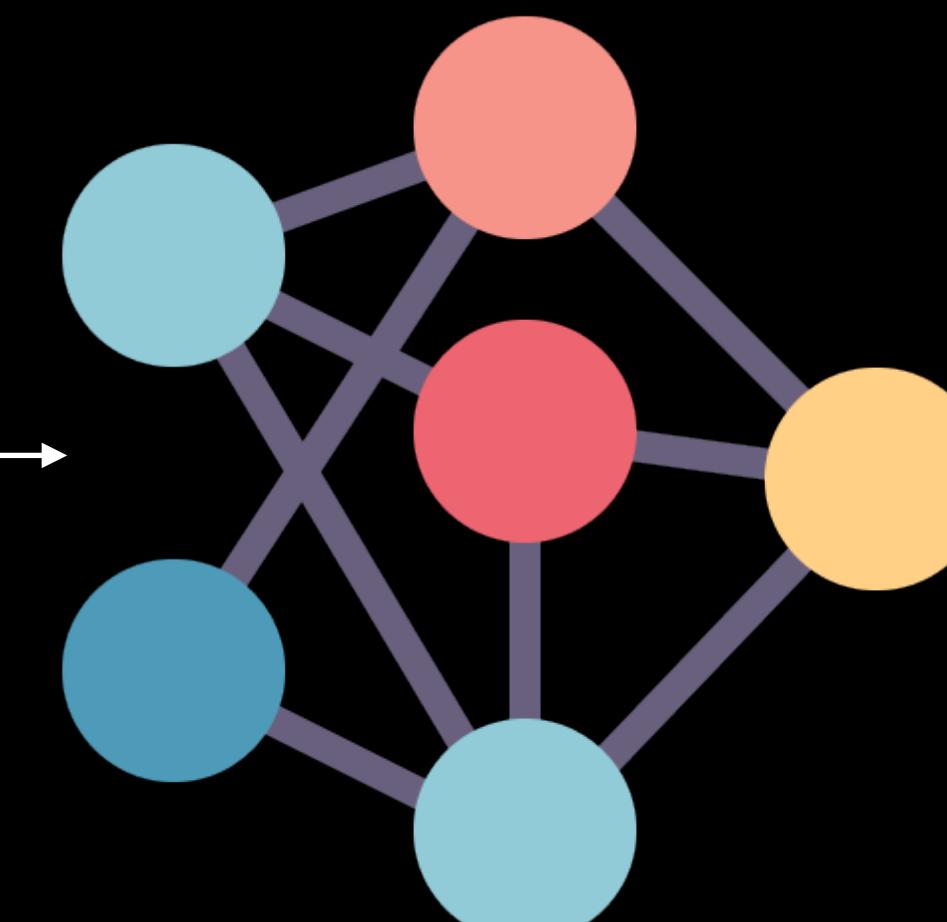
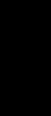
2. Show examples



1. Initialise with random weights (only at beginning)

$[[0.092, 0.210, 0.415],$
 $[0.778, 0.929, 0.030], \rightarrow$
 $[0.019, 0.182, 0.555],$

...,



4. Repeat with more examples

$\rightarrow [116, 78, 15],$
 $\rightarrow [117, 43, 96], \rightarrow$
 $[125, 87, 23],$

...,

$[[0.983, 0.004, 0.013],$
 $\rightarrow [0.110, 0.889, 0.001], \rightarrow$
 $[0.023, 0.027, 0.985],$

...,

Ramen,
Spaghetti

3. Update representation outputs

Inputs

Numerical encoding

Learns representation patterns/features/weights)

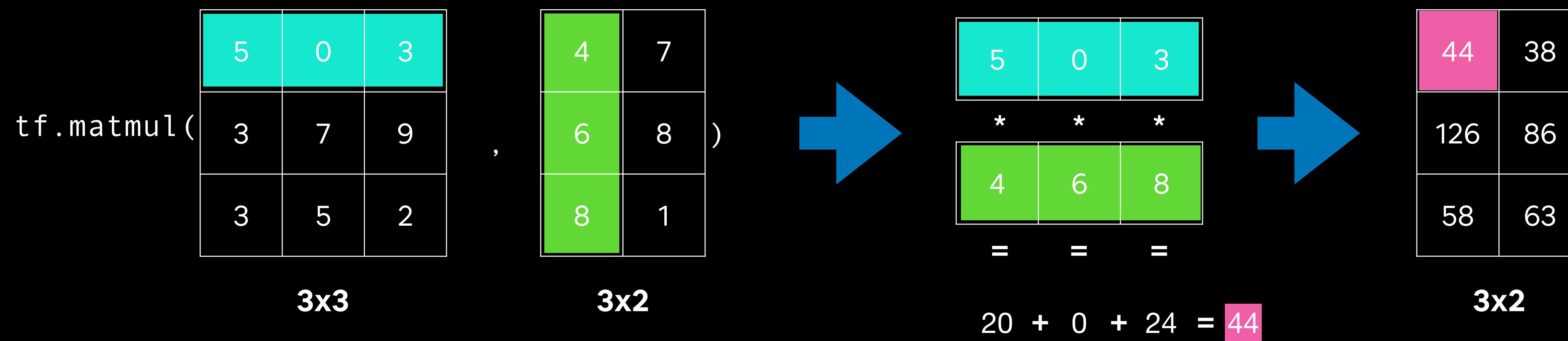
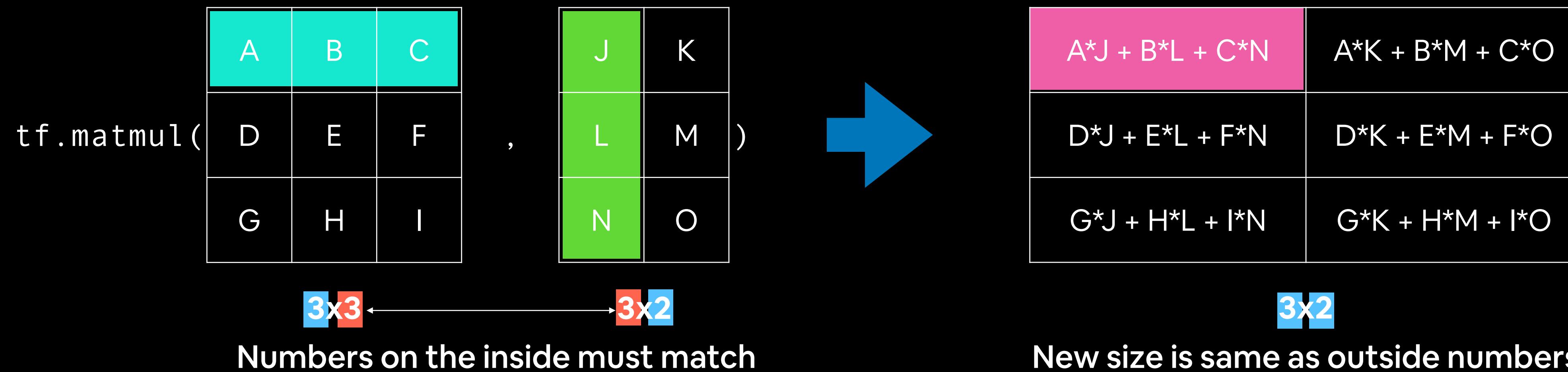
Representation outputs

Outputs

Tensor attributes

Attribute	Meaning	Code
Shape	The length (number of elements) of each of the dimensions of a tensor.	<code>tensor.shape</code>
Rank	The number of tensor dimensions. A scalar has rank 0, a vector has rank 1, a matrix is rank 2, a tensor has rank n.	<code>tensor.ndim</code>
Axis or dimension	A particular dimension of a tensor.	<code>tensor[0]</code> , <code>tensor[:, 1]...</code>
Size	The total number of items in the tensor.	<code>tf.size(tensor)</code>

Dot product



For a live demo, checkout www.matrixmultiplication.xyz