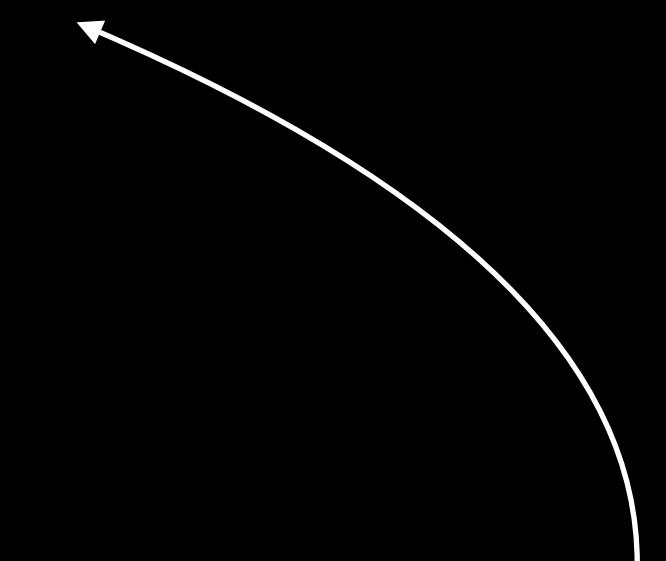


Deep Learning with  PyTorch

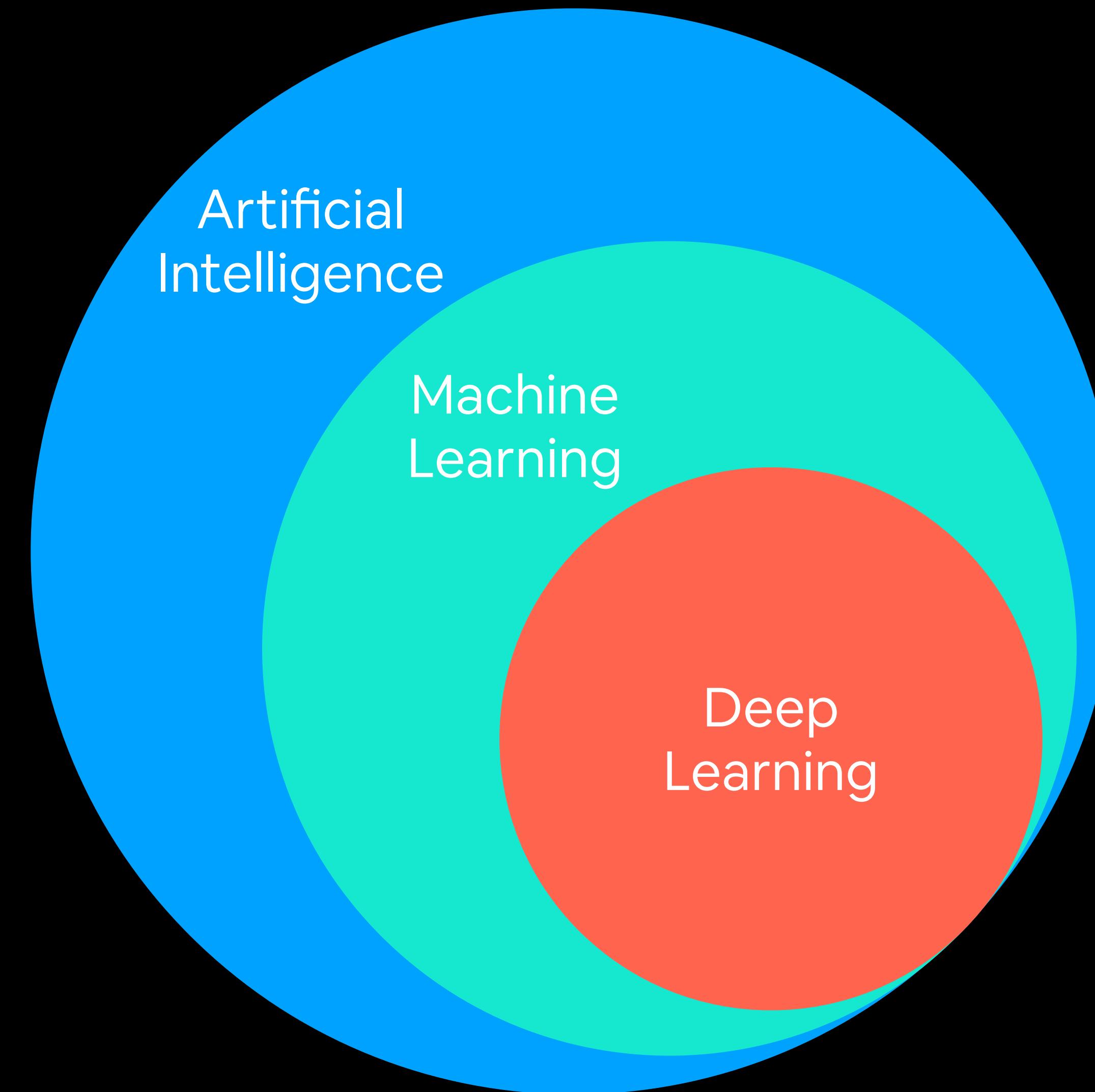
“What is deep learning?”

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



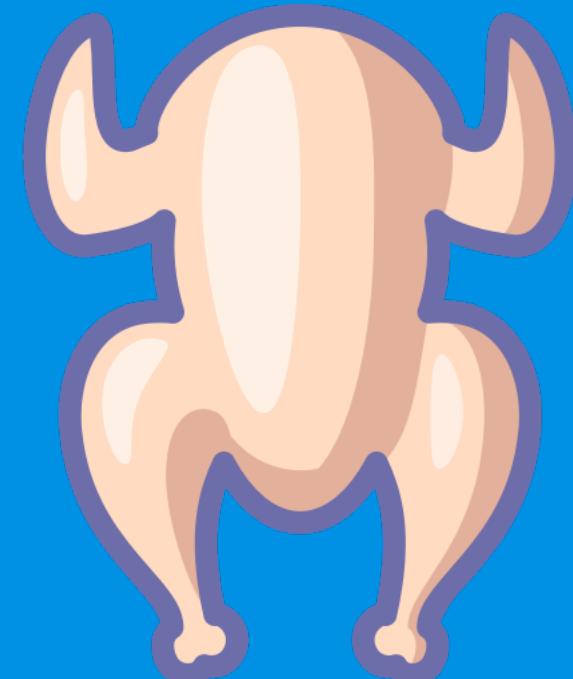
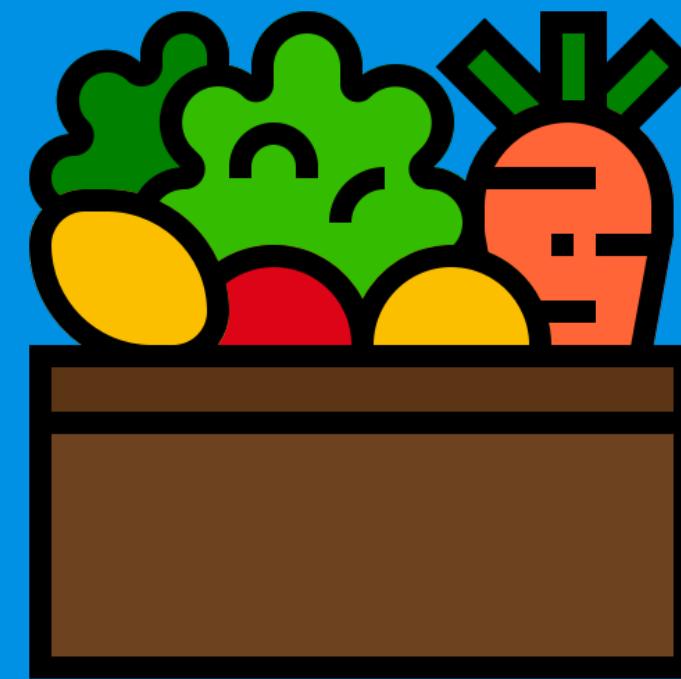
The computer does this part.  
How?  
Code & math.  
We're going to be writing the code.

# Machine Learning vs. Deep Learning



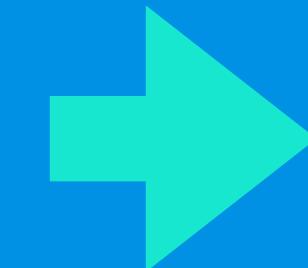
## Traditional programming

### Inputs

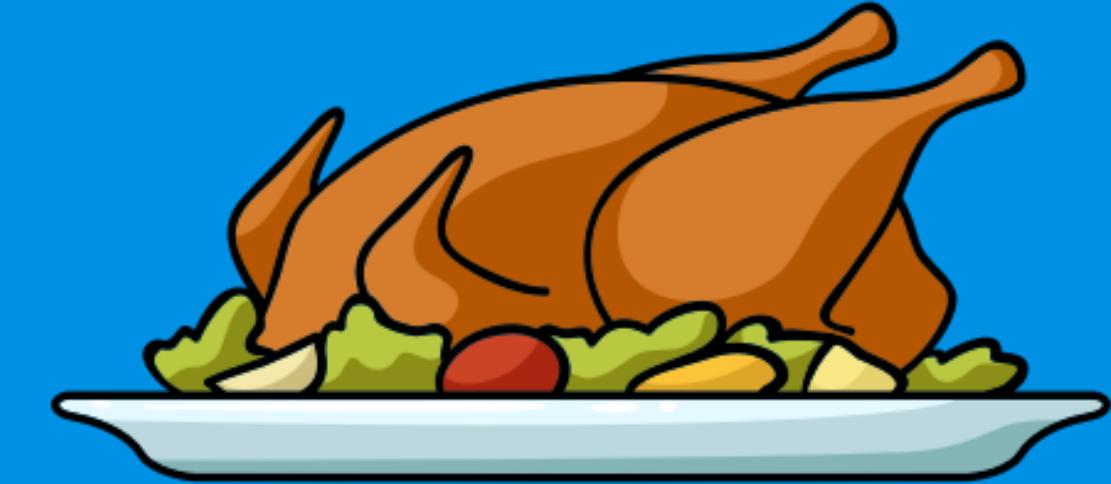


### Rules

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



### Output

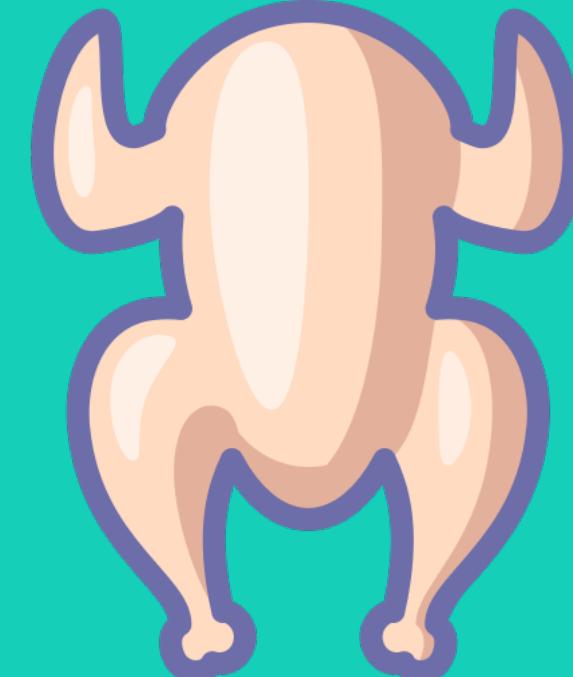
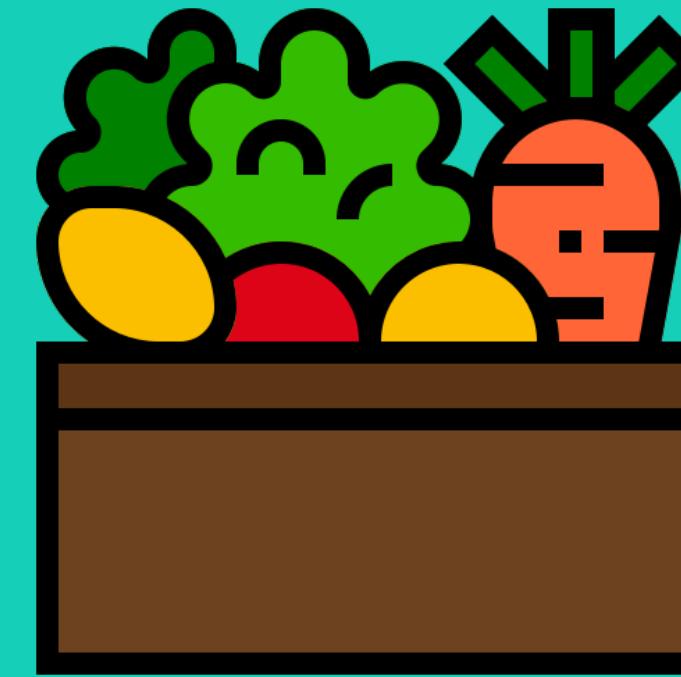


Starts with

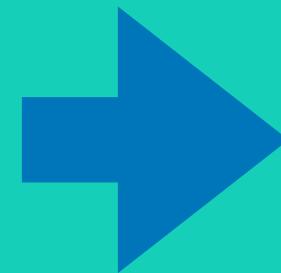
Makes

## 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?~~

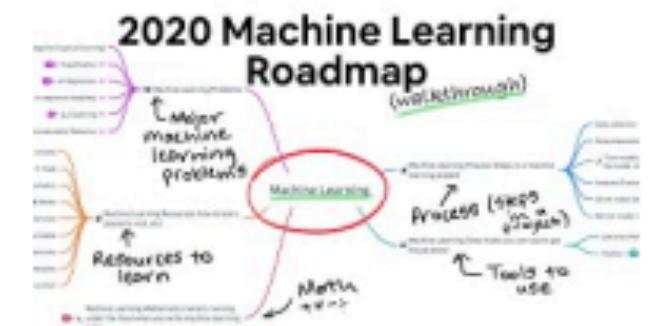
Better reason: For a complex problem, can you think of all the rules?

(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)

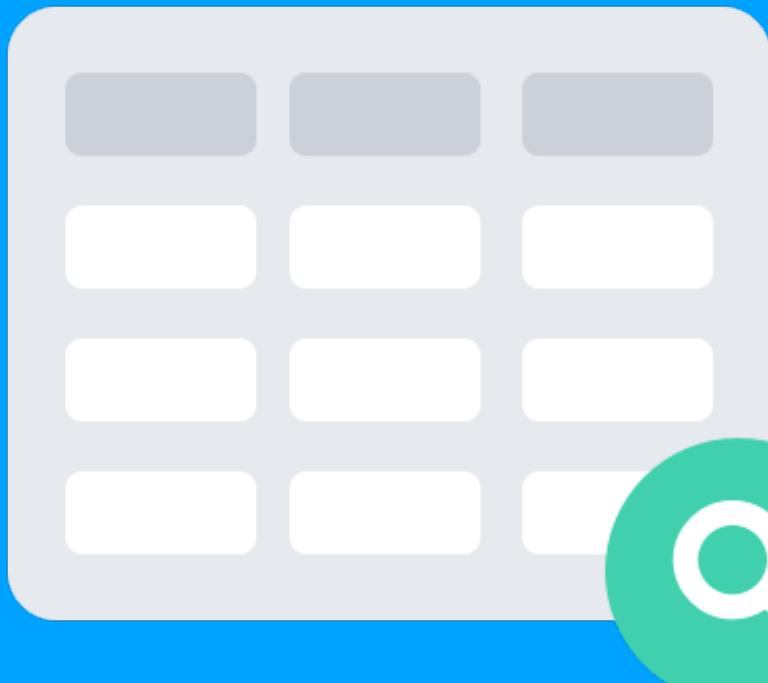
# Machine Learning vs. Deep Learning

## 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

Algorithm: gradient  
boosted machine

dmlc  
**XGBoost**



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.<sup>[1][2][3]</sup>

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.<sup>[4][5][6]</sup>

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:

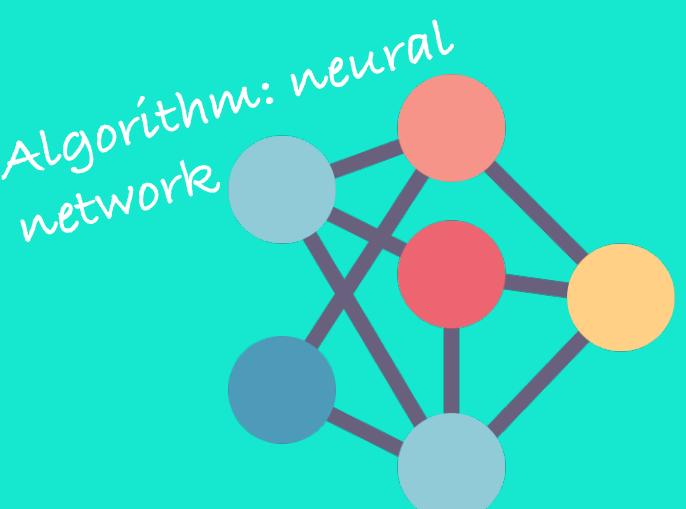
1. Learn Python
2. Learn Math/Stats/Probability
3. Learn software engineering
4. Build

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
- Gradient boosted models
- 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

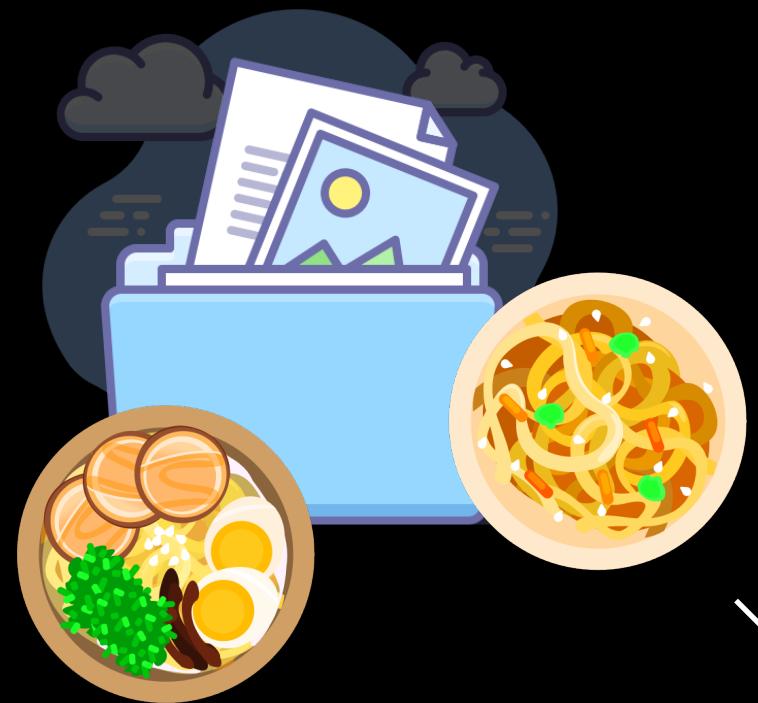
What we're focused on building  
(with PyTorch)

Structured data ← → Unstructured data

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

“What are neural networks?”

# Neural Networks



(before data gets used  
with a neural network,  
it needs to be turned  
into numbers)

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.



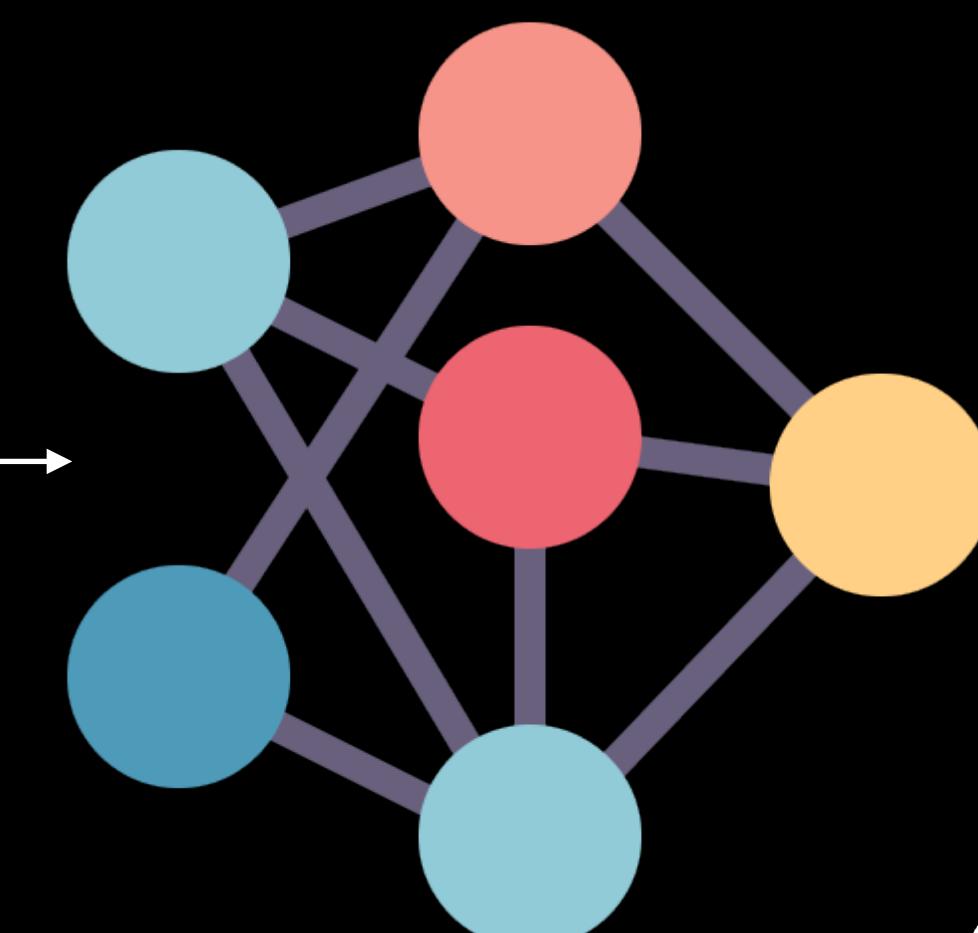
Inputs

Numerical  
encoding

Learns  
representation  
(patterns/features/weights)

Representation  
outputs

Outputs



(choose the appropriate  
neural network for your  
problem)

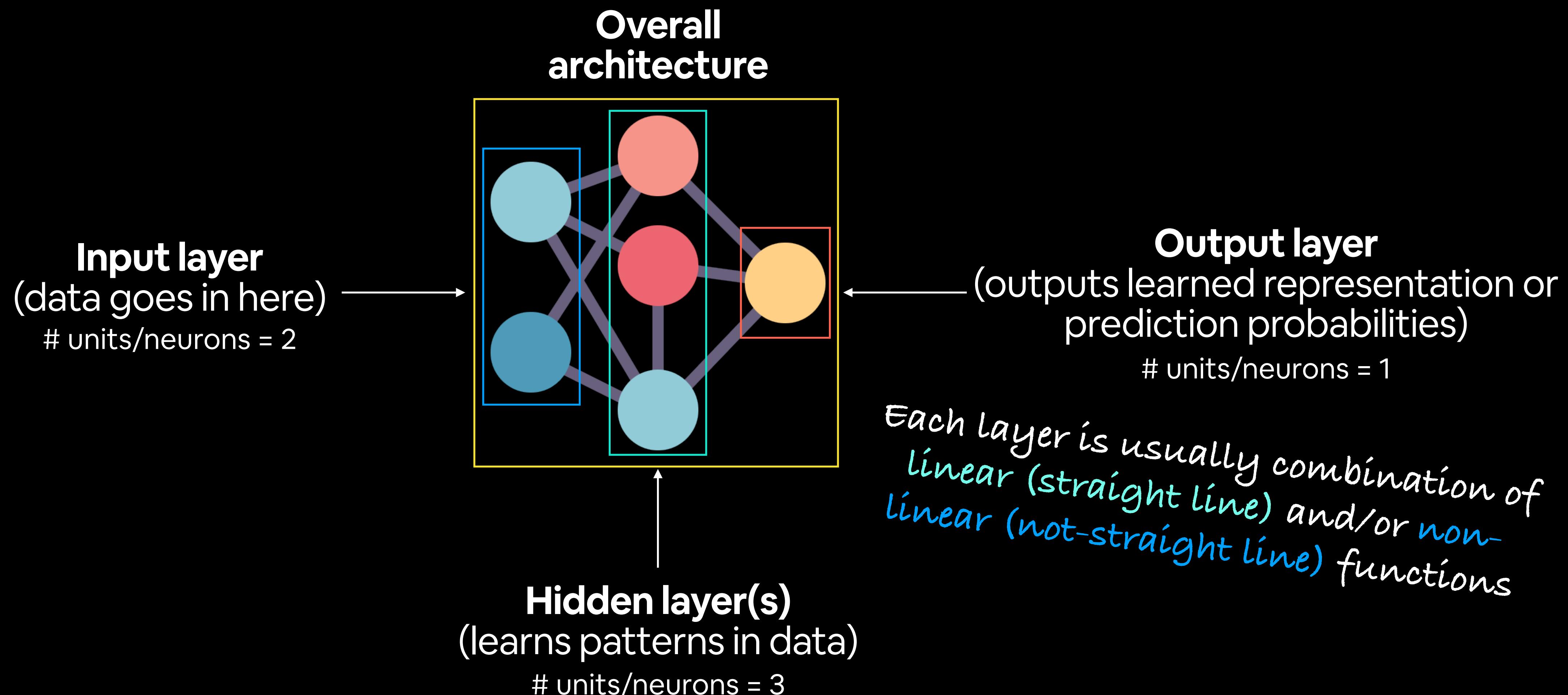
[ [116, 78, 15],  
→ [117, 43, 96], → [0.983, 0.004, 0.013],  
[125, 87, 23], ... , [0.110, 0.889, 0.001], → Not a disaster  
..., [0.023, 0.027, 0.985], ... , ]

(a human can  
understand these)

Ramen,  
Spaghetti

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

# 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.

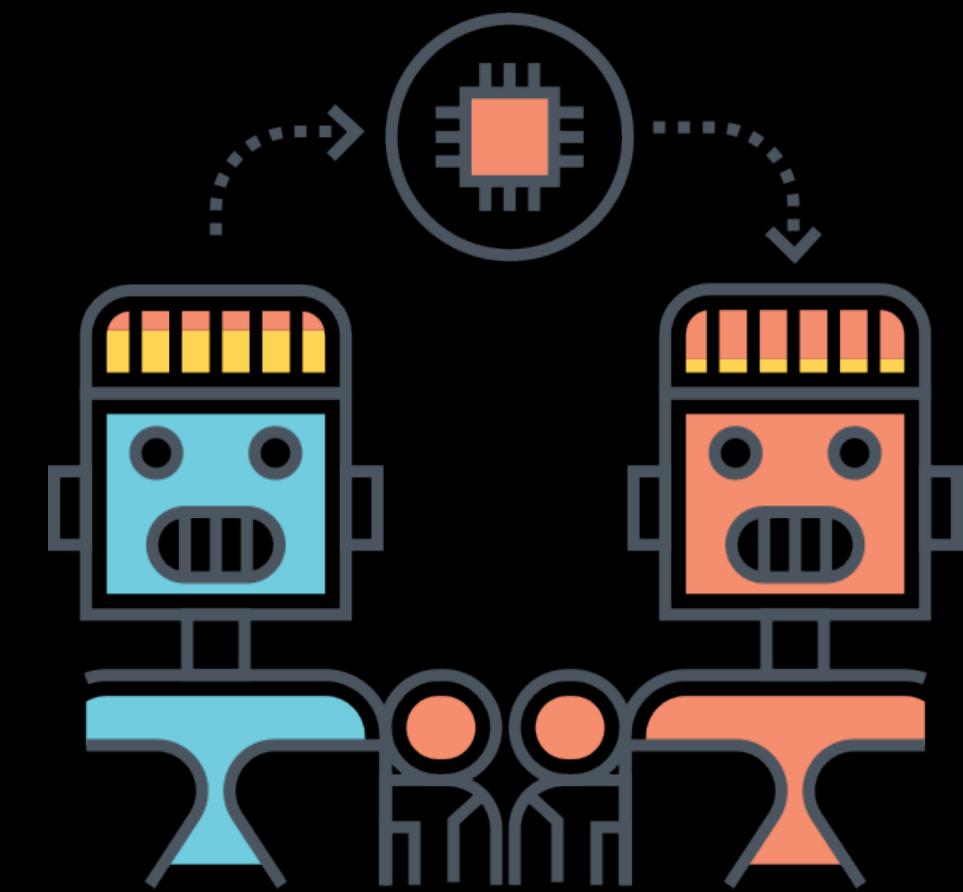
# Types of Learning



Supervised  
Learning



Unsupervised &  
Self-supervised  
Learning



Transfer  
Learning

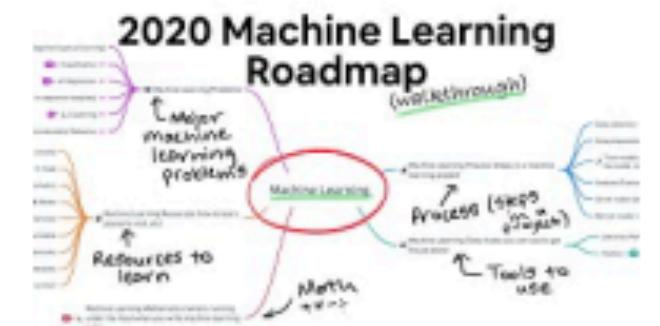
We'll be writing code to do these,  
but the style of code can be adopted across learning paradigms.

“What is deep learning actually  
used for?”



# YouTube

Yashaswi Kulshreshtha commented on your video



## 2020 Machine Learning Roadmap

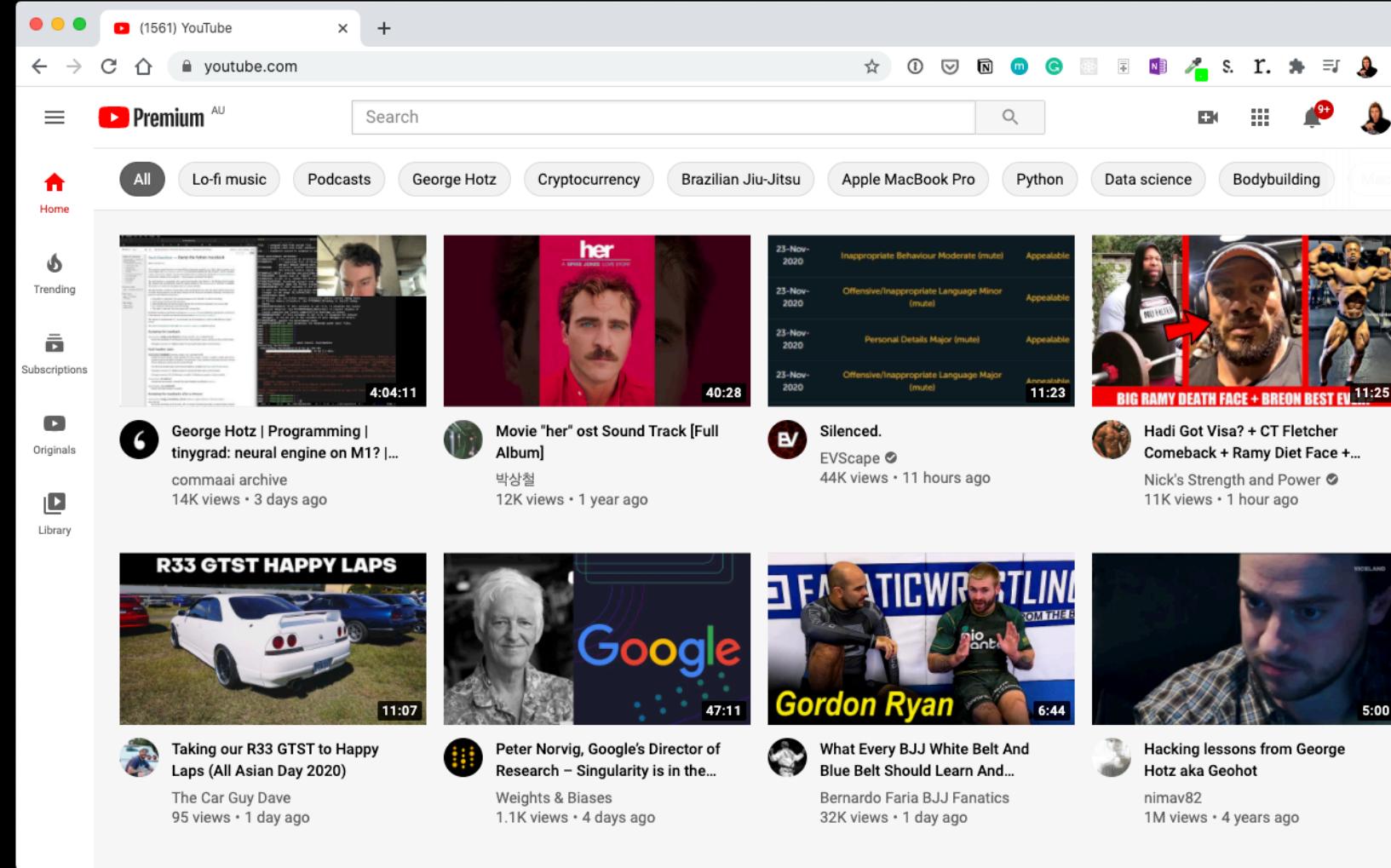


**Yashaswi Kulshreshtha**

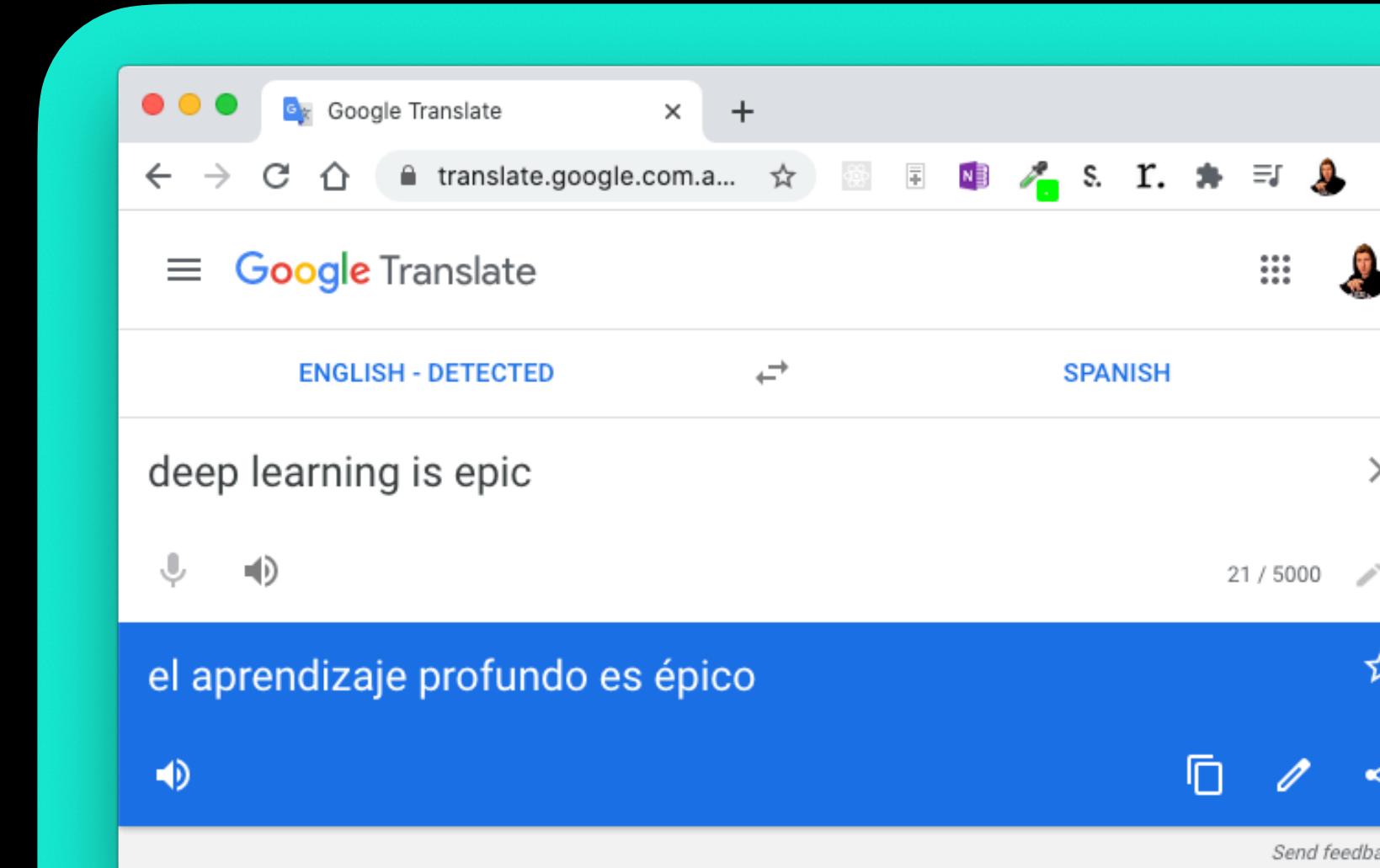
I think you can use ML for literally anything as long as you can convert it into numbers and program it to find patterns. Literally it could be anything any input or output from the universe

**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](mailto: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](mailto:daniel@mrdourke.com)  
Hay daniel...

Congratulations! U win \$1139239230

Spam

## Natural Language Processing (NLP)

## Sequence to sequence (seq2seq)

## Classification/regression



“What is PyTorch?”

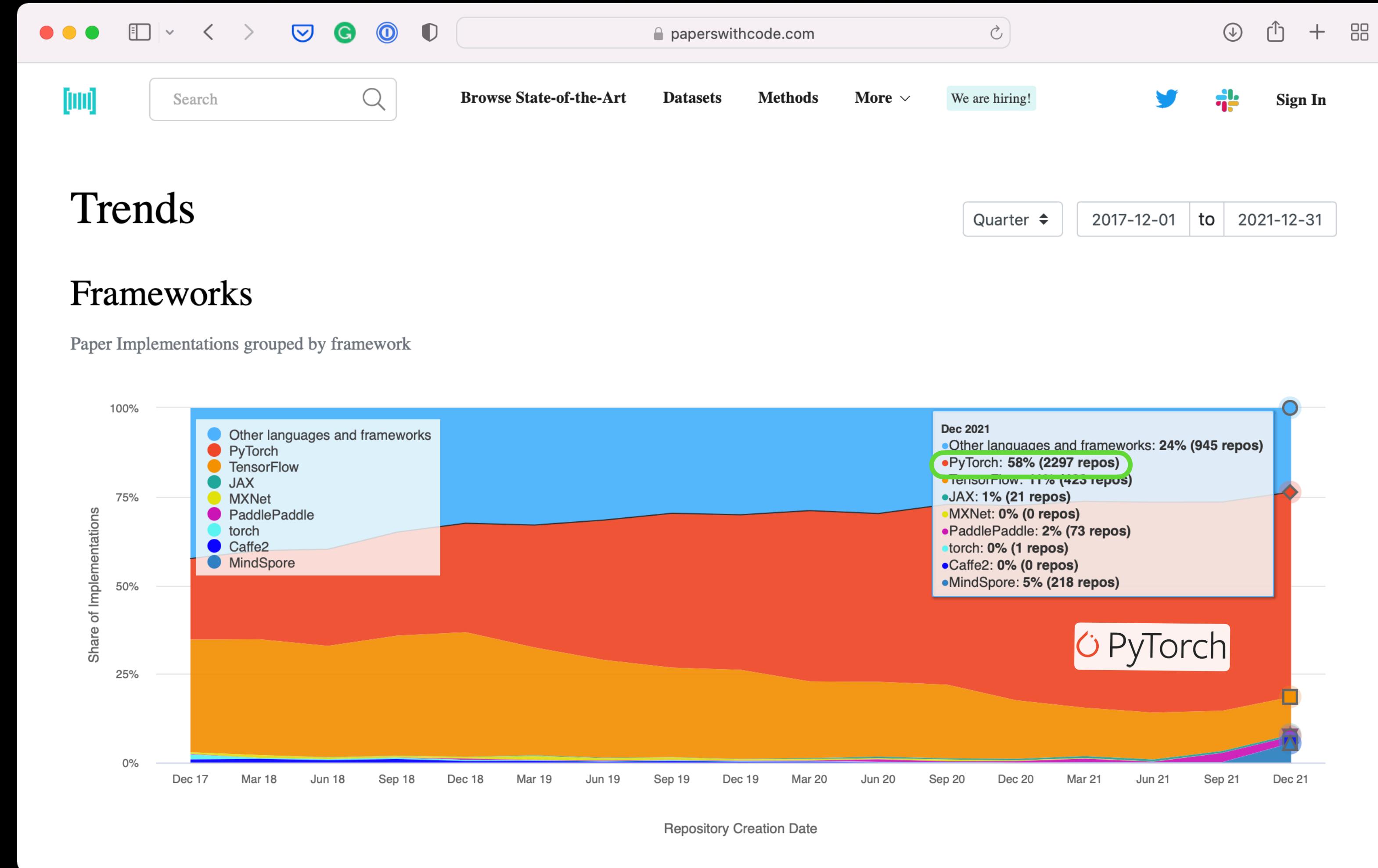


# What is PyTorch?



- Most popular research deep learning framework\*
- Write fast deep learning code in Python (able to run on a GPU/many GPUs)
- Able to access many pre-built deep learning models (Torch Hub/[`torchvision.models`](#))
- Whole stack: preprocess data, model data, deploy model in your application/cloud
- Originally designed and used in-house by Facebook/Meta (now open-source and used by companies such as Tesla, Microsoft, OpenAI)

# Why PyTorch?



Research favourite

# Why PyTorch?



A screenshot of a Twitter post from Francois 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 and PyTorch". The timestamp at the bottom left is "7:03 AM · Nov 21, 2020 · Twitter for Android".

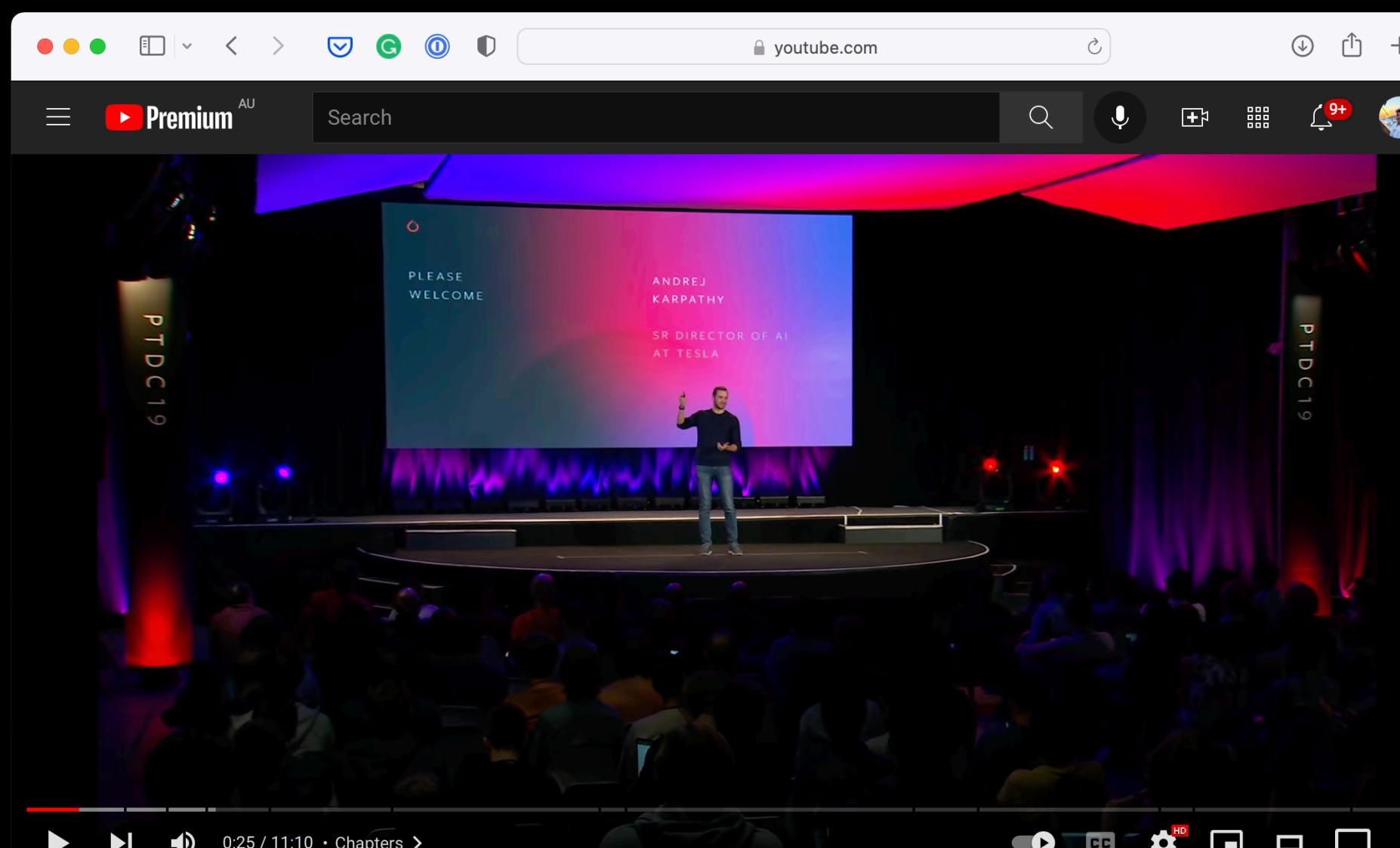
François Chollet   
@fchollet

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 and PyTorch

7:03 AM · Nov 21, 2020 · Twitter for Android

**Source:** [@fchollet Twitter](#)

# Why PyTorch?



PyTorch at Tesla - Andrej Karpathy, Tesla  
407,684 views • Nov 7, 2019  
9.8K DISLIKE SHARE SAVE ...

OpenAI Standardizes on PyTorch

We are standardizing OpenAI's deep learning framework on PyTorch. In the past, we implemented projects in many frameworks depending on their relative strengths. We've now chosen to standardize to make it easier for our team to create and share optimized implementations of our models.

January 30, 2020  
1 minute read

OpenAI PyTorch

A screenshot of a GitHub repository page for PyTorch. The repository has 8400+ stars and 1700+ forks. It is described as "The Incredible PYTORCH".

PyTorch  
Aug 7, 2020 • 11 min read • Listen

## AI for AG: Production machine learning for agriculture

Author: [Chris Padwick](#), Director of Computer Vision and Machine Learning at Blue River Technology

How did farming affect your day today? If you live in a city, you might feel d. Agriculture is a

RESEARCH

## PyTorch builds the future of AI and machine learning at Facebook

June 2, 2021

Facebook's AI models perform trillions of inference operations per day for the billions of people that use our technologies. The increasing workload demand means we have to continually improve our infrastructure. Which is why, today we're announcing that we're moving our core AI systems to PyTorch.

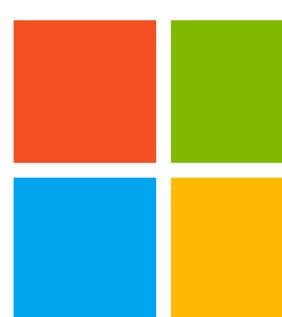
A screenshot of the Facebook MetaAI website. It features a large image of a agricultural field and text about PyTorch building the future of AI and machine learning at Facebook.

MetaAI

Research Publications People Events Tools Datasets Blog Join Us

Share on Facebook Share on Twitter

Our Work

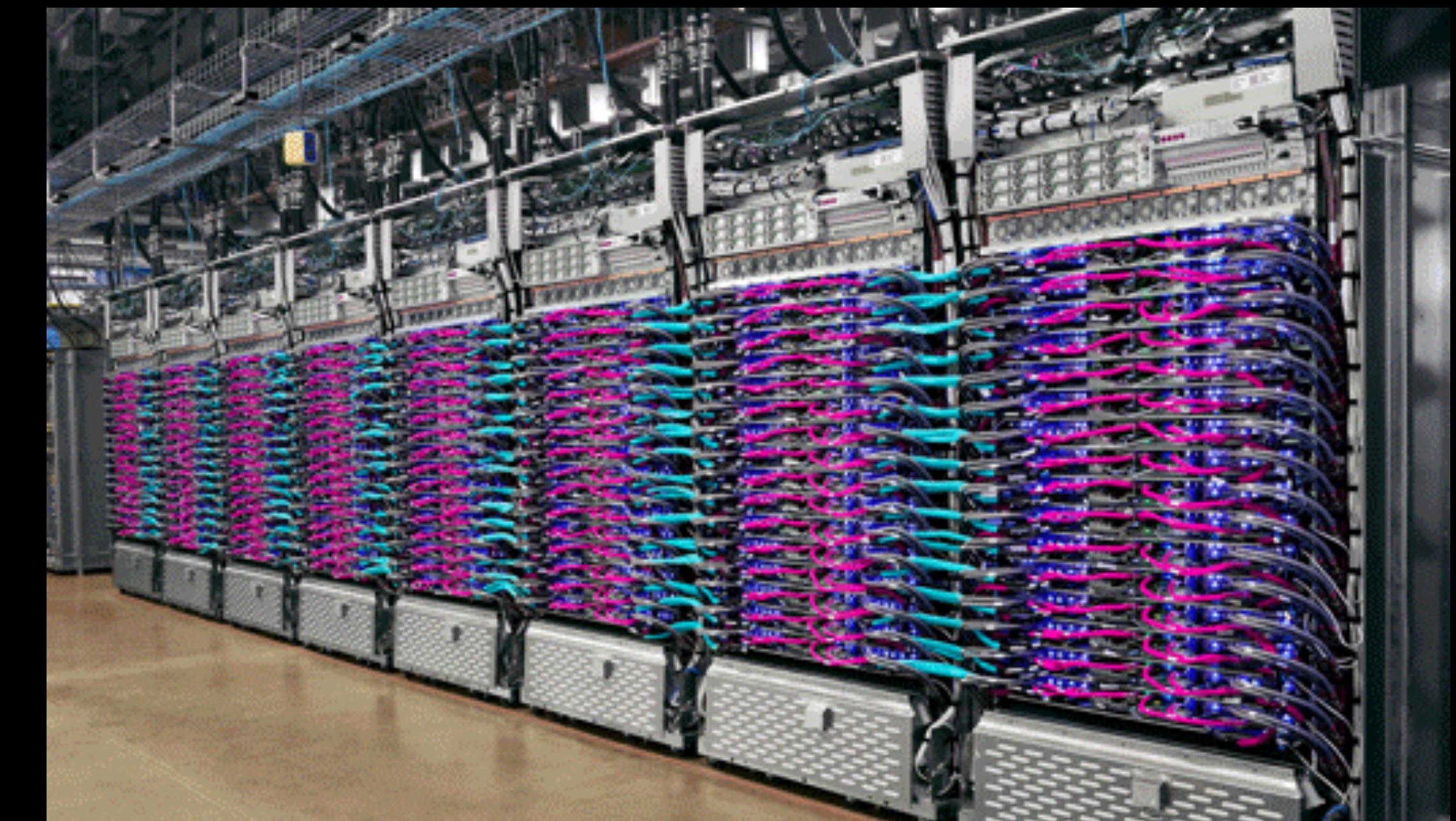


# Microsoft

# What is a GPU/TPU?



GPU (Graphics Processing Unit)

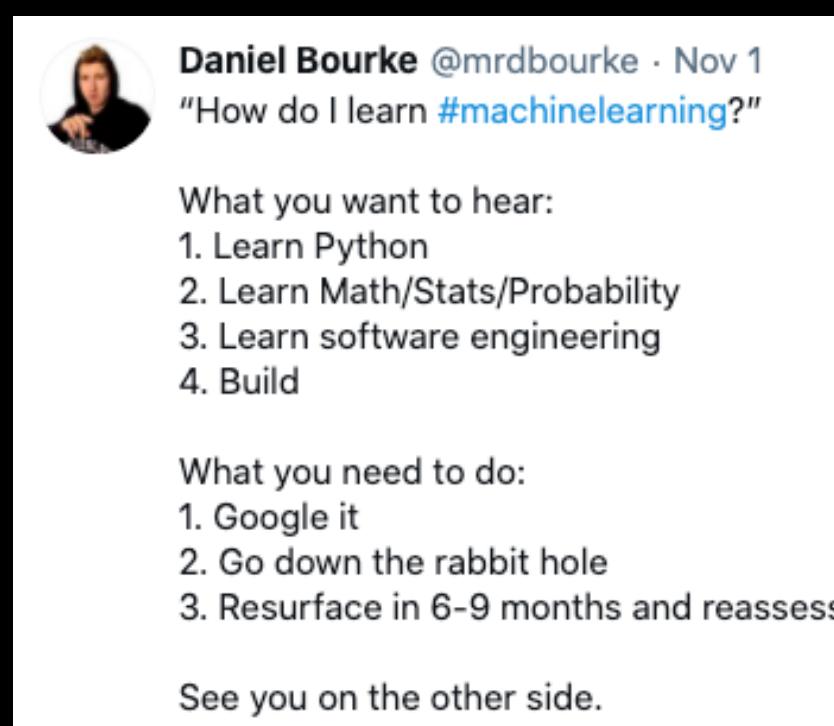


TPU (Tensor Processing Unit)

“What is a tensor?”

# Neural Networks

**These are tensors!**

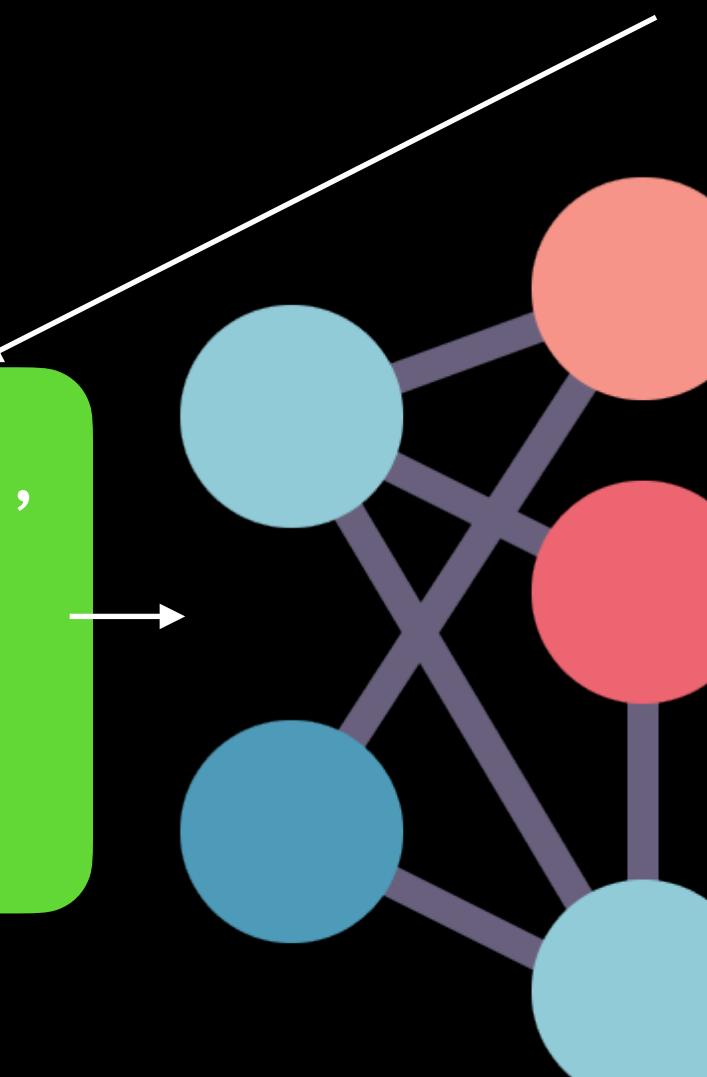


Inputs

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],  
 ... ,
```



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],  
 ... ,
```

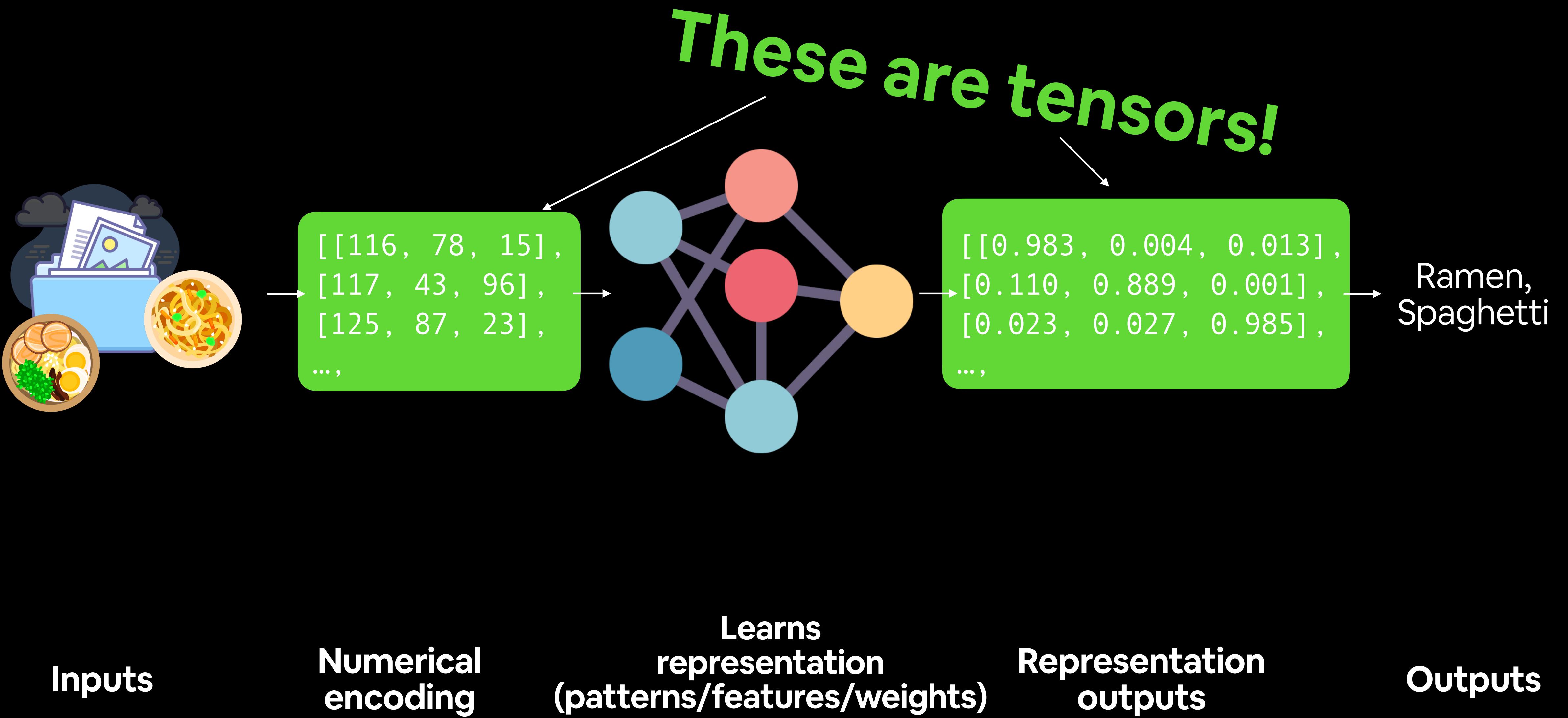
(a human can understand these)

Ramen, Spaghetti

Not spam

"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**

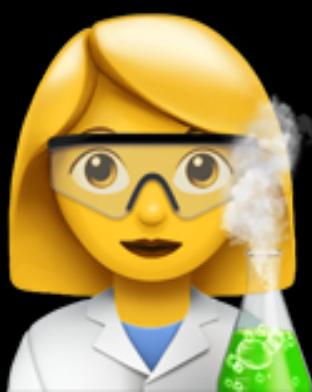
# What we're going to cover

(broadly)

- Now:
  - PyTorch basics & fundamentals (dealing with tensors and tensor operations)
- Later:
  - 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!)

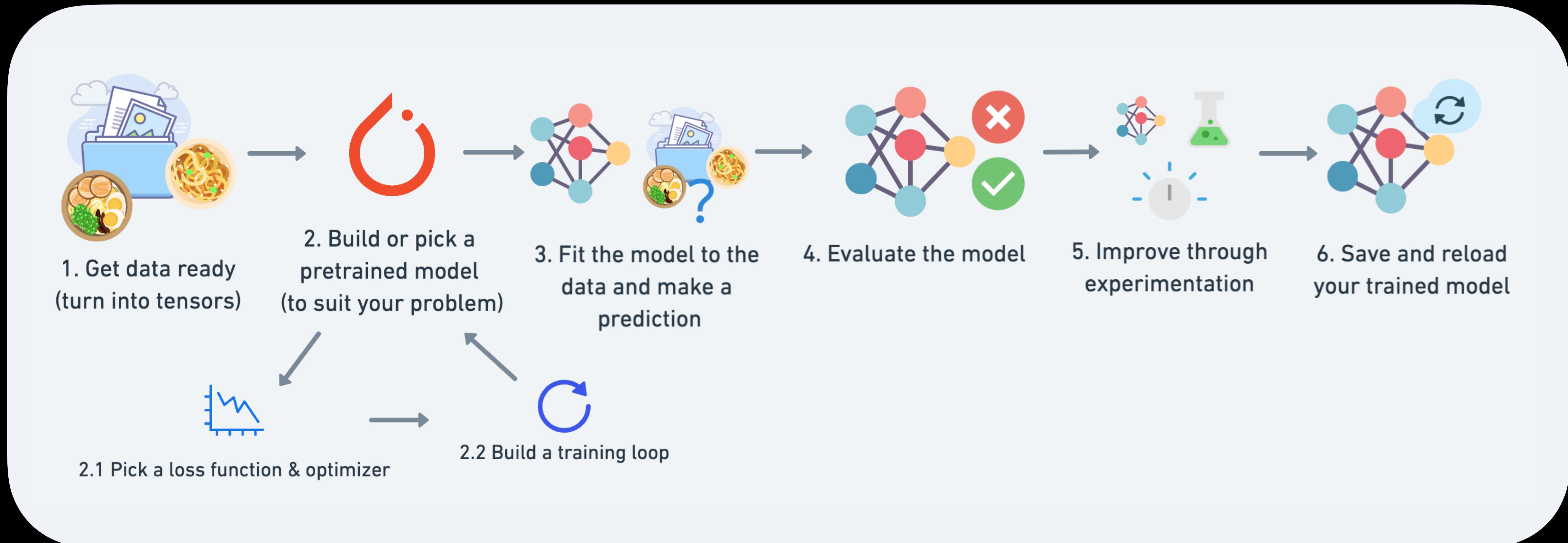
**How:**



# What we're going to cover

## A PyTorch workflow

(one of many)



“How should I approach  
this course?”

# How to approach this course

```
1 # 1. Construct a model class that subclasses nn.Module
2 class CircleModelV0(nn.Module):
3     def __init__(self):
4         super().__init__()
5         # 2. Create 2 nn.Linear layers
6         self.layer_1 = nn.Linear(in_features=2, out_features=5)
7         self.layer_2 = nn.Linear(in_features=5, out_features=1)
8
9     # 3. Define a forward method containing the forward pass computation
10    def forward(self, x):
11        # Pass the data through both layers
12        return self.layer_2(self.layer_1(x))
13
14 # 4. Create an instance of the model and send it to target device
15 model_0 = CircleModelV0().to(device)
16 model_0
```

## 1. Code along

Motto #1: if in doubt, run the code!



(including the  
“dumb” ones)

## 4. Ask questions

## 5. Do the exercises

## 6. Share your work



Motto #3:  
visualize, visualize, visualize!



## 3. Visualize what you don't understand



# How **not** to approach this course

Avoid:



“**I can’t learn**  
”

---

# Resources

This course

## Course materials

A screenshot of a GitHub repository page for 'mrdbourke/pytorch-deep-learning'. The repository has 28 issues, 11 forks, and 76 stars. The 'Code' tab is selected, showing a list of files and commits. Notable commits include 'update exercises', 'Update make\_docs.yml', and 'add exercises and solutions for 01'. The repository is described as 'Materials for upcoming beginner-friendly PyTorch course (work in progress)'.

<https://www.github.com/mrdbourke/pytorch-deep-learning>

## Course Q&A

A screenshot of the GitHub Discussions page for the same repository. It features a purple header with the text 'Welcome to pytorch-deep-learning Discussions!'. Below the header, there's a search bar and a 'New discussion' button. Categories listed include 'Announcements', 'General', 'Ideas', and 'Q&A'. A message from 'mrdbourke' welcomes users to the discussions.

<https://www.github.com/mrdbourke/pytorch-deep-learning/discussions>

## Course online book

A screenshot of the 'Zero to Mastery Learn PyTorch for Deep Learning' website. It features a dark theme with white text. The main content area says 'Welcome to the Learn PyTorch for Deep Learning book (work in progress)'. It explains that this will be the homepage for the online book version of the course. The course teaches foundations of deep learning and PyTorch. A sidebar lists '00. PyTorch Fundamentals' and other sections. At the bottom, it says 'Made with Material for MkDocs'.

<https://learnpytorch.io>

## PyTorch website & forums

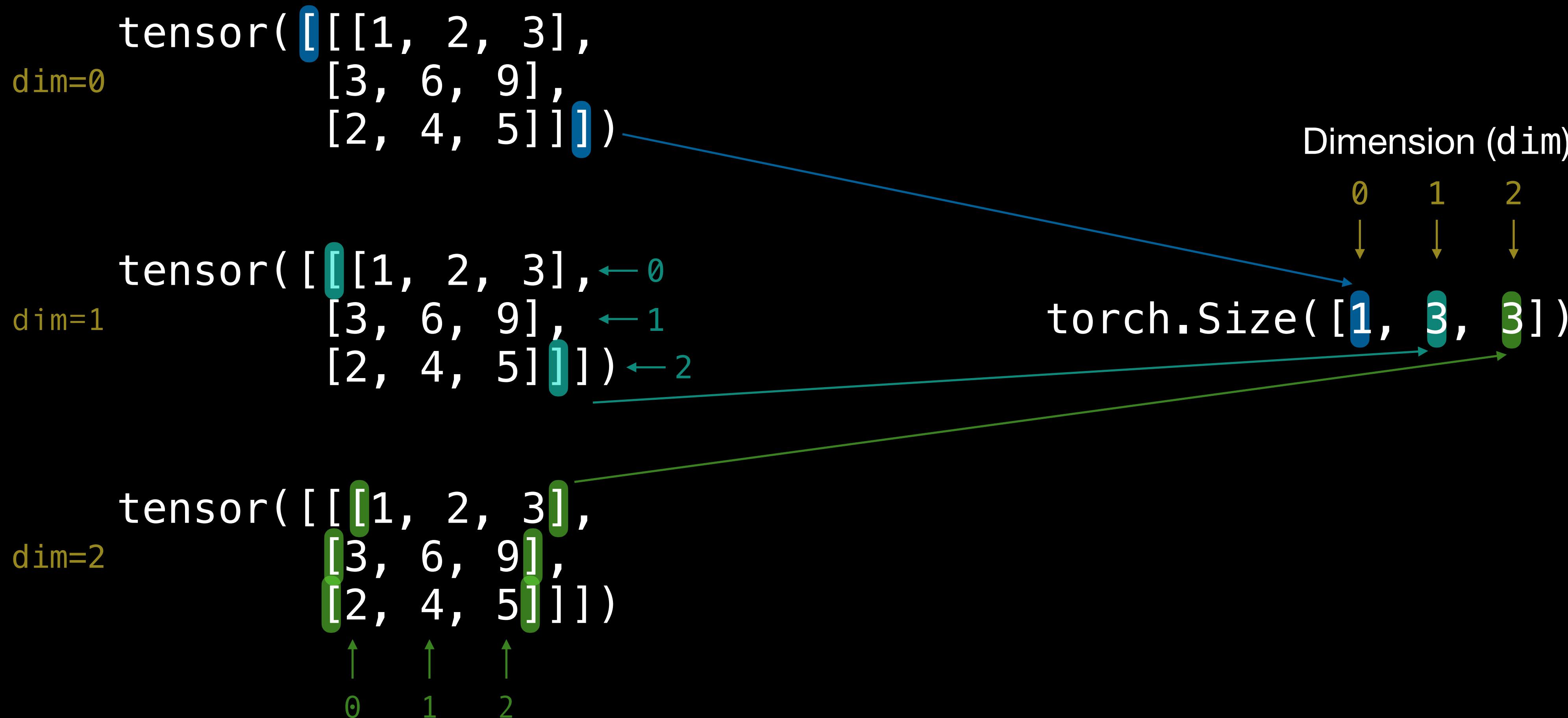
A screenshot of the official PyTorch website. The background is a colorful abstract design. The text 'FROM RESEARCH TO PRODUCTION' is prominently displayed. Below it, it says 'An open source machine learning framework that accelerates the path from research prototyping to production deployment.' There's a 'Install >' button and a note about the 'PyTorch 1.10 Release, including CUDA Graphs APIs, TorchScript improvements'.

All things PyTorch

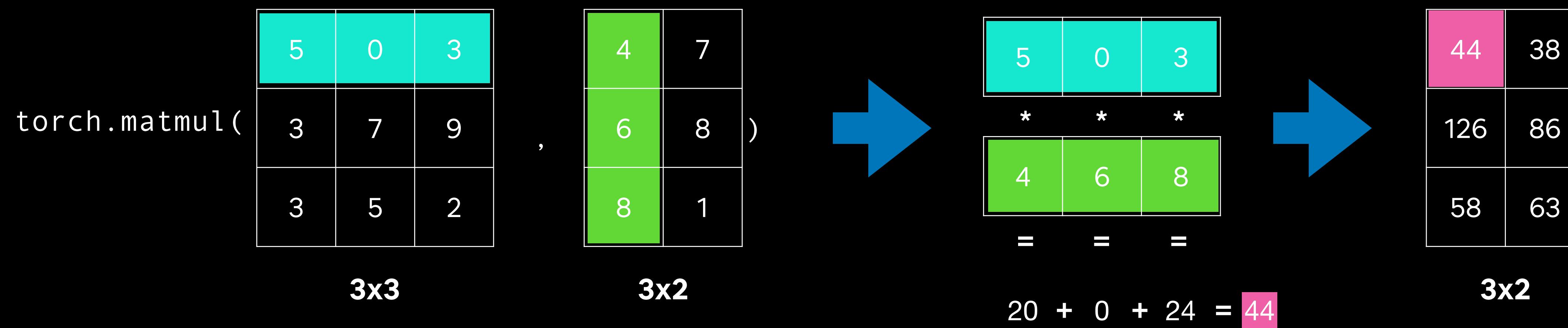
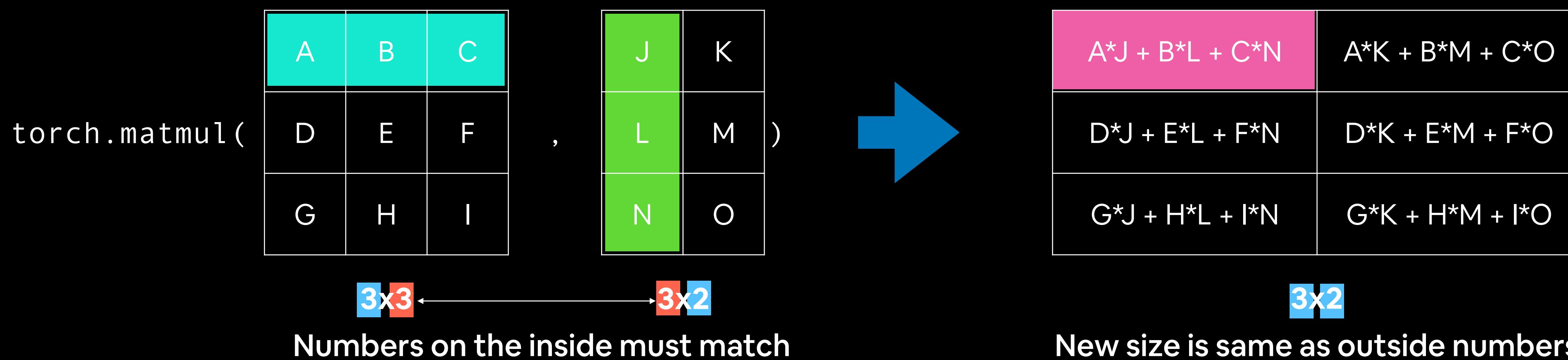
A screenshot of the PyTorch forums at 'discuss.pytorch.org'. The top navigation includes 'Sign Up', 'Log In', and a search bar. The main content shows a list of topics under categories like 'vision', 'nlp', 'autograd', etc. Each topic has a timestamp and a brief description. For example, 'Model train, val, test workflow verification in PyTorch' was posted 24m ago.

Let's code!

# Tensor dimensions

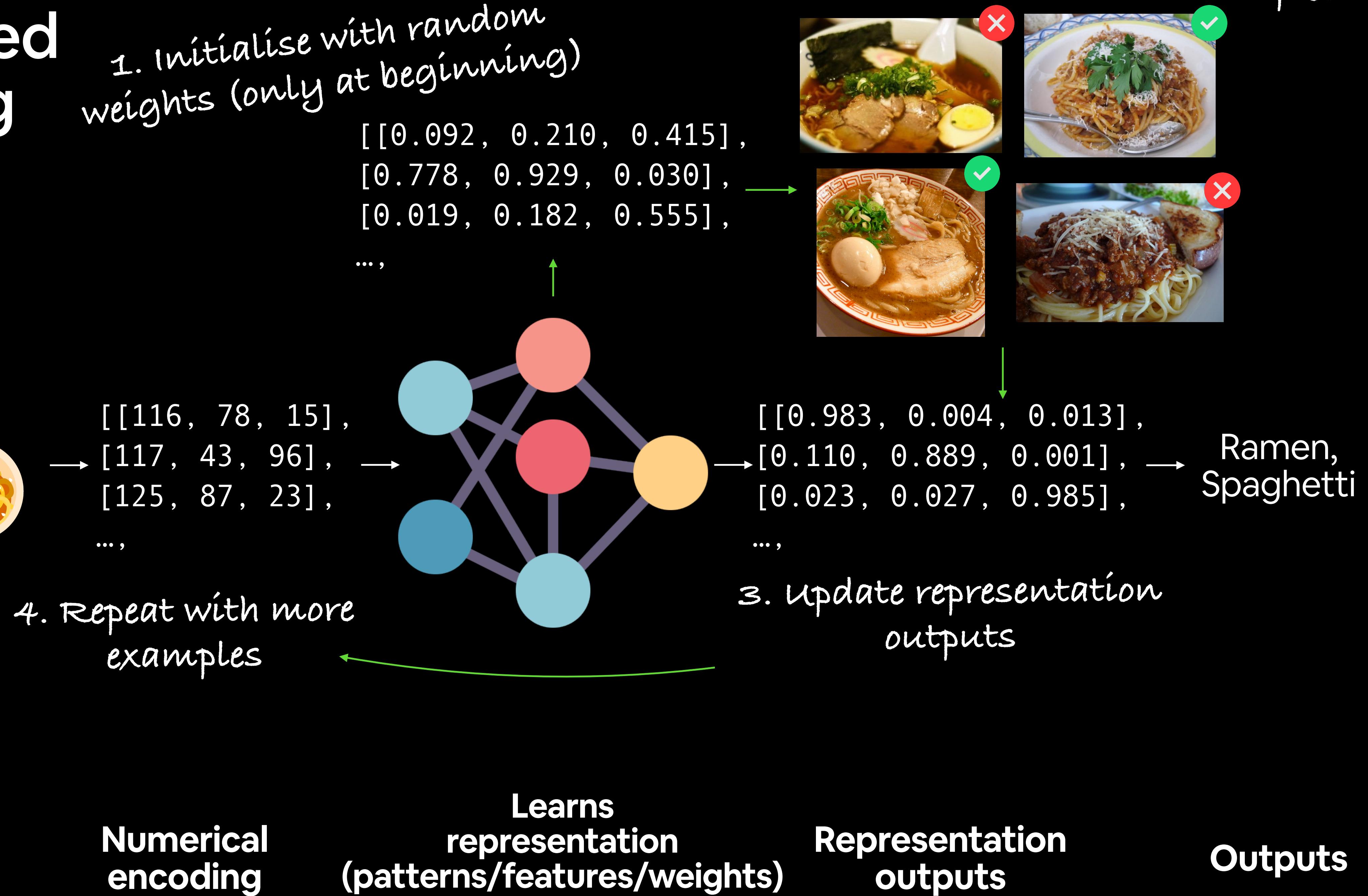
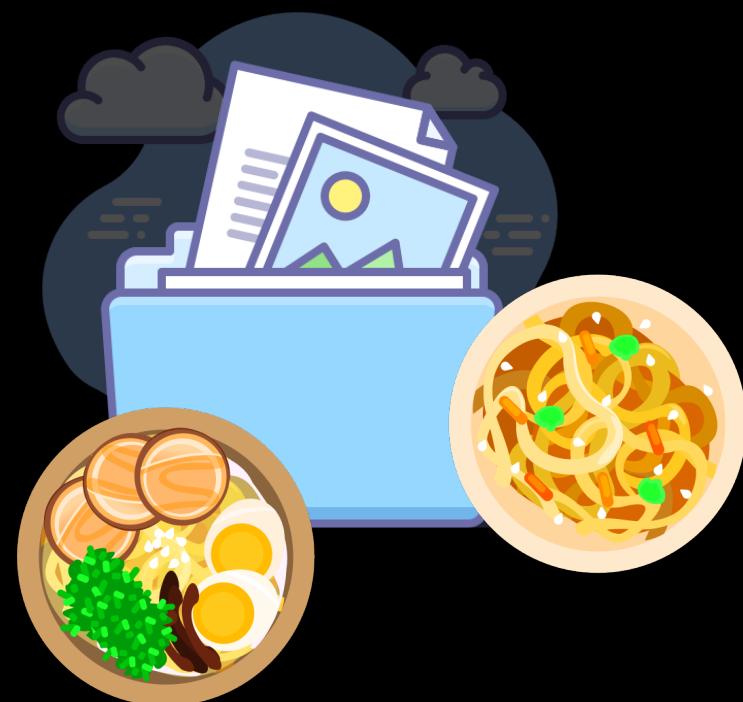


# Dot product



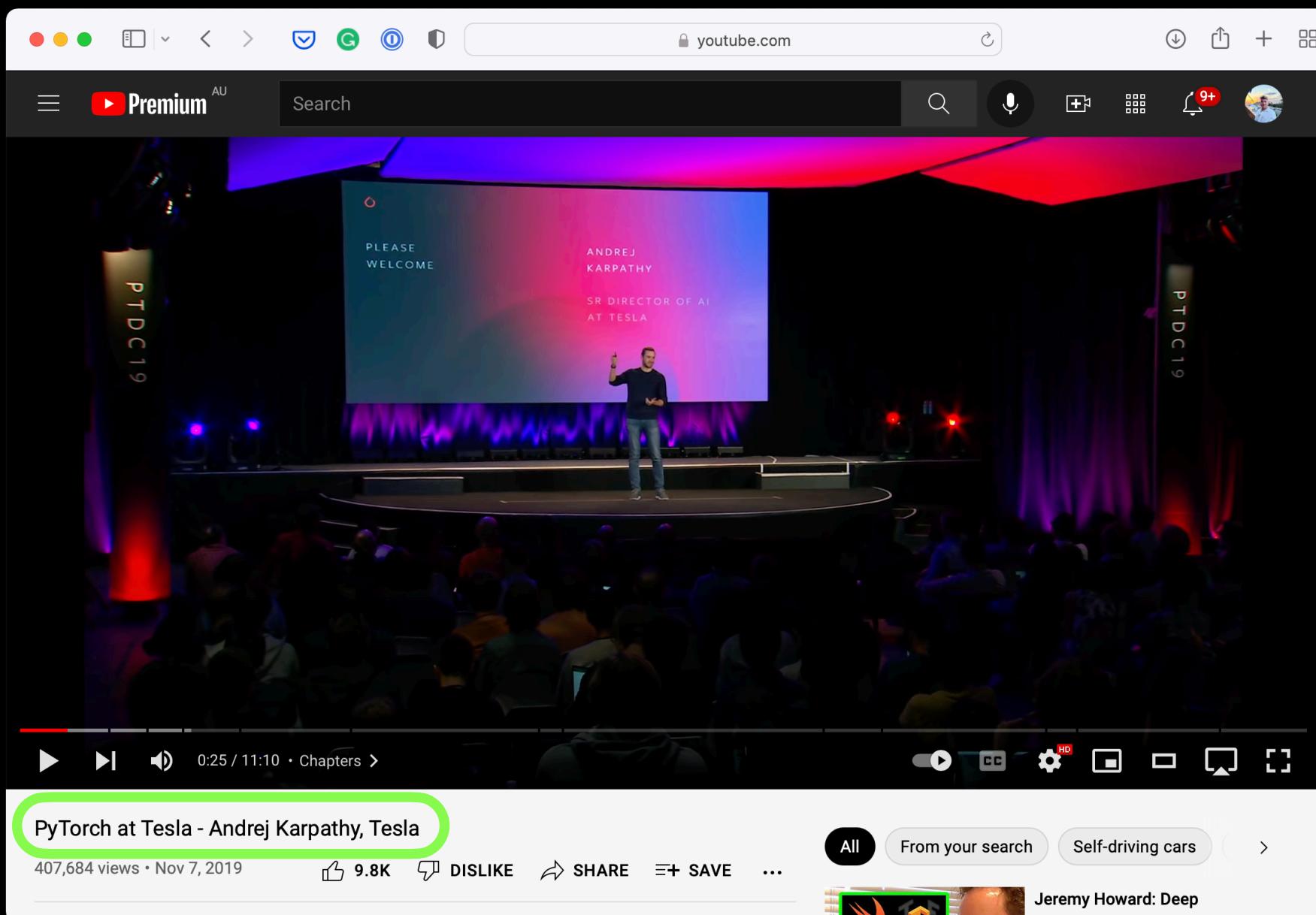
For a live demo, checkout [www.matrixmultiplication.xyz](http://www.matrixmultiplication.xyz)

# Supervised learning (overview)



# Tensor attributes

Attribute	Meaning	Code
Shape	The length (number of elements) of each of the dimensions of a tensor.	<code>tensor.shape</code>
Rank/dimensions	The total 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> or <code>tensor.size()</code>
Specific axis or dimension (e.g. “1st axis” or “0th dimension”)	A particular dimension of a tensor.	<code>tensor[0]</code> , <code>tensor[:, 1]...</code>



## OpenAI Standardizes on PyTorch

We are standardizing OpenAI's deep learning framework on PyTorch. In the past, we implemented projects in many frameworks depending on their relative strengths. We've now chosen to standardize to make it easier for our team to create and share optimized implementations of our models.

January 30, 2020  
1 minute read

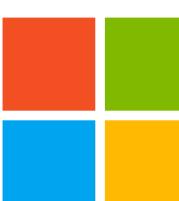
## AI for AG: Production machine learning for agriculture

Author: [Chris Padwick](#), Director of Computer Vision and Machine Learning at Blue River Technology

How did farming affect your day today? If you live in a city, you might feel disconnected from the farms and fields that produce your food. Agriculture is a core piece of our lives, but we often take it for granted.



A 2017 prototype of See & Spray, Blue River Technology's precision weed control machine

 Microsoft

PyTorch builds the future of AI and machine learning at Facebook

June 2, 2021

Facebook's AI models perform trillions of inference operations every day for the billions of people that use our technologies. Meeting this growing workload demand means we have to continually evolve our AI frameworks. Which is why, today we're announcing that we're migrating all our AI systems to PyTorch.

 Meta AI

RESEARCH

PyTorch builds the future of AI and machine learning at Facebook

June 2, 2021

Share on Facebook Share on Twitter

Our Work

Facebook AI's Joelle

