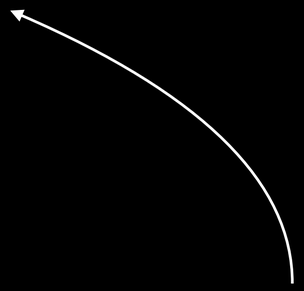
Deep Learning with 

“What is deep learning?”

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



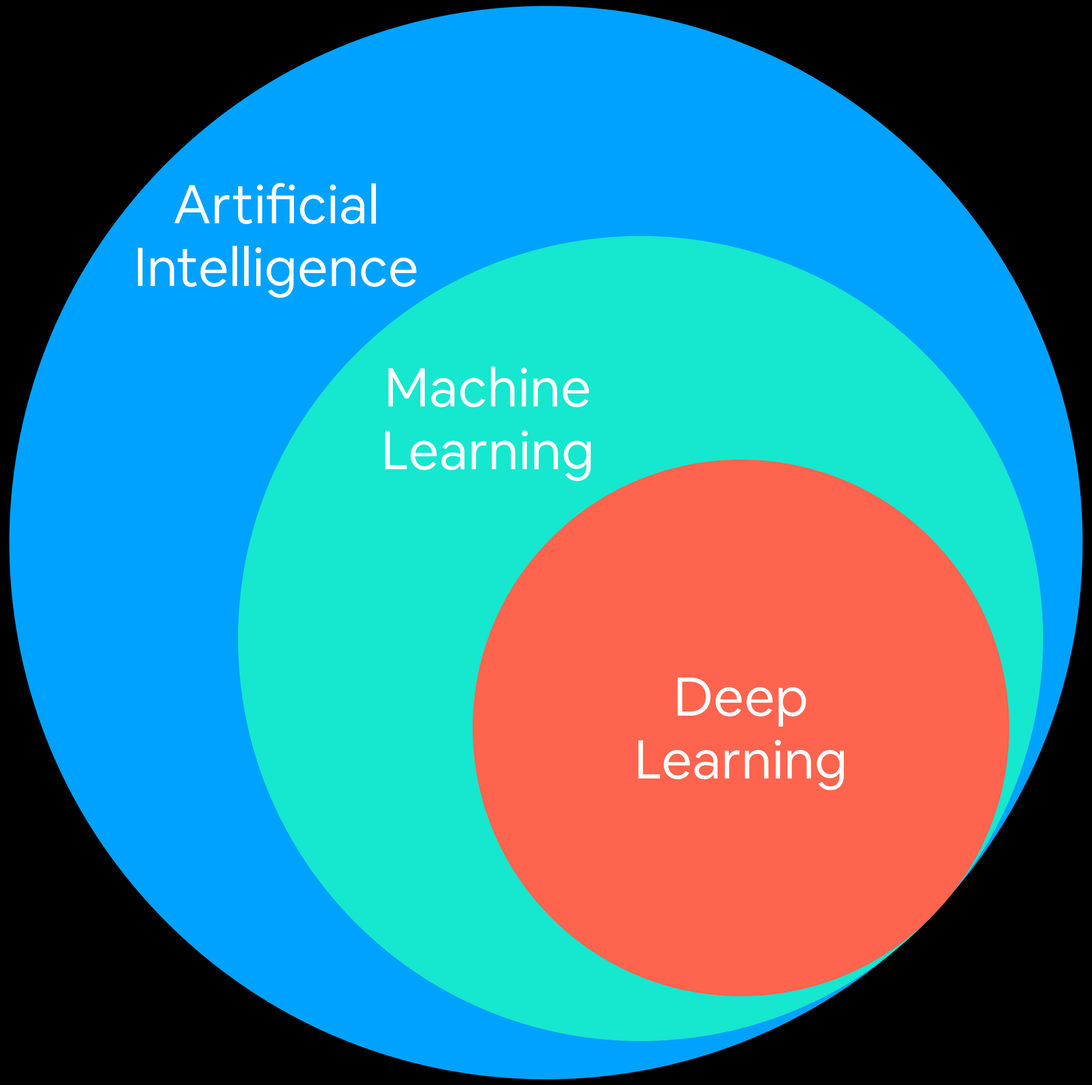
The computer does this part.

How?

Code & math.

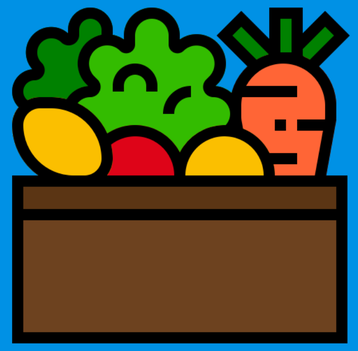
We're going to be writing the code.

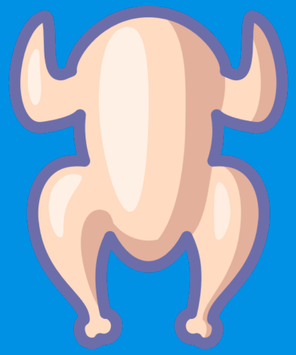
Machine Learning vs. Deep Learning



Traditional programming

Inputs

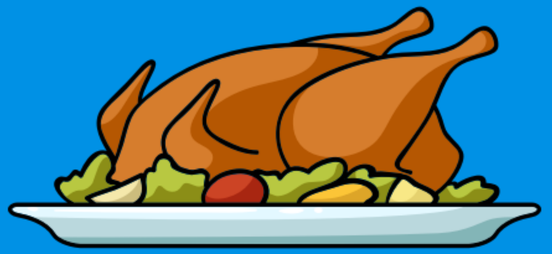




Rules

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

Output

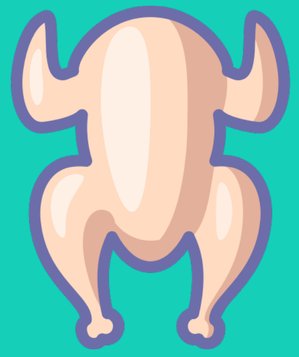


Starts with

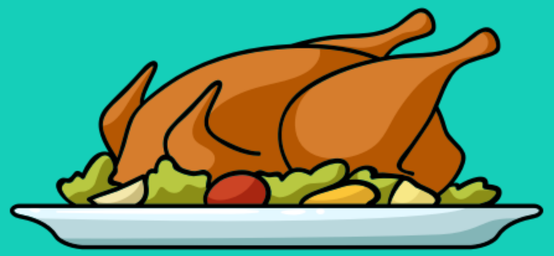
Makes

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)?”

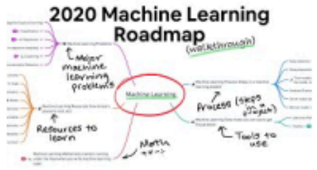
Good reason: ~~Why not?~~

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?

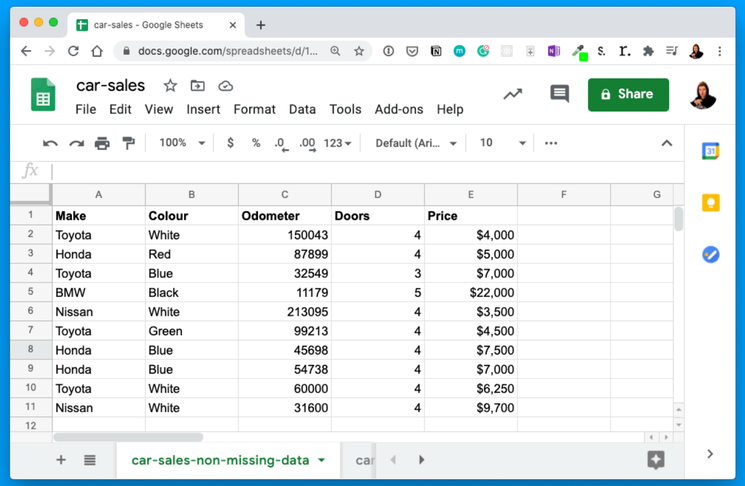
What deep learning is (typically) not good for 🤖🚫

- **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



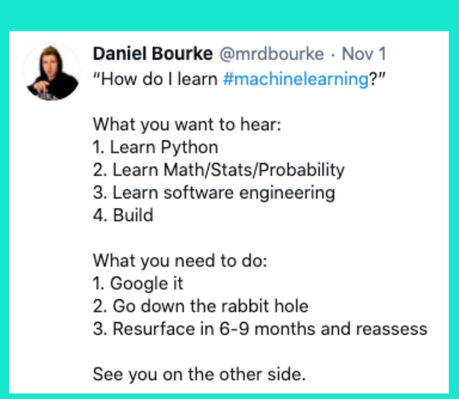
	A	B	C	D	E	F	G
1	Make	Colour	Odometer	Doors	Price		
2	Toyota	White	150043	4	$4,000		
3	Honda	Red	87899	4	$5,000		
4	Toyota	Blue	32549	3	$7,000		
5	BMW	Black	11179	5	$22,000		
6	Nissan	White	213095	4	$3,500		
7	Toyota	Green	99213	4	$4,500		
8	Honda	Blue	45698	4	$7,500		
9	Honda	Blue	54738	4	$7,000		
10	Toyota	White	60000	4	$6,250		
11	Nissan	White	31600	4	$9,700		
12							

Algorithm: gradient boosted machine

dmlc XGBoost

Structured data

Deep Learning



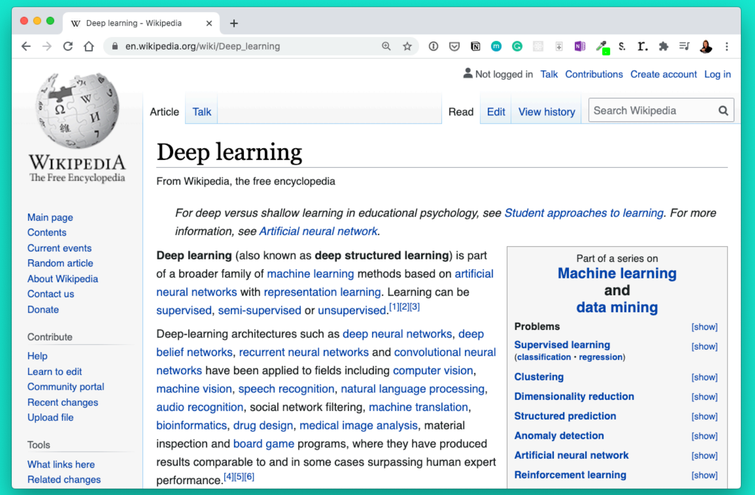
Daniel Bourke @mrdbourke · Nov 1

"How do I learn #machinelearning?"

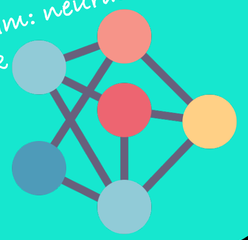
What you want to hear:
1. Learn Python
2. Learn Math/Stats/Probability
3. Learn software engineering
4. Build

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.



Algorithm: neural network



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)

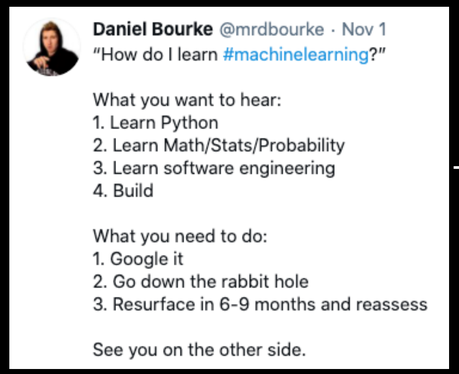
(depending how you represent your problem, many algorithms can be used for both)

Structured data ⟷ **Unstructured data**

“What are neural networks?”

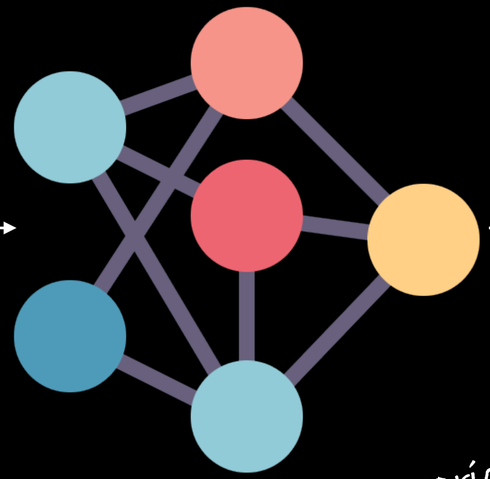
Neural Networks

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



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

Each of these nodes is called a "hidden unit" or "neuron".



(choose the appropriate neural network for your problem)

```
[[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

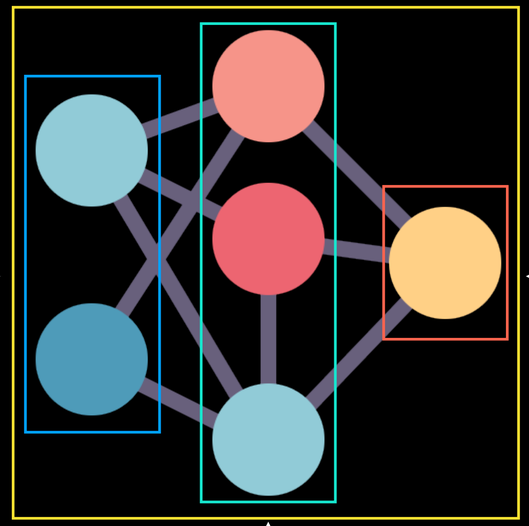
Not a diaster

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

Inputs | **Numerical encoding** | **Learns representation (patterns/features/weights)** | **Representation outputs** | **Outputs**

Anatomy of Neural Networks

Overall architecture



Input layer
(data goes in here)
units/neurons = 2

Output layer
(outputs learned representation or prediction probabilities)
units/neurons = 1

Each layer is usually combination of linear (straight line) and/or non-linear (not-straight line) functions

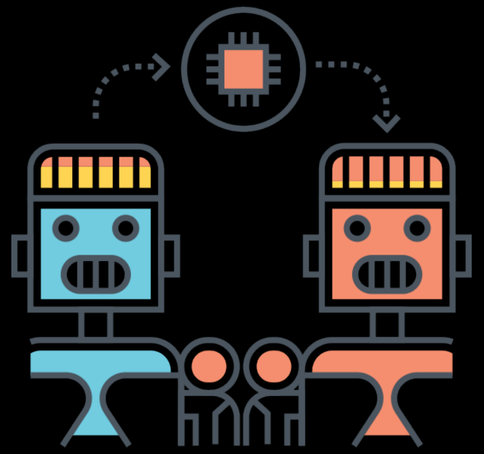
Hidden layer(s)
(learns patterns in data)
units/neurons = 3

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



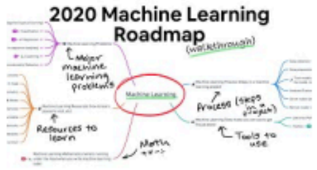
Transfer Learning

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



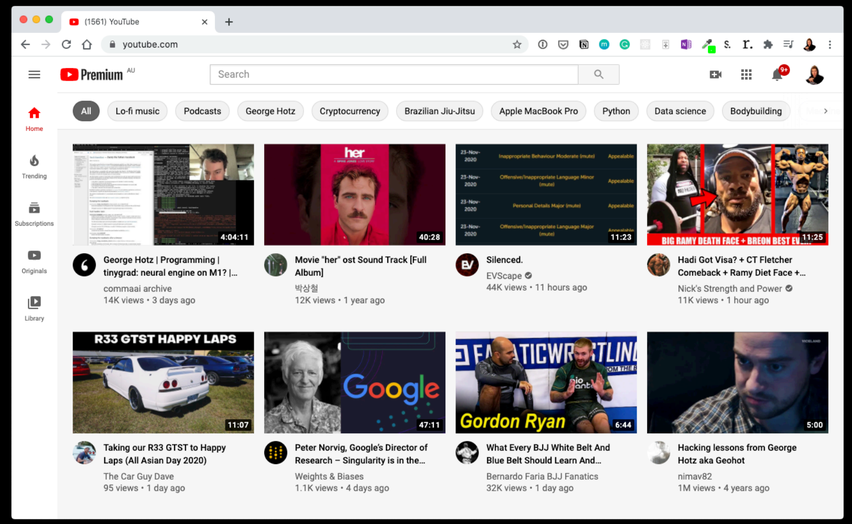
2020 Machine Learning Roadmap

Yashaswi Kulshreshtha

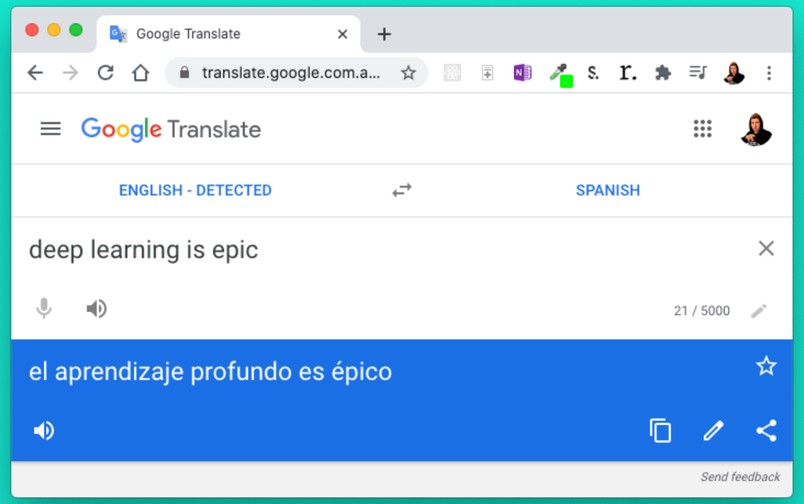
I think you can use ML for literally anything as long as you can convert it into numbers and program it to find patterns. Literally it could be anything any input or output from the universe

Source: 2020 Machine Learning Roadmap video.

(some) Deep Learning Use Cases



Recommendation

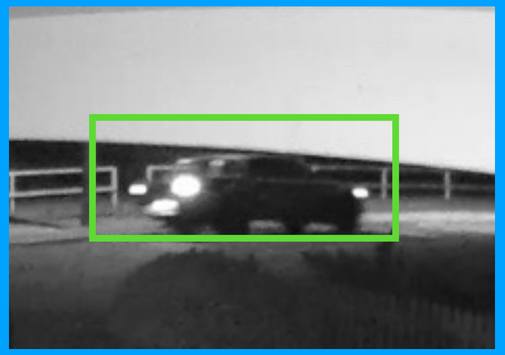


Translation

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

Speech recognition

Sequence to sequence (seq2seq)



Computer Vision

To: daniel@mrdbourke.com
Hey Daniel,

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

Not spam

To: daniel@mrdbourke.com
Hay daniel...

C0ongratu1ations! U win $1139239230

Spam

Natural Language Processing (NLP)

Classification/regression


PyTorch

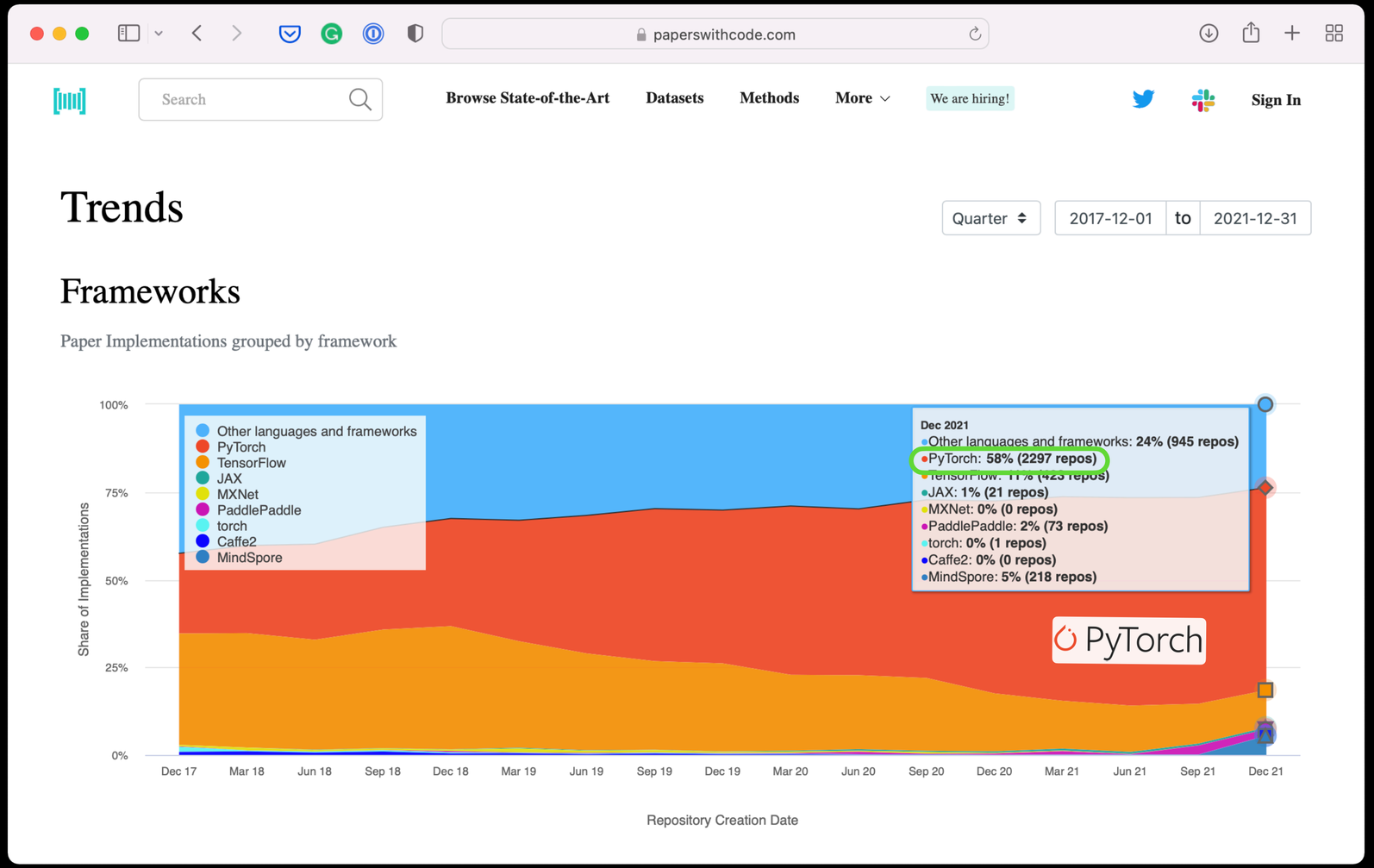
“What is PyTorch?”

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)

***Source:** paperswithcode.com/trends February 2022

Why PyTorch?



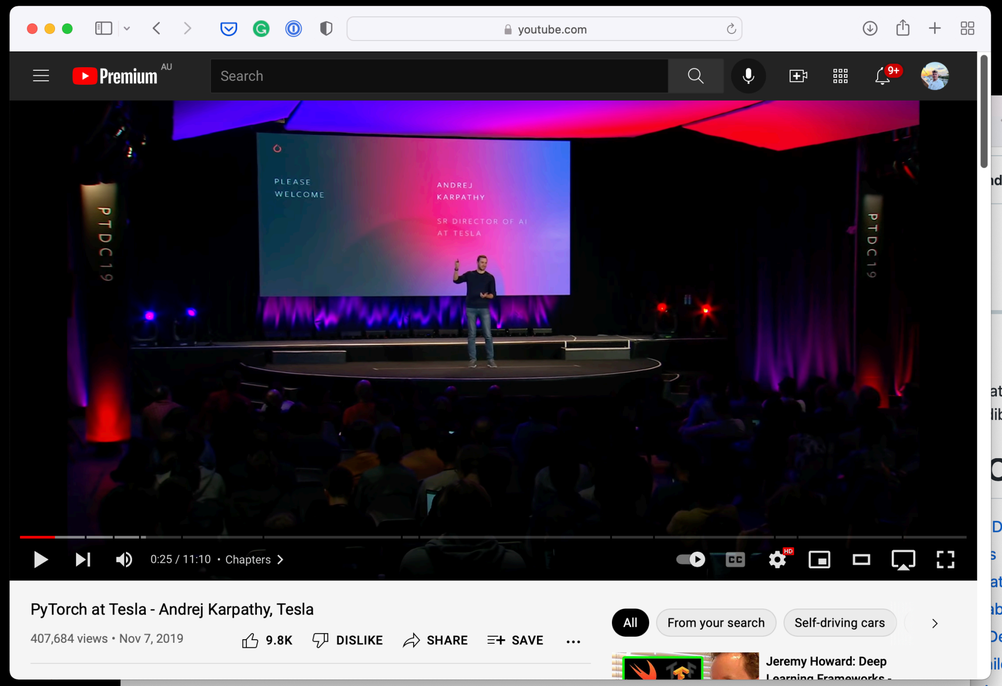
Research favourite

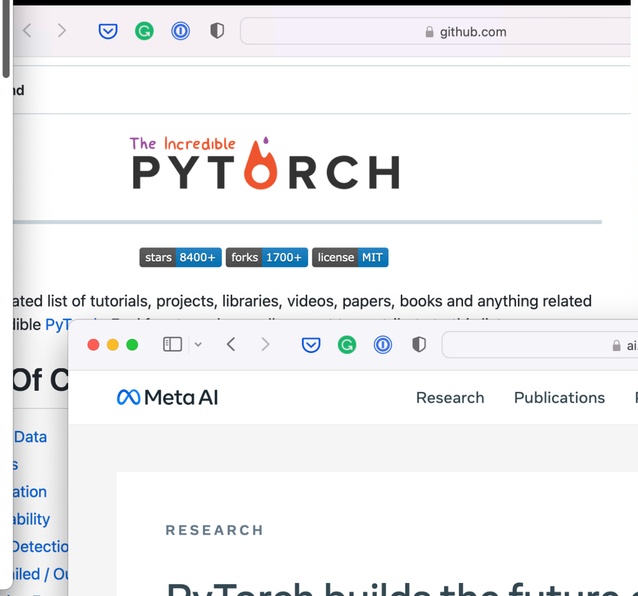
Source: paperswithcode.com/trends February 2022

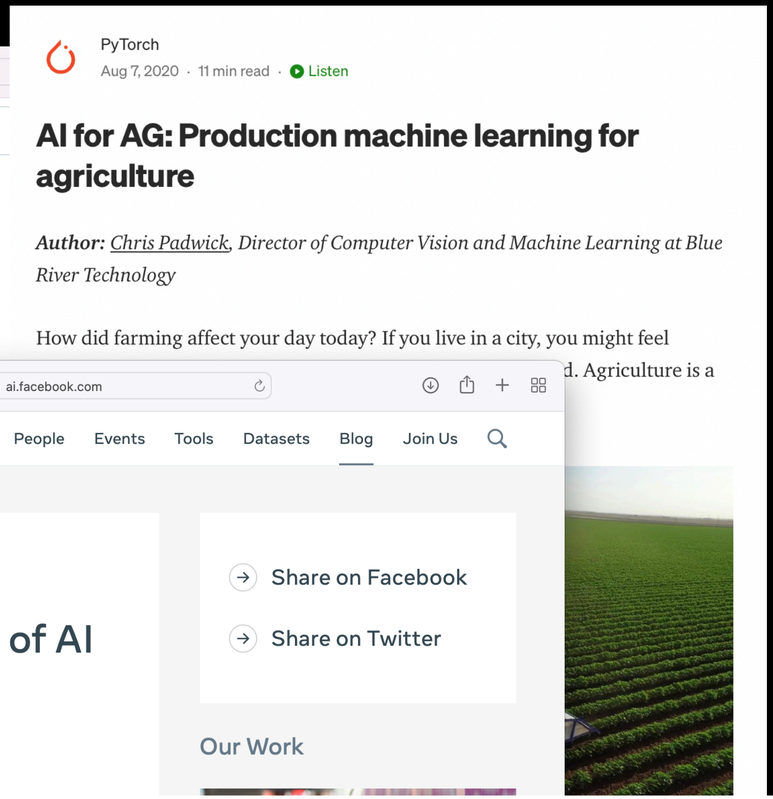
Why PyTorch?

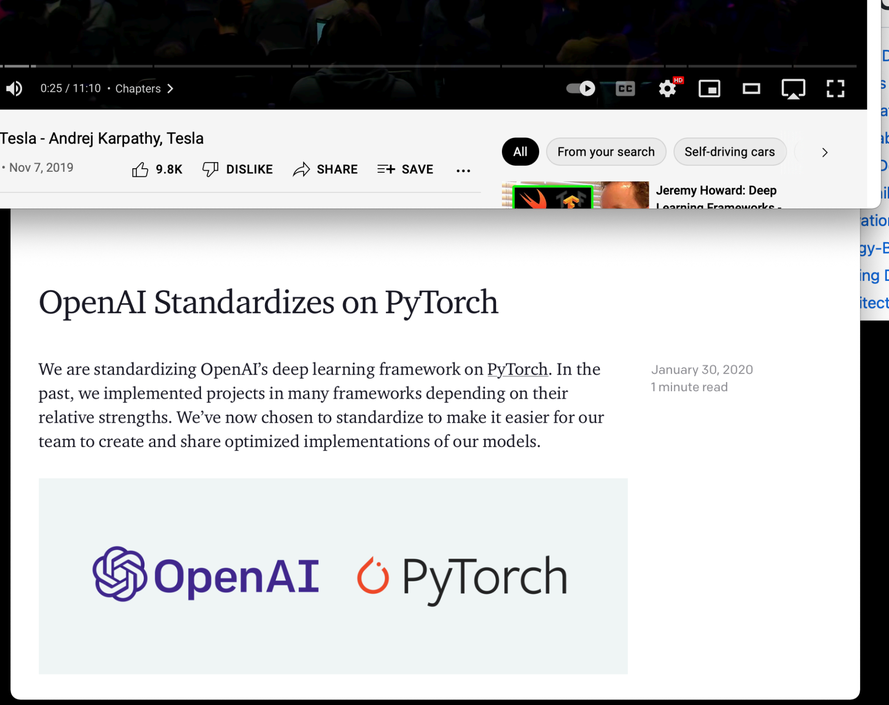
Source: @fchollet Twitter

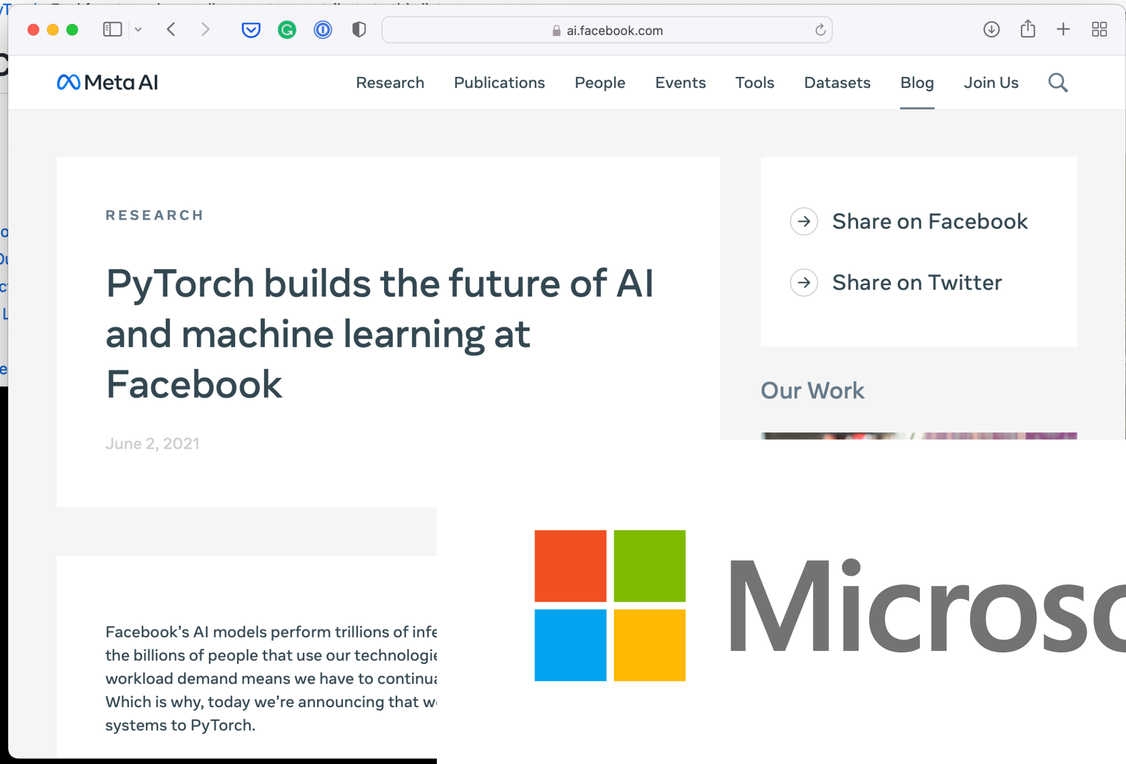
Why PyTorch?





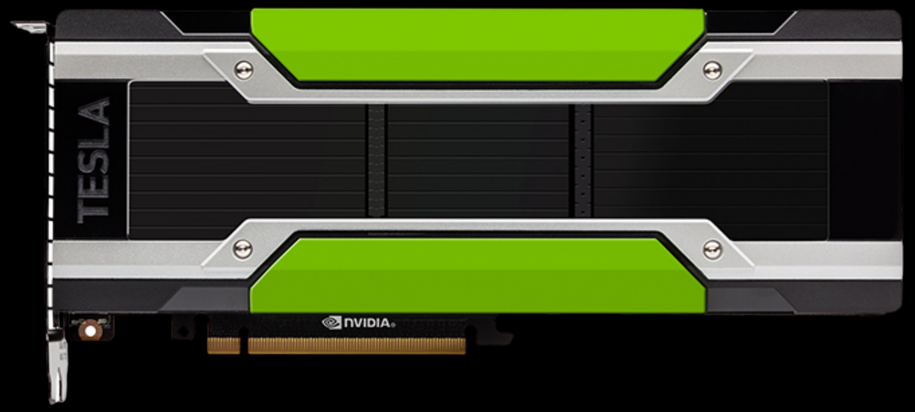






Microsoft

What is a GPU/TPU?



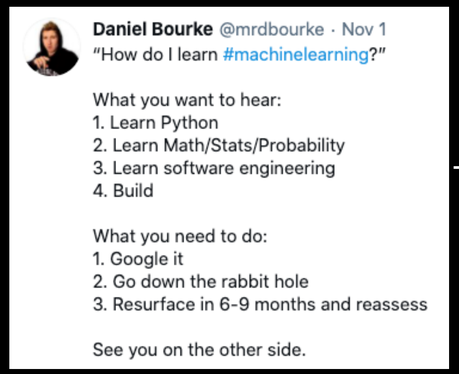
GPU (Graphics Processing Unit)

TPU (Tensor Processing Unit)

“What is a tensor?”

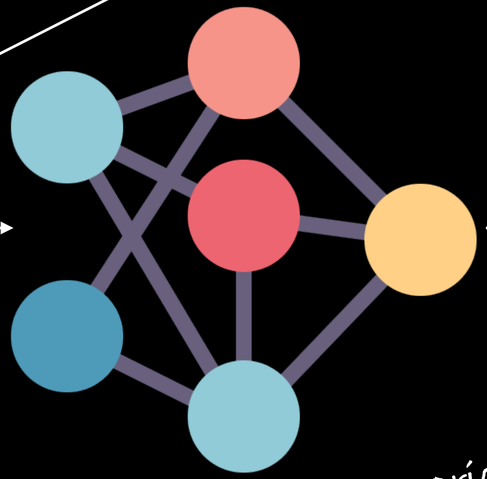
Neural Networks

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



These are tensors!

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



(choose the appropriate neural network for your problem)

```
[[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?"

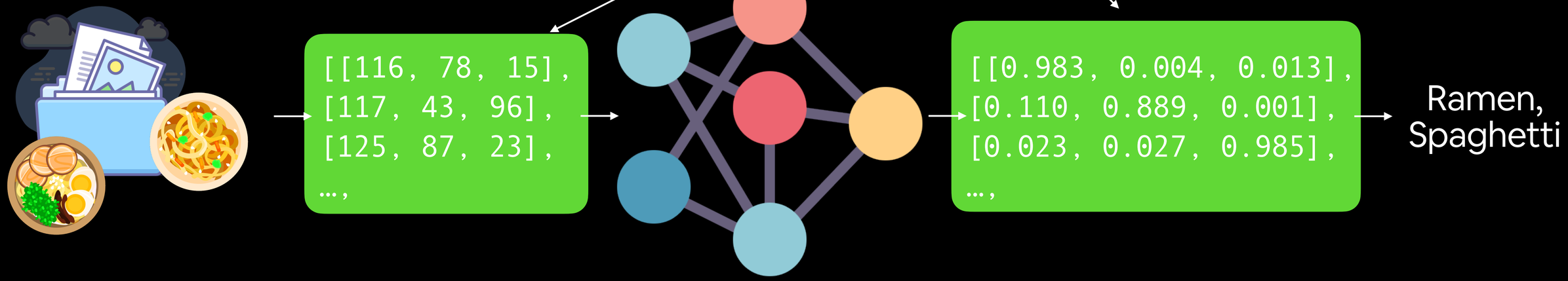
Inputs

Numerical encoding

Learns representation (patterns/features/weights)

Representation outputs

Outputs



Inputs

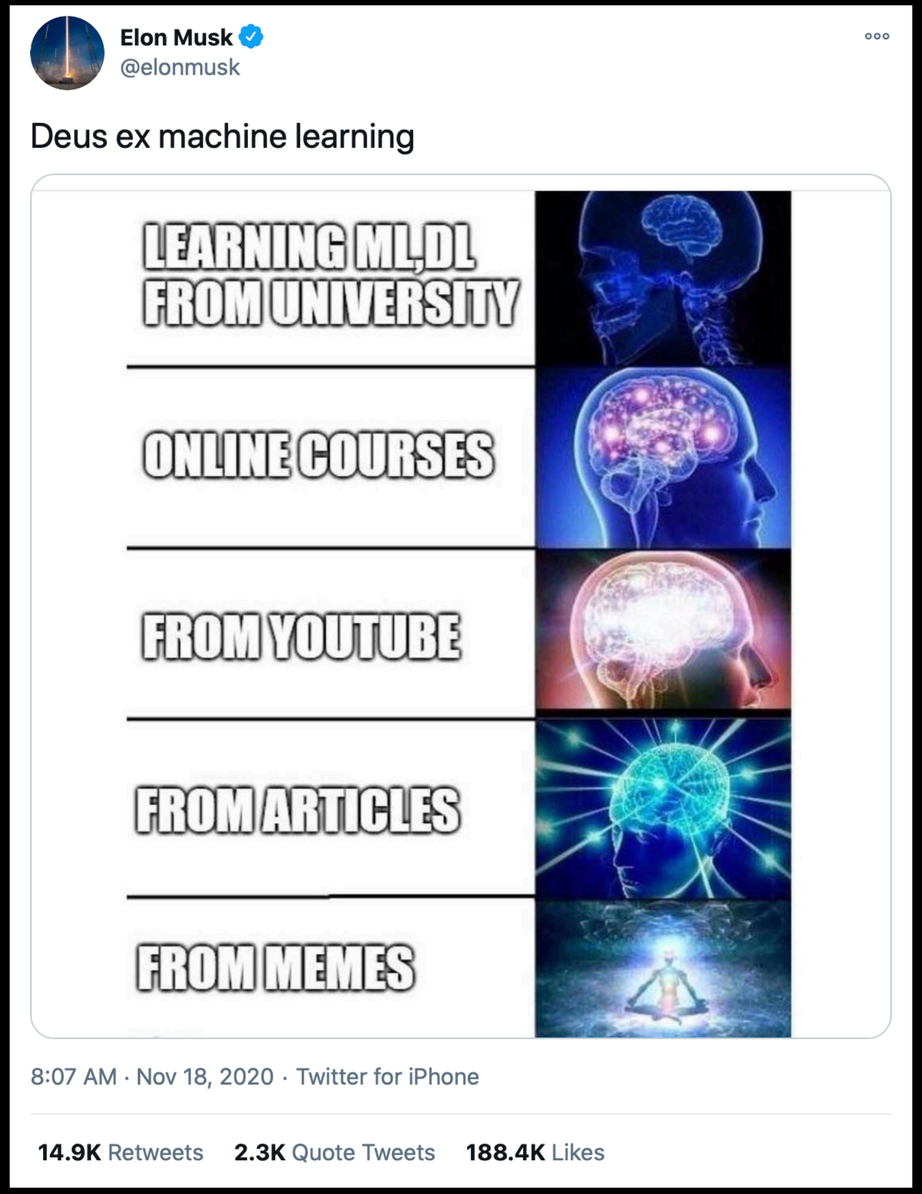
Numerical encoding

Learns representation (patterns/features/weights)

Representation outputs

Outputs

“What are we going to cover?”



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

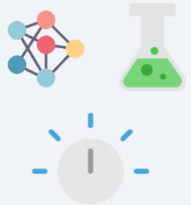
(one of many)

1. Get data ready (turn into tensors)

2. Build or pick a pretrained model (to suit your problem)

3. Fit the model to the data and make a prediction

4. Evaluate the model



5. Improve through experimentation

6. Save and reload your trained model

2.1 Pick a loss function & optimizer

2.2 Build a training loop

“How should I approach this course?”

How to approach this course

```python
# 1. Construct a model class that subclasses nn.Module
class CircleModelV0(nn.Module):
    def __init__(self):
        super().__init__()
        # 2. Create 2 nn.Linear layers
        self.layer_1 = nn.Linear(in_features=2, out_features=5)
        self.layer_2 = nn.Linear(in_features=5, out_features=1)

    # 3. Define a forward method containing the forward pass computation
    def forward(self, x):
        # Pass the data through both layers
        return self.layer_2(self.layer_1(x))

# 4. Create an instance of the model and send it to target device
model_0 = CircleModelV0().to(device)
model_0
```

1. Code along

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

Motto #2:
Experiment, experiment, experiment!

2. Explore and experiment

Motto #3:
Visualize, visualize, visualize!

3. Visualize what you don't understand

(including the "dumb" ones)

4. Ask questions

5. Do the exercises

6. Share your work

How not to approach this course

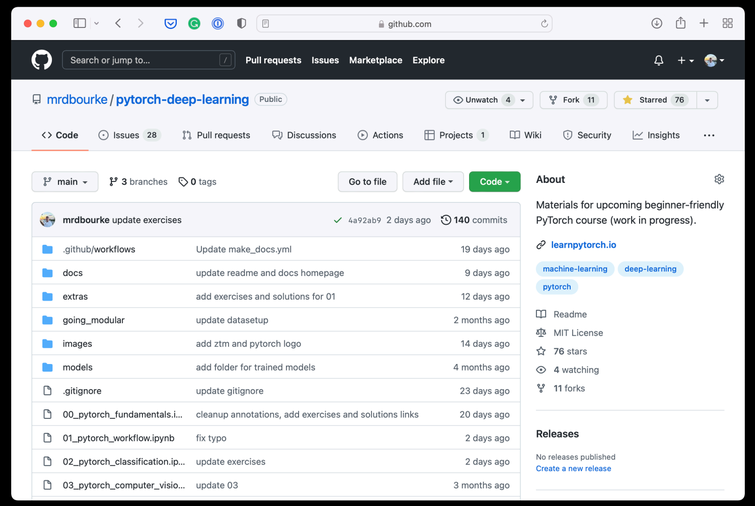
Avoid:

“I ~~can’t~~ learn _____”

Resources

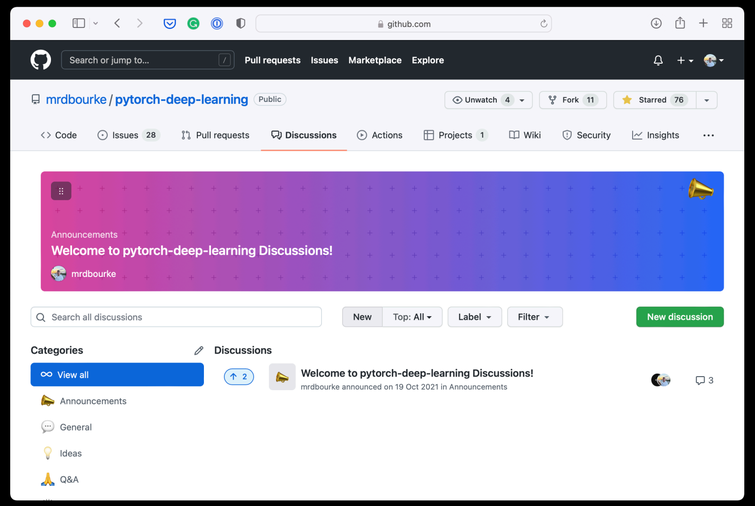
This course

Course materials



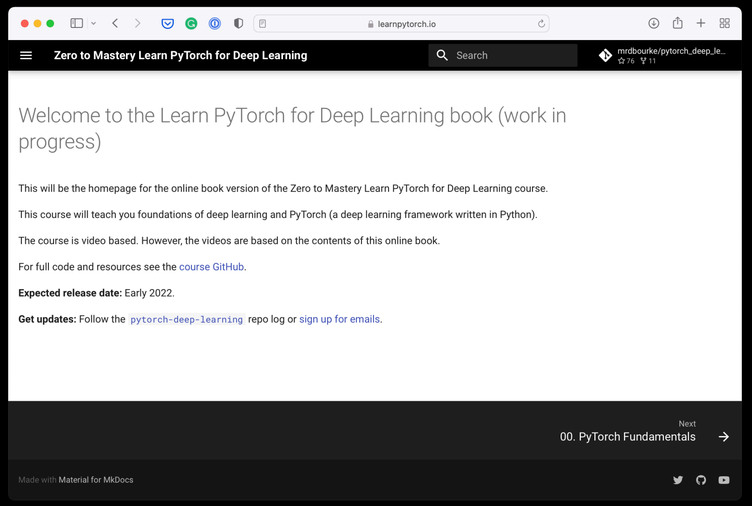
https://www.github.com/mrdbourke/pytorch-deep-learning

Course Q&A



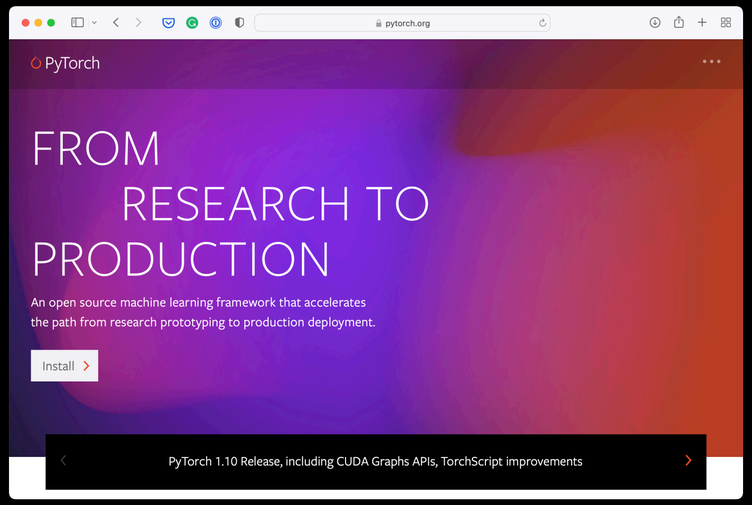
https://www.github.com/mrdbourke/pytorch-deep-learning/discussions

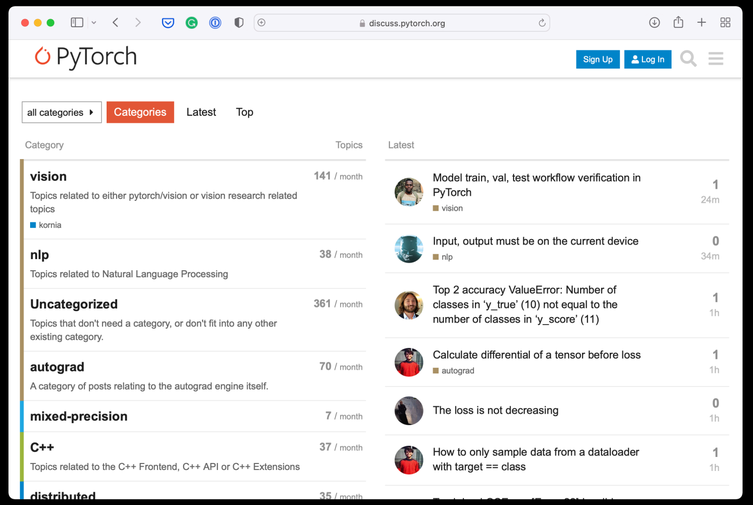
Course online book



https://learnpytorch.io

PyTorch website & forums

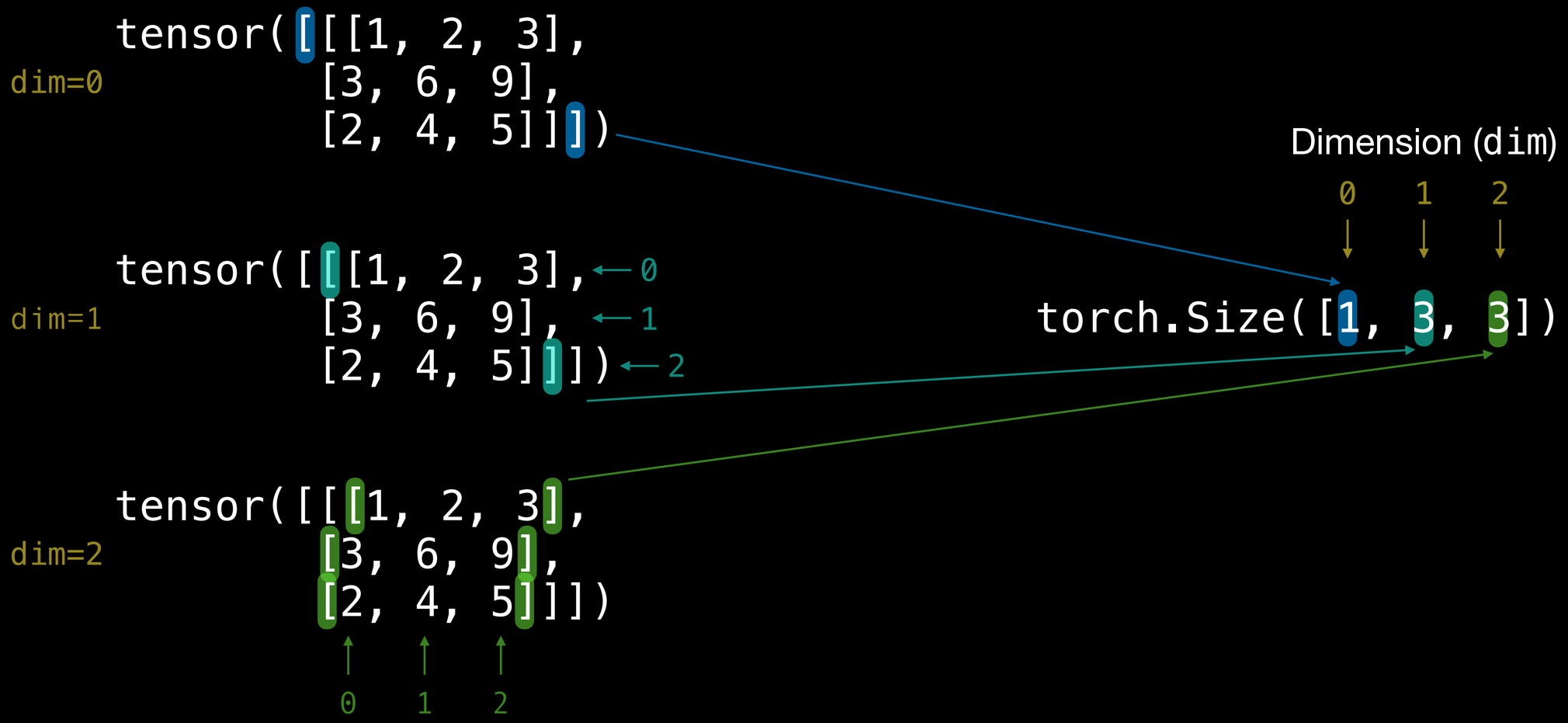




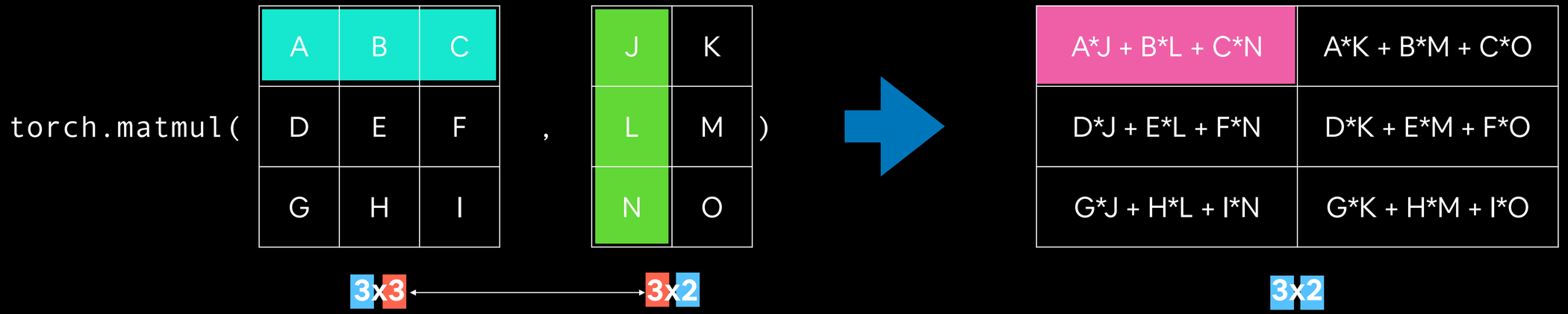
All things PyTorch

Let’s code!

Tensor dimensions

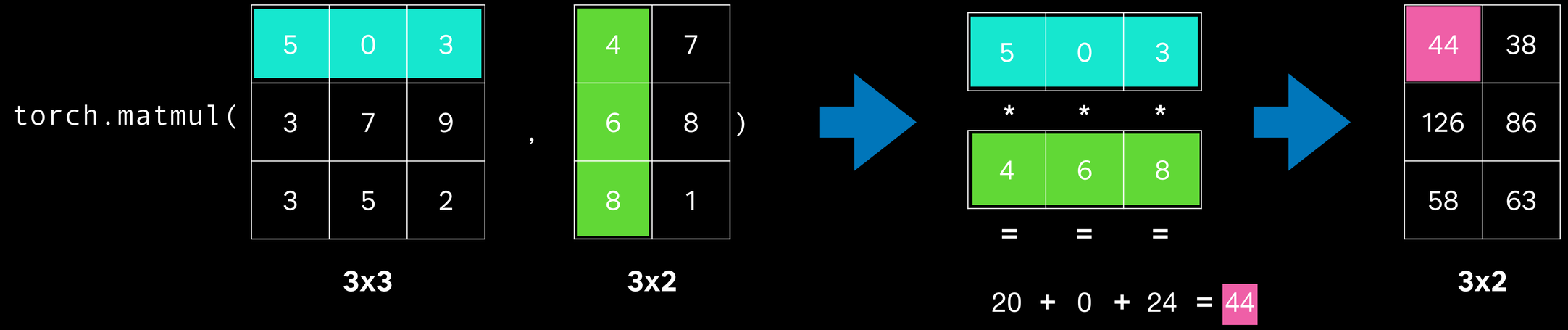


Dot product



Numbers on the inside must match

New size is same as outside numbers



For a live demo, checkout www.matrixmultiplication.xyz

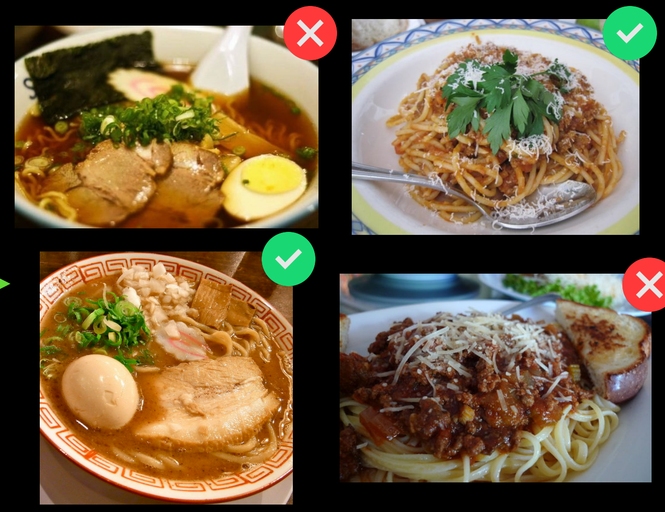
Supervised learning

(overview)

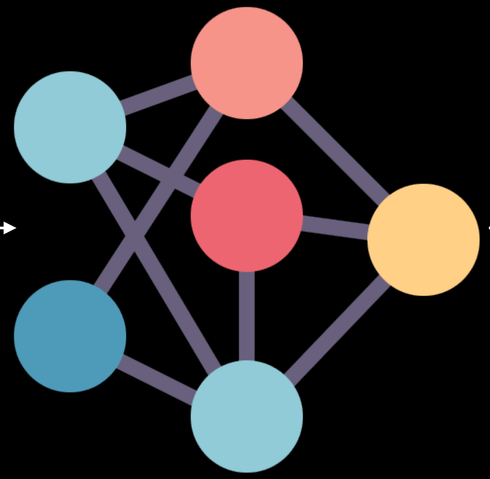
1. Initialise with random weights (only at beginning)

```
[[0.092, 0.210, 0.415],
[0.778, 0.929, 0.030],
[0.019, 0.182, 0.555],
…,
```

2. Show examples



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



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

Ramen, Spaghetti

3. Update representation outputs

4. Repeat with more examples

Inputs	Numerical encoding	Learns representation (patterns/features/weights)	Representation outputs	Outputs

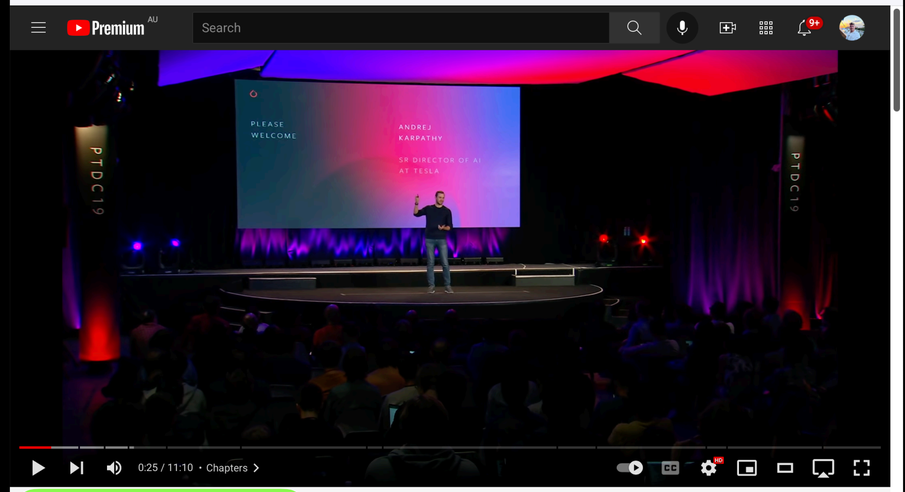
Tensor attributes

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

youtube.com

Premium AU

Search



0:25 / 11:10 • Chapters

PyTorch at Tesla - Andrej Karpathy, Tesla

407,684 views • Nov 7, 2019

9.8K DISLIKE SHARE SAVE

All | From your search | Self-driving cars

Jeremy Howard: Deep Learning Frameworks

openai.com

API RESEARCH BLOG ABOUT

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

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 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

ai.facebook.com

Meta AI Research Publications People Events Tools Datasets Blog Join Us

RESEARCH

PyTorch builds the future of AI and machine learning at Facebook

June 2, 2021

Share on Facebook

Share on Twitter

Our Work

RESEARCH

Facebook AI's Joelle

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.

paperswithcode.com

Search

Browse State-of-the-Art Datasets Methods More We are hiring! Sign In

Trends

Quarter 2017-12-01 to 2021-12-31