Lab2: Build an ANN

11210IPT 553000

Deep Learning in Biomedical Optical Imaging

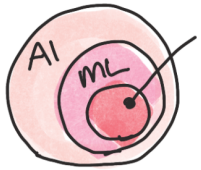
2023/09/25

Outlines

- Deep Learning Frameworks
- Kaggle & Dataset

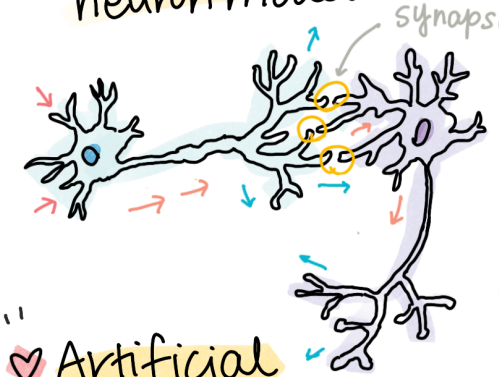
@girlie_mac
@shwars

Introduction to Neural Networks

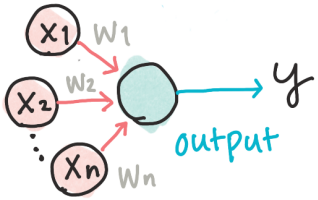


Neural Networks
↳ Deep Learning

Biological neuron model



Artificial

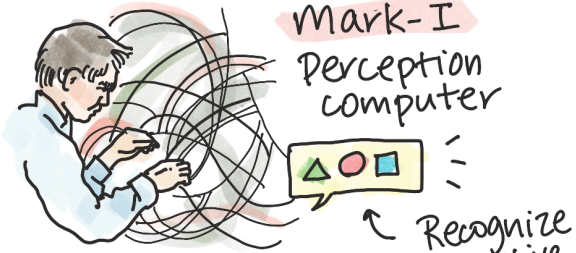


$$y = f\left(\sum_{i=1}^{n} x_i w_i\right)$$

Perception

= single-layer neural network ↔ multi-layer perception

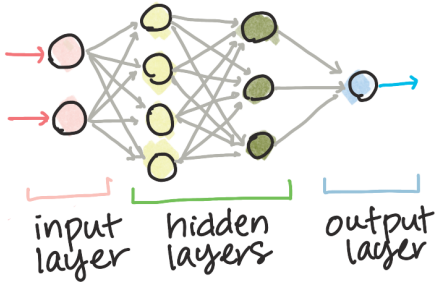
1957 Frank Rosenblatt



- 400 inputs
- 1 binary output

↖ Recognize primitive shapes

Back propagation method to train



Frameworks

↙ Low-level APIs

- build computational graph

Keras

↖ High-level APIs

source

Deep Learning Frameworks

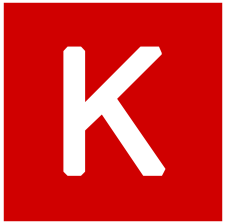
Deep Learning Frameworks

	Keras	TensorFlow	PyTorch
Level of API	high-level API[1]	Both high & low level APIs	Lower-level API[2]
Speed	Slow	High	High
Architecture	Simple, more readable and concise	Not very easy to use	Complex[3]
Debugging	No need to debug	Difficult to debugging	Good debugging capabilities
Dataset Compatibility	Slow & Small	Fast speed & large	Fast speed & large datasets
Popularity Rank	1	2	3
Uniqueness	Multiple back-end support	Object Detection Functionality	Flexibility & Short Training Duration
Created By	Not a library on its own	Created by Google	Created by Facebook[4]
Ease of use	User-friendly	Incomprehensive API	Integrated with Python language
Computational graphs used	Static graphs	Static graphs	Dynamic computation graphs[5]

TensorFlow vs Keras: A Comparison: https://towardsdatascience.com/tensorflow-vs-keras-d51f2d68fdfc
ML03: PyTorch vs. TensorFlow: https://medium.com/analytics-vidhya/ml03-9de2f0dbd62d

Deep Learning Frameworks

```
model = keras.models.Sequential()
model.add(keras.layers.Dense(1,input_shape=(2,),activation='sigmoid'))
model.summary()
```



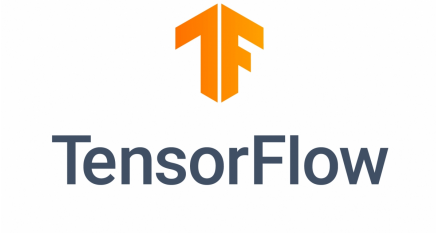
```
model.compile(optimizer=keras.optimizers.SGD(learning_rate=0.2),loss='binary_crossentropy',metrics=['acc'])
```

Reference

1. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroPyTorch.ipynb
2. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKerasTF.ipynb
3. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKeras.ipynb

Deep Learning Frameworks

```
W = tf.Variable(tf.random.normal(shape=(2,1)),dtype=tf.float32)
b = tf.Variable(tf.zeros(shape=(1,),dtype=tf.float32))

learning_rate = 0.1

@tf.function
def train_on_batch(x, y):
  with tf.GradientTape() as tape:
    z = tf.matmul(x, W) + b
    loss = tf.reduce_mean(tf.nn.sigmoid_cross_entropy_with_logits(labels=y,logits=z))
  dloss_dw, dloss_db = tape.gradient(loss, [W, b])
  W.assign_sub(learning_rate * dloss_dw)
  b.assign_sub(learning_rate * dloss_db)
  return loss
```



Reference

1. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroPyTorch.ipynb
2. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKerasTF.ipynb
3. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKeras.ipynb

Deep Learning Frameworks

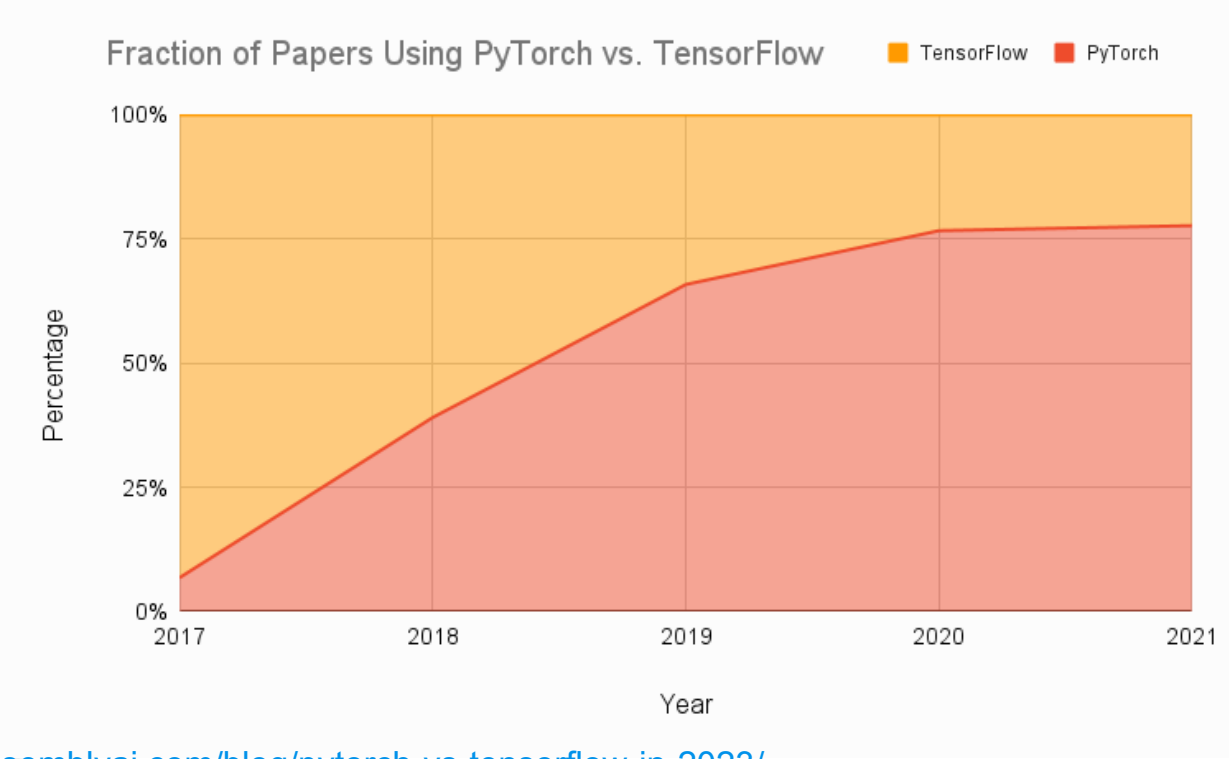
```
class Network():
  def __init__(self):
     self.W = torch.randn(size=(2,1),requires_grad=True)
     self.b = torch.zeros(size=(1,),requires_grad=True)

  def forward(self,x):
    return torch.matmul(x,self.W)+self.b

  def zero_grad(self):
    self.W.data.zero_()
    self.b.data.zero_()

  def update(self,lr=0.1):
    self.W.data.sub_(lr*self.W.grad)
    self.b.data.sub_(lr*self.b)

net = Network()
```


```
def train_on_batch(net, x, y):
  z = net.forward(x).flatten()
  loss = torch.nn.functional.binary_cross_entropy_with_logits(input=z,target=y)
  net.zero_grad()
  loss.backward()
  net.update()
  return loss
```

Reference

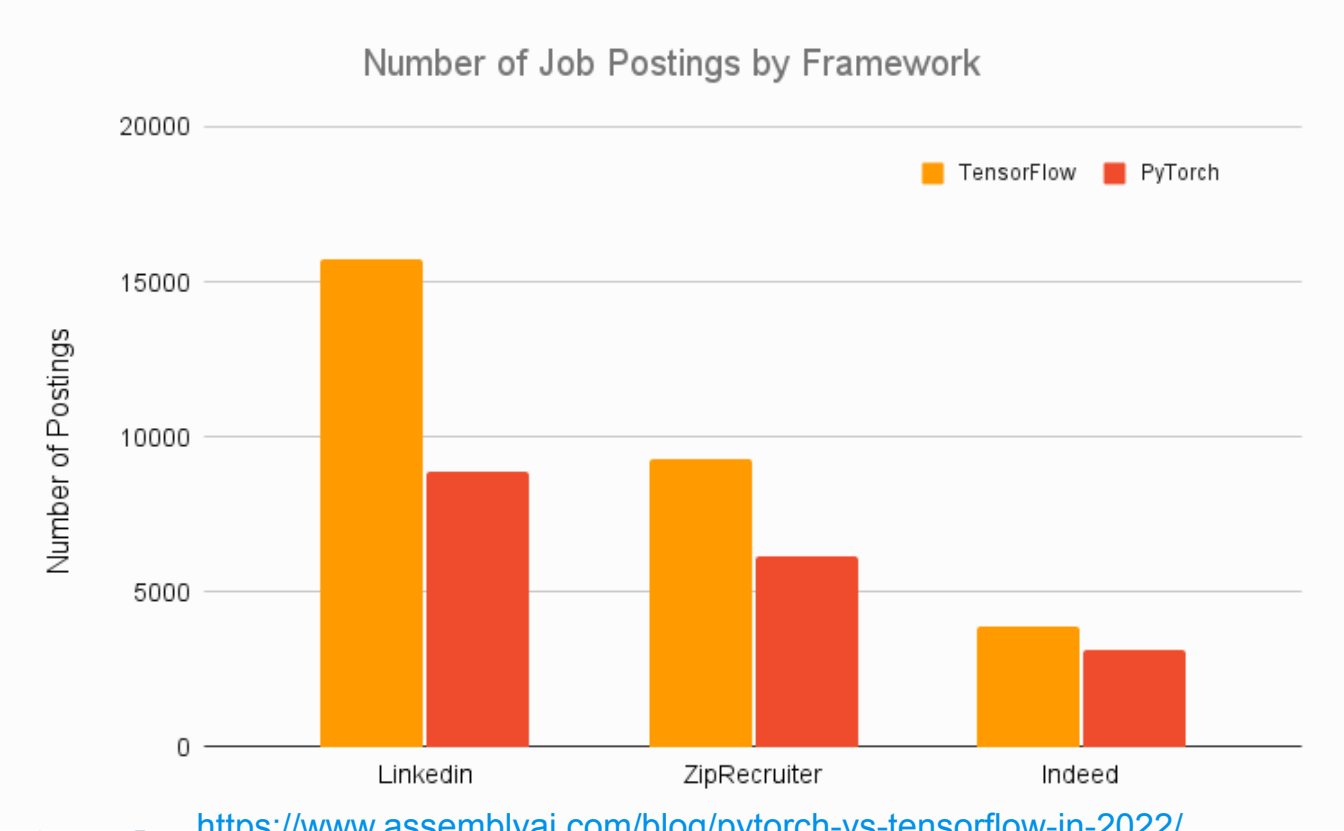
1. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroPyTorch.ipynb
2. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKerasTF.ipynb
3. https://github.com/microsoft/AI-For-Beginners/blob/main/lessons/3-NeuralNetworks/05-Frameworks/IntroKeras.ipynb

Frameworks - Research Papers



https://www.assemblyai.com/blog/pytorch-vs-tensorflow-in-2023/
https://thegradient.pub/state-of-ml-frameworks-2019-pytorch-dominates-research-tensorflow-dominates-industry/

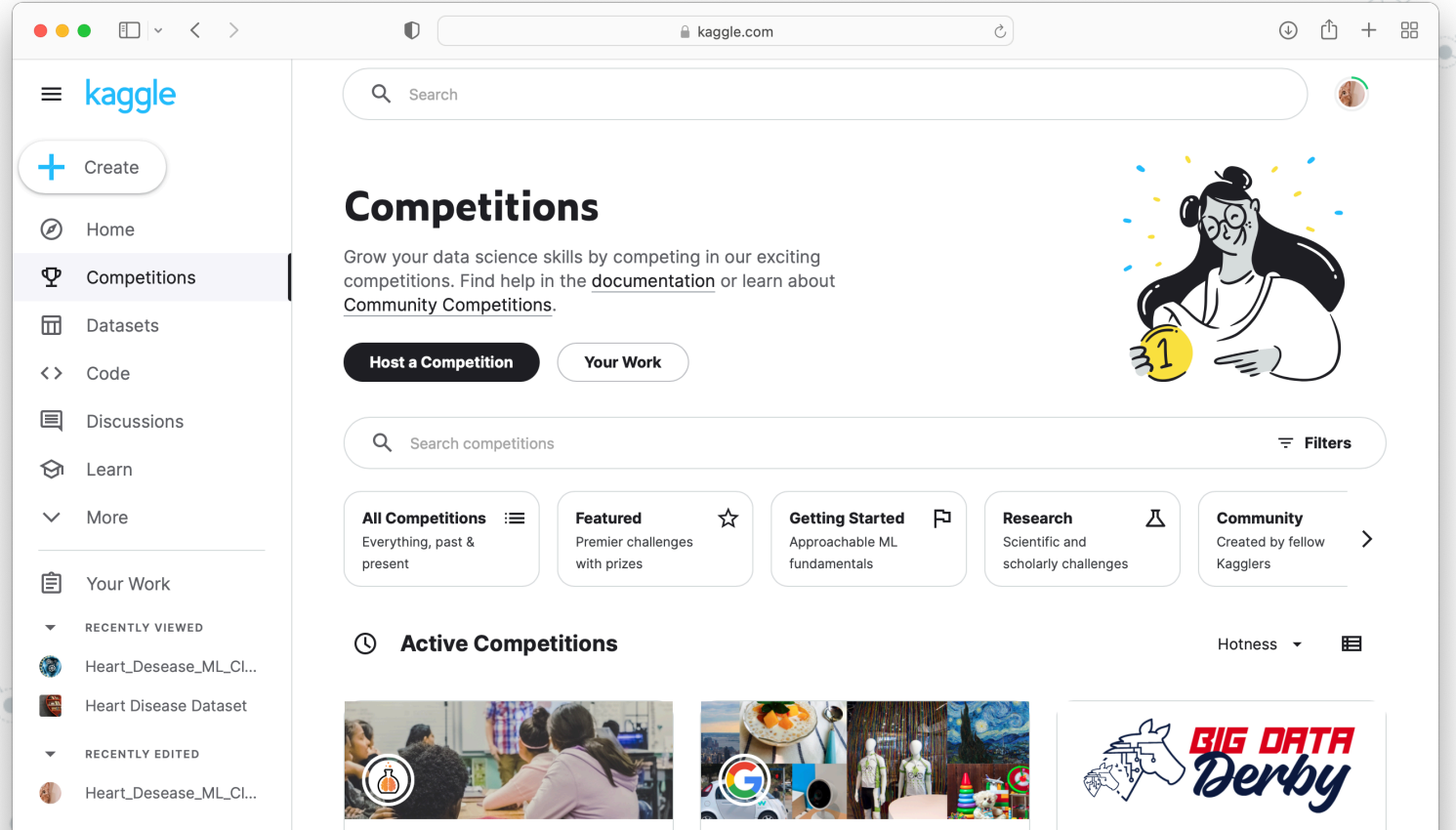
Frameworks - Jobs



https://www.assemblyai.com/blog/pytorch-vs-tensorflow-in-2022/

Kaggle

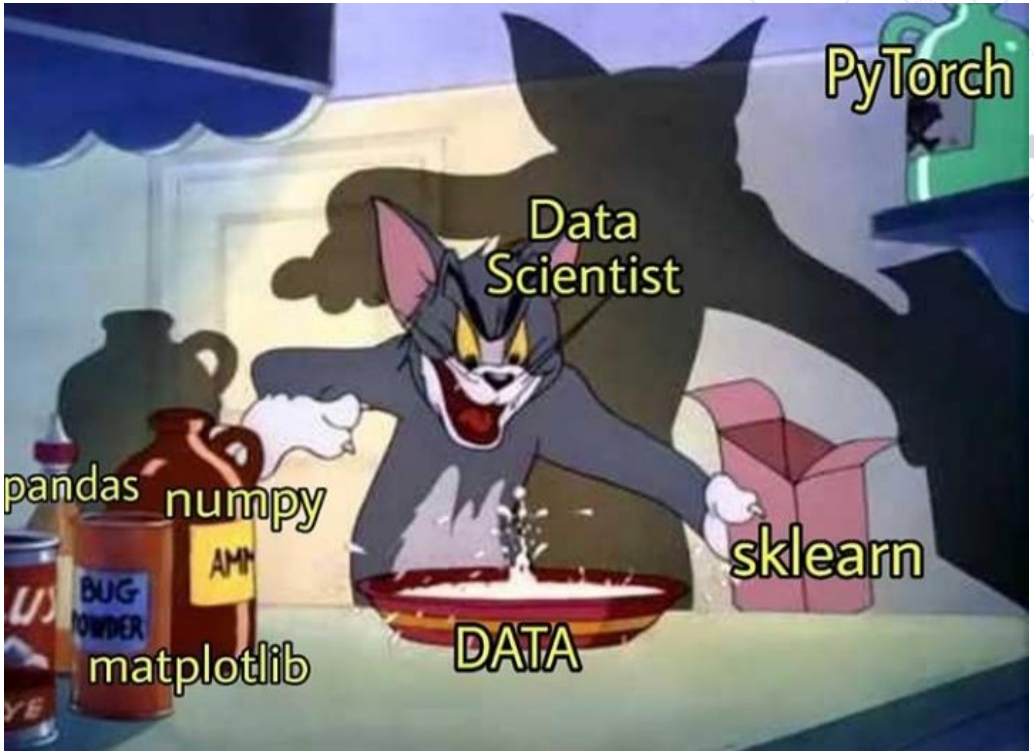
https://www.kaggle.com/



Chest X-Ray Image Dataset

- This is an open dataset from Kaggle.
- It has 3 folders (train, test, val) and 2 classes (pneumonia/ normal) for each category.
- The original dataset is an imbalanced dataset, the ratio of pneumonia and normal is roughly 2.5:1.
- For the lab, we took 1000 images for each category and resized all images into 256×256.
- The pixel values are range by [0, 255].

CODING TIME!!


PyTorch
Data Scientist
pandas
numpy
sklearn
BUG POWDER
matplotlib
DATA