Introduction to Tensorflow for NFL Sports Analytics

Pranav Tavildar

UConn Sports Analytics Symposium 2022

October 8, 2022

Table of contents

1. Introduction to Tensorflow

2. Concepts and Basis

3. Neural Networks

4. Making our Model

5. Conclusion

What is Tensorflow?

Open Source Deep Learning Library that makes creating Models more:

- Easy
- Flexible
- Fast
- Reproducable

The History of Tensorflow

Evolution of Tensorflow

- Product of Google Brain Team
- Evolution from Distbelief (2011)
- Possible because of Tensor Processing Unit

Tensorflow vs Alternatives

Tensorflow

- Preferred by Companies
- Easily Deployable and Versatile

Pytorch

- Preferred by Research Labs
- Easily Debuggable

When you're the only one of your friends who uses PyTorch instead of TensorFlow

why I am more successful than you.

Caffe

- Used by Researchers and Startups
- Useful for Small but Specific and fast use cases

What is a Tensor?

- Tensor is data with dimension.
- 0-d tensor: scalar

$$c = 5$$

- 1-d tensor: vector

$$\mathrm{c} = \begin{pmatrix} 1 \\ \vdots \\ 5 \end{pmatrix}$$

- 2-d tensor: matrix

$$\mathrm{c} = \begin{pmatrix} 1 & \cdots & 5 \\ \vdots & \ddots & \vdots \\ 5 & \cdots & 5 \end{pmatrix}$$

Why Use Tensors?

- Vector, matrix operations and gradient (derivative) are dominant in machine learning and deep learning.
- GPU structure leads to a powerful ability to solve linear tensor operations.

How to install Tensorflow Locally

- Anaconda management (GUI): click "environment" –> choose "Not installed" –> search "tensorflow"
- Anaconda Prompt:

  ```
  conda install tensorflow
  ```

- Pip:

  ```
  pip install tensorflow
  ```

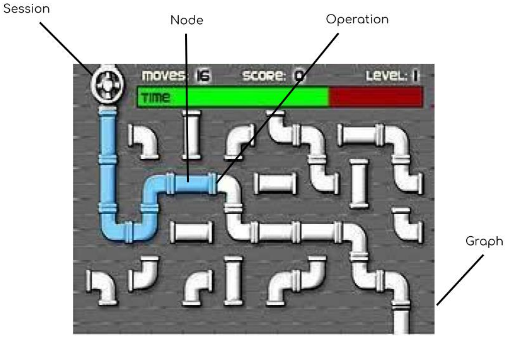
- Verify whether your installation is successful: (open jupyter notebook or run at .py file)

```python
import tensorflow as tf
import tensorflow.compat.v1 as tfv1
tf.__version__
```

Components of Tensorflow

- Operations: Linear Algebra Data Operations
- Graph: A skeleton structure which holds all the data and operations
- Nodes: Contains Constants and Variables
- Session: The *flow in tensorflow. Not used anymore in Tensorflow V2

An Illustration Using Pipes



Session

Node

Operation

Graph

Tensorflow V1 versus V2

- V1: Nodes → Operations → Graph → Session → import the data
- V2: import the data → Nodes → Operations (eager execution)

Figure: Me when I see how much easier it is to develop with eager execution

Let's get into the code!

Open the tv1vtf2.ipynb file from the code

Basic Operations

Given x and y,

- x+y (element-wise): tf.add(x,y)
- x-y (element-wise): tf.subtract(x,y)
- x*y (element-wise): tf.multiply(x,y)
- x/y (element-wise): tf.divide(x,y)
- x*y (matrix style): tf.matmul(x,y)
- x<y (judgment): tf.less(x,y)
- x>y (judgment): tf.greater(x,y)
- x<=y (judgment): tf.less_equal(x,y)

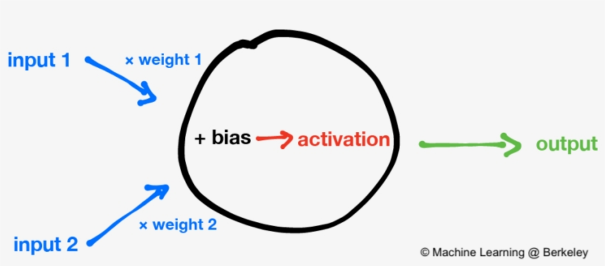
ANN

One of the greatest strengths of tensorflow is its ability to mimic learning and make predictions using data. This is done through structures known as Neural Networks. In the scope of this workshop, we're only covering Artificial Neural Networks known as ANNs.

The Neuron

Approach like a black box with input and output. The following are components of a Neuron

- Node: Has a Bias
- Weights: this is the connection between nodes
- Activation: A function that determines how the neuron will fire



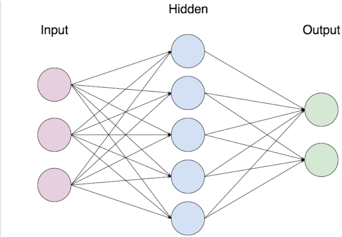
© Machine Learning @ Berkeley

Network

Many neurons together make up a layer of which there are three types

- Input
- Output
- Hidden

During training the weights in the hidden layer will be adjusted in a process known as Back Propagation in order to improve accuracy



Gradient Descent

Gradient Descent is how the Machine Learns! This is a feedback cycle that aims to minimize the loss function by descending down the derivative of the lost function until you reach the minimum

Example

$$(GD) : \hat{\beta}^{(t+1)} = \hat{\beta}^{(t)} - \gamma \nabla E_n \left[\ell \left(y, \hat{y} \left(\hat{\beta}^{(t)} \right) \right) \right]$$

NFL Play-By-Play Analysis

Now it is time to create our Tensorflow Model! We're using play-by-play data from the NFL 2021-2022 season gathered from NFLSavant. Let's try to create a model that predicts whether the next play is going to be a first down given data.

Takeaways

For Data Engineering

- Drop values in order to get rid of empty, redundant, or outlying/influential points
- Encode Categorical Data using techniques like one-hot encoding
- Normalize your data before using it.

For Building and Analyzing Model

- Use sigmoid activation function for output layers with one node
- Use binary cross-entropy as a loss function
- Watch out for overfitting

Shameless Promo

Thank you very much for attending! The material for this workshop is accessible here –> `Session Material`.

Join UConn Data Science Club!

Our UConntact –> `Our UConntact`.

Our Discord –> `Discord`.