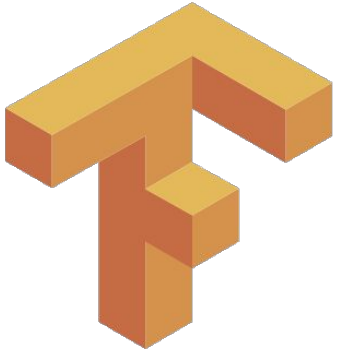
inovex

Neural Networks & Deep Learning



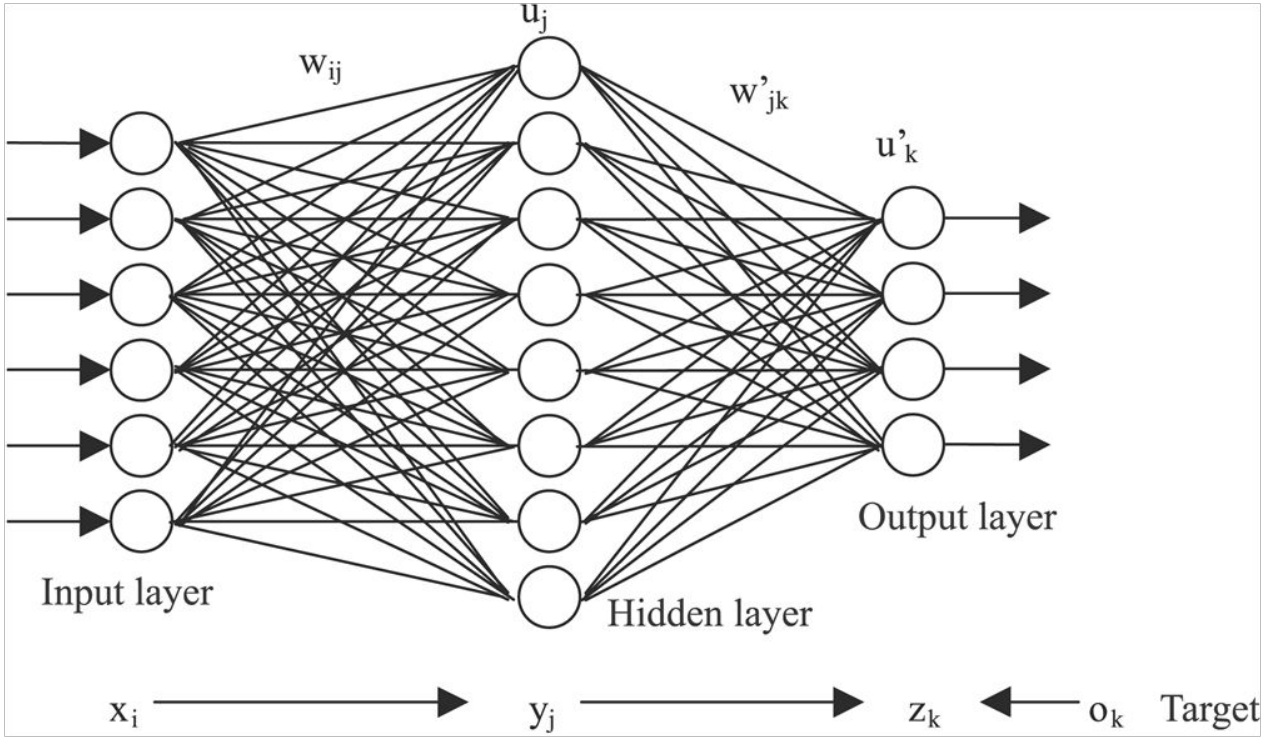
Neural Networks & Deep Learning

Overview

- first idea of neural networks developed in the 40's by Warren McCulloch and Walter Pitts
- abstraction of the human brain
- hard to train → sank into obscurity
- renaissance with Deep Learning

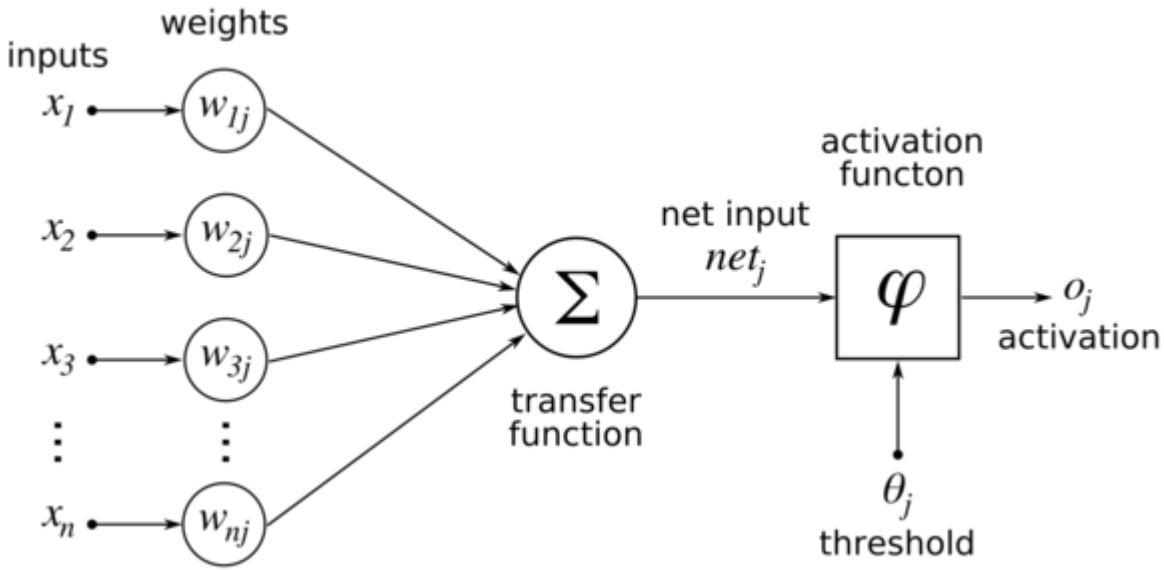
Neural Networks & Deep Learning

Overview



Neural Networks & Deep Learning

Neurons & Edges



Neural Networks & Deep Learning

Layer

- Input Layer → receives data
- Hidden Layer → computation
- Output Layer → outputs results

Neural Networks & Deep Learning

Overview

- a very complex function with a lot of parameters
- as you add more hidden layers complexity will rise
- complexity → find good parameters

How to train neural networks if they are so complex ???

Neural Networks & Deep Learning

Training

- Backpropagation
  - Gradient Descent & Chain Rule
- training in batches of data
- training on GPUs → very fast!

awesome explanation how backpropagation works:

https://ayearofai.com/rohan-lenny-1-neural-networks-the-backpropagation-algorithm-explained-abf4609d4f9d

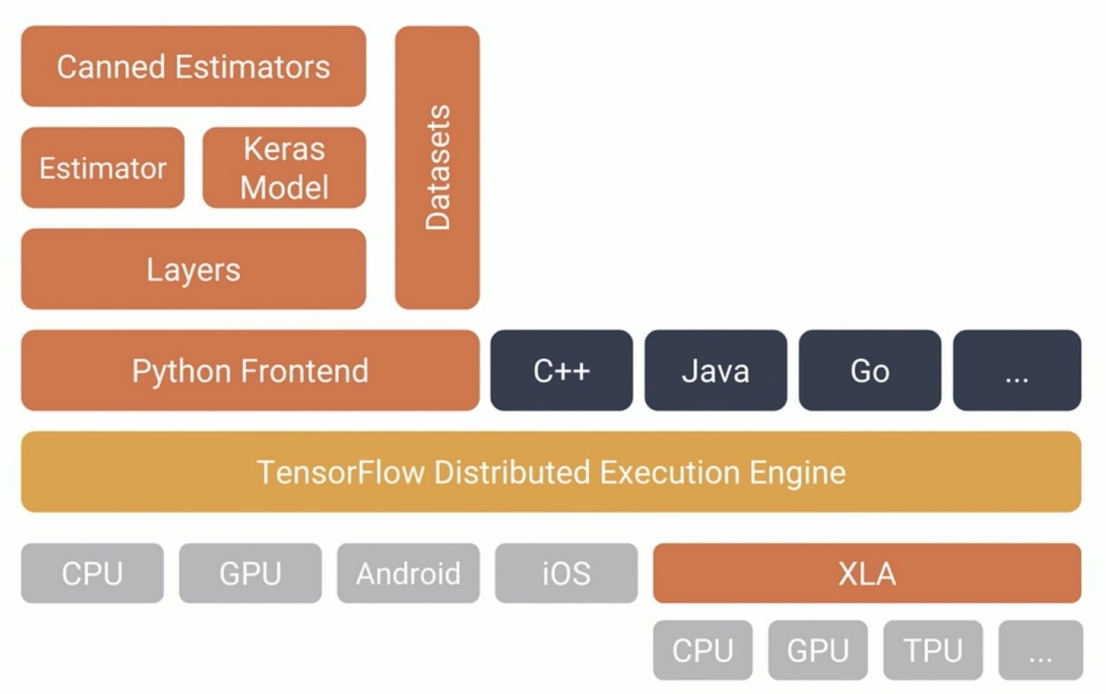
Neural Networks & Deep Learning

Deep Learning

- based on neural networks
- hipster rebranding of neural networks
- very deep neural networks → high complexity
- special architectures → Click here
- automated feature engineering
- capable to solve very challenging tasks

Neural Networks & Deep Learning

Neural Networks in TensorFlow



Neural Networks & Deep Learning

Neural Networks in TensorFlow - Example Code

```
import tensorflow as tf


# data input pipeline & placeholder definition
...
# net definition
dense1 = tf.layers.dense(input_placeholder,1000,activation=tf.nn.relu)
logits = tf.layers.dense(dense1,10,activation=tf.nn.relu)
output = tf.nn.softmax(logits)

# loss & train operation
loss = tf.losses.softmax_cross_entropy(labels_placeholder,logits)
train_op = tf.train.AdamOptimizer(learning_rate=0.001).minimize(loss)
```

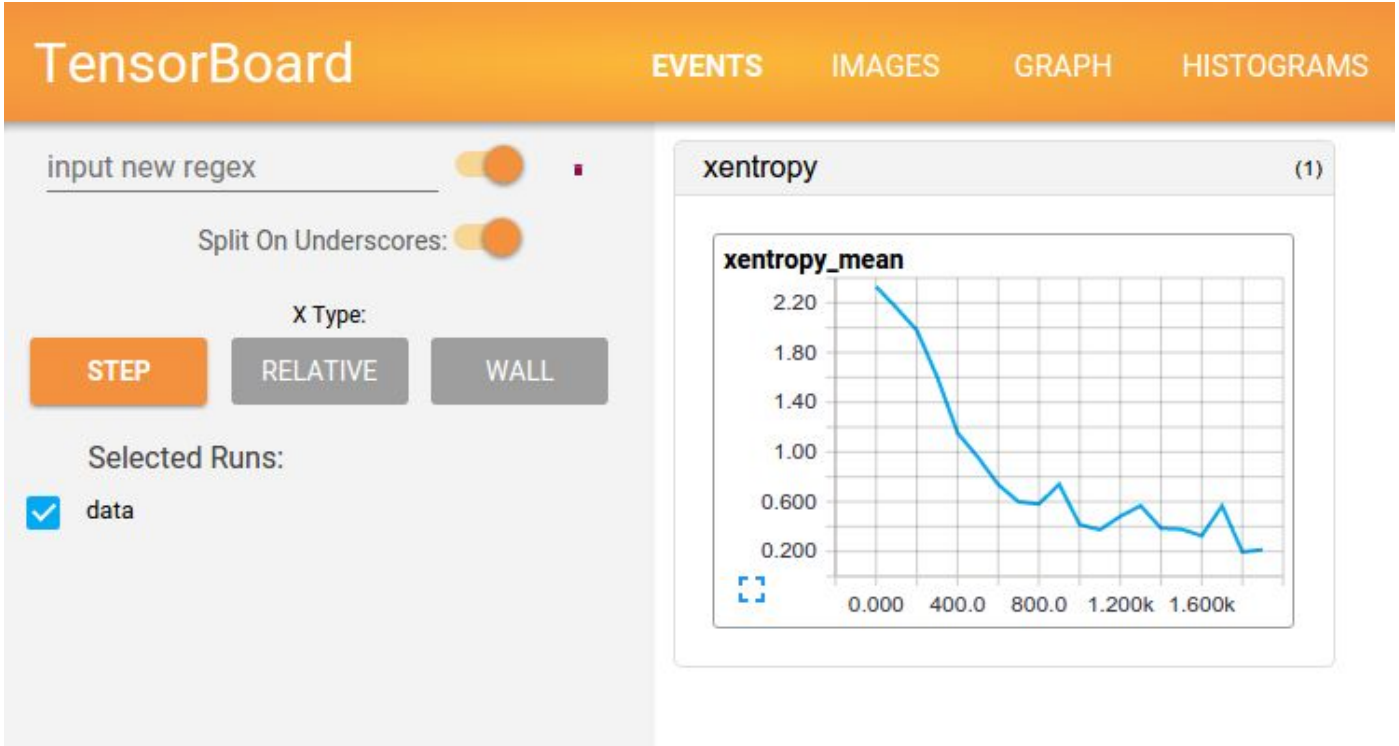
Neural Networks & Deep Learning

Neural Networks in TensorFlow - Example Code

```
init = tf.global_variables_initializer()

with tf.Session() as sess:
  sess.run(init)

  # training loop
  for i in range(1000):
    sess.run(train_op,feed_dict={input_placeholder: input_data_batch, labels_placeholder:input_labels_batch})

...
```

Neural Networks & Deep Learning

TensorBoard



now let’s see what neural networks look like:

http://playground.tensorflow.org/