Introduction to Artificial Intelligence
Deep Learning - Tensor Flow

Janyl Jumadinova
December 2, 2016

Credit: Google Workshop
Neural Networks

["this", "movie", "was", "great"]

Input →

Hidden →

Output

(score) →

[.7]
Neural Networks

Input Hidden Output (label)

pixels( )

["cat"]
A fully connected NN layer

\[ y_1 = \sigma \left( W_{1,1} x_1 + W_{1,2} x_2 + W_{1,3} x_3 + b_1 \right) \]
\[ y_2 = \sigma \left( W_{2,1} x_1 + W_{2,2} x_2 + W_{2,3} x_3 + b_2 \right) \]
\[ y_3 = \sigma \left( W_{3,1} x_1 + W_{3,2} x_2 + W_{3,3} x_3 + b_3 \right) \]
Implementation as Matrix Multiplication

\[
\begin{align*}
  y_1 &= \sigma \left( W_{1,1} x_1 + W_{1,2} x_2 + W_{1,3} x_3 + b_1 \right) \\
  y_2 &= \sigma \left( W_{2,1} x_1 + W_{2,2} x_2 + W_{2,3} x_3 + b_2 \right) \\
  y_3 &= \sigma \left( W_{3,1} x_1 + W_{3,2} x_2 + W_{3,3} x_3 + b_3 \right)
\end{align*}
\]

\[
\begin{bmatrix}
  y_1 \\
  y_2 \\
  y_3
\end{bmatrix} = \sigma \left( \begin{bmatrix}
  W_{1,1} & W_{1,2} & W_{1,3} \\
  W_{2,1} & W_{2,2} & W_{2,3} \\
  W_{3,1} & W_{3,2} & W_{3,3}
\end{bmatrix} \begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix} + \begin{bmatrix}
  b_1 \\
  b_2 \\
  b_3
\end{bmatrix} \right)
\]
Non-Linear Data Distributions
Each neuron implements a relatively simple mathematical function.

\[ y = g(\overline{w} \cdot \overline{x} + b) \]
Each neuron implements a relatively simple mathematical function.

\[ y = g(\bar{w} \cdot \bar{x} + b) \]

The composition of $10^6 - 10^9$ such functions is powerful.
“A core idea in deep learning is that we assume that the data was generated by the composition of factors or features, potentially at multiple levels in a hierarchy.”
Results get better with:

- more data
- bigger models
- more computation
Results get better with:

- more data
- bigger models
- more computation

Better algorithms, new insights and improved methods help, too!
- **Open source** Machine Learning library
- Especially useful for **Deep Learning**
- For research and production
- **Apache 2.0** license
- [tensorflow.org](http://tensorflow.org)
Adoption of Deep Learning Tools on GitHub

TensorFlow (GitHub Launch Nov. 2015)
- GitHub Stars: 25,041
- GitHub Forks: 9,607

Caffe (GitHub Launch Sep. 2013)
- GitHub Stars: 10,474
- GitHub Forks: 6,227

Torch (GitHub Launch Jan. 2012)
- GitHub Stars: 4,671
- GitHub Forks: 1,288

Theano (GitHub Launch Jan. 2008)
- GitHub Stars: 3,829
- GitHub Forks: 1,399
Tensor Flow

- Operates over **tensors**: n-dimensional arrays
Tensor Flow

- Operates over **tensors**: n-dimensional arrays
- Using a **flow graph**: data flow computation framework
Tensor Flow

- Operates over **tensors**: n-dimensional arrays
- Using a **flow graph**: data flow computation framework

A multidimensional array.

A graph of operations.
Tensor Flow

- 5.7 ← Scalar
- Number, Float, etc.
Tensor Flow

\((x_1, x_2, x_3, \ldots, x_n) \leftrightarrow \text{Vector! (List, Tuple)}\)
Tensor Flow

\[
\begin{align*}
(x_1, x_2, x_3, \ldots, x_n) \\
\end{align*}
\]

\[
\begin{align*}
(x_1, x_2, x_3, \ldots, x_n) \\
\end{align*}
\]

\[
\begin{align*}
(x_1, x_2, x_3, \ldots, x_n) \\
\end{align*}
\]
Tensor Flow

- Tensors have a **Shape** that is described with a vector

  \[ [1000, 256, 256, 3] \]

  - 10000 Images
  - Each Image has 256 Rows
  - Each Row has 256 Pixels
  - Each Pixel has 3 values (RGB)
Tensor Flow

- Tensors have a **Shape** that is described with a vector
- `[1000, 256, 256, 3]`
- 10000 Images
- Each Image has 256 Rows
- Each Row has 256 Pixels
- Each Pixel has 3 values (RGB)
Tensor Flow

Computation is a dataflow graph
Tensor Flow

Computation is a dataflow graph with tensors

Edges are N-dimensional arrays: *Tensors*
Computation is a dataflow graph with state
Core TensorFlow data structures and concepts

- **Graph**: A TensorFlow computation, represented as a dataflow graph:
  - collection of ops that may be executed together as a group.
Core TensorFlow data structures and concepts

- **Graph**: A TensorFlow computation, represented as a dataflow graph:
  - collection of ops that may be executed together as a group.
- **Operation**: a graph node that performs computation on tensors
Core TensorFlow data structures and concepts

- **Graph**: A TensorFlow computation, represented as a dataflow graph:
  - collection of ops that may be executed together as a group.
- **Operation**: a graph node that performs computation on tensors
- **Tensor**: a handle to one of the outputs of an Operation:
  - provides a means of computing the value in a TensorFlow Session.
Tensor Flow

- Constants

- Placeholders: must be fed with data on execution.

- Variables: a modifiable tensor that lives in TensorFlow’s graph of interacting operations.

- Session: encapsulates the environment in which Operation objects are executed, and Tensor objects are evaluated.
Tensor Flow

- **Constants**
- **Placeholders**: must be fed with data on execution.
Tensor Flow

- **Constants**
- **Placeholders**: must be fed with data on execution.
- **Variables**: a modifiable tensor that lives in TensorFlow’s graph of interacting operations.
Tensor Flow

- **Constants**
- **Placeholders**: must be fed with data on execution.
- **Variables**: a modifiable tensor that lives in TensorFlow’s graph of interacting operations.
- **Session**: encapsulates the environment in which Operation objects are executed, and Tensor objects are evaluated.
### Tensor Flow

<table>
<thead>
<tr>
<th>Category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element-wise math ops</td>
<td>Add, Sub, <strong>Mul</strong>, Div, Exp, Log, Greater, Less...</td>
</tr>
<tr>
<td>Matrix ops</td>
<td><strong>Concat</strong>, Slice, <strong>Split</strong>, Constant, Rank, <strong>Shape</strong>...</td>
</tr>
<tr>
<td>Matrix ops</td>
<td><strong>MatMul</strong>, MatrixInverse, MatrixDeterminant...</td>
</tr>
<tr>
<td>Stateful ops</td>
<td><strong>Variable</strong>, Assign, AssignAdd...</td>
</tr>
<tr>
<td>NN building blocks</td>
<td><strong>SoftMax</strong>, Sigmoid, <strong>ReLU</strong>, <strong>Convolution2D</strong>...</td>
</tr>
<tr>
<td>Checkpointing ops</td>
<td>Save, Restore</td>
</tr>
<tr>
<td>Queue &amp; synch ops</td>
<td>Enqueue, Dequeue, MutexAcquire...</td>
</tr>
<tr>
<td>Control flow ops</td>
<td>Merge, Switch, Enter, Leave...</td>
</tr>
</tbody>
</table>