Deep Neural Networks (DNN)
Deep learning

- representation learning
- unsupervised feature learning
Shallow machine learning

effective in solving many simple or well-constrained problems

- Gaussian mixture models (GMMs)
- conditional random fields (CRFs)
- maximum entropy (MaxEnt)
- support vector machines (SVMs)
- logistic regression
- multilayer perceptrons (MLPs)
Model

- **powerful - can solve hard problems**
  - any algorithm can be implemented by deep circuit
  - human neurons - 100Hz
  - brain is not exponentially large

- **trainable**
  - stochastic gradient descent (SGD) - it somehow works despite of non-convex
  - lot of data
What has changed?

- computers were slow -> small networks were too small
- datasets were small
- SGD can not work!

Embarrassingly, they did not succeed because they used the “small random weights” for the initialization, which works great for shallow nets but simply doesn’t work for deep nets at all. When the nets are deep, the many weight matrices all multiply each other, so the effect of a suboptimal scale is amplified.
Tricks

- preprocess data
  - mean is 0 and variance is 1
  - values in dimension should be linear, e.g. scale using log
- mini-batches for SGD
- learning rate (LR) - 0.1 is most universal, but rather smaller than larger
- use validation set to adjust LR
- weight initialization - 0.02 * randn (num_params) is most universal
Data augmentation

automatically create more data, e.g.:

- rotate images
- add noise to images or sound
Dropout

- turn off some neurons while training
- while testing change weights according to dropout probability
Pretraining

- unsupervised
- restricted Boltzmann machines (RBMs)
- autoencoders
- training layer by layer
Autoencoders

output

hidden

input

encode

decode
Activation function
Applications

- sound
- image
- text - the smallest improvement to state-of-the-art
- multimodal
Speech recognition

- first successful application of DNN
- major products: Microsoft Cortana, Xbox, Skype Translator, Google Now, Apple Siri, Baidu, Nuance
- PER reduced from 21.7 (GMM-HMM) to 17.9
• error rate reduced from 0.56 (SVM) to 0.23
Image recognition - faces
Image recognition

faces

cars

elephants
NLP - language model

Predict the next character in a stream of text using plenty of context

- Take Wikipedia, NYT, papers
- Arrange its letters in a sequence
- Train RNN to predict the next character from its past
  - Train on string fragments of 100 characters
  - 5 days on 8 high-end GPUs with 4GB of RAM each
**Recurrent** network with the Stiefel information for logistic regression methods. Along with either of the algorithms previously (two or more skewprecision) is more similar to the model with the same average mismatched graph. Though this task is to be studied under the reward transform, such as (c) and (C) from the training set, based on target activities for articles a \( a \geq 2(6) \) and (4.3). The PHDPic (PDB) matrix of cav’va using the three relevant information contains for timeing measurements. Moreover, because of the therap tor, the aim is to improve the score to the best patch randomly, but for each initially four data sets. As shown in Figure 11, it is more than 100 steps, we used \( \text{?? \to \infty with } 1000 \)
NLP - word embeddings

- words are mapped to vectors for use in NN
NLP - word2vec

(Mikolov et al., NAACL HLT, 2013)
NLP - dependency parsing

Stanford parser 3.5.0 (October 2014)

<table>
<thead>
<tr>
<th>Parser</th>
<th>Dev</th>
<th>Test</th>
<th>Speed (sent/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UAS</td>
<td>UAS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LAS</td>
<td>LAS</td>
<td></td>
</tr>
<tr>
<td>standard</td>
<td>89.9</td>
<td>89.7</td>
<td>51</td>
</tr>
<tr>
<td>eager</td>
<td>90.3</td>
<td>89.9</td>
<td>63</td>
</tr>
<tr>
<td>Malt:sp</td>
<td>90.0</td>
<td>89.9</td>
<td>560</td>
</tr>
<tr>
<td>Malt:eager</td>
<td>90.1</td>
<td>90.1</td>
<td>535</td>
</tr>
<tr>
<td>MSTParser</td>
<td>92.1</td>
<td>92.0</td>
<td>12</td>
</tr>
<tr>
<td>Our parser</td>
<td>92.2</td>
<td>92.0</td>
<td>1013</td>
</tr>
</tbody>
</table>
A person riding a motorcycle on a dirt road.

Two dogs play in the grass.

A skateboarder does a trick on a ramp.

A group of young people playing a game of frisbee.

Two hockey players are fighting over the puck.

A little girl in a pink hat is blowing bubbles.
Figure 1. Our model generates free-form natural language descriptions of image regions.
References

- D. Chen and C. Manning, *A Fast and Accurate Dependency Parser using Neural Networks*
- I. Suskever, J. Martens i G.E. Hinton, *Generating Text with Recurrent Neural Networks*
- J. Martens, *Deep learning via Hessian-free optimization*