NLP
Deep Learning

Neural QA - QANTA
Links about Theano and Qanta

• [http://deeplearning.net/software/theano/tutorial/](http://deeplearning.net/software/theano/tutorial/)
  – Tutorial about Theano

• [https://cs.umd.edu/~miyyer/qblearn/](https://cs.umd.edu/~miyyer/qblearn/)
  – A Neural Network for Factoid Question Answering over Paragraphs
  – Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III
  – EMNLP 2014
  – [https://cs.umd.edu/~miyyer/data/question_data.tar.gz](https://cs.umd.edu/~miyyer/data/question_data.tar.gz)
  – [https://cs.umd.edu/~miyyer/qblearn/qanta.tar.gz](https://cs.umd.edu/~miyyer/qblearn/qanta.tar.gz)
Dependency-Tree
Recursive Neural Networks

• Unlike constituent parses, words reside at nodes other than leaves
• Need to combine word vector of node with “hidden” vectors of children

[slides from Chris Hidey]
Dependency-Tree
Recursive Neural Networks

\[ h_{helots} = f(W_v \cdot x_{helots} + b) \]

\[ h_{called} = f(W_v \cdot x_{called} + b + W_{DOBJ} \cdot h_{helots}) \]
Dependency-Tree
Recursive Neural Networks

$h_n = f(W_v \cdot x_n + b + \sum_{k \in K(n)} W_{R(n,k)} \cdot h_k)$
Cost Function

\[ C(S, \theta) = \sum_{s \in S} \sum_{z \in Z_t} \max(0, 1 - x_c \cdot h_s + x_z \cdot h_s) \]

\[ x_c = \text{correct} \]
\[ x_z = \text{incorrect} \]

\[ \min_\theta \frac{1}{N} \sum_{t \in T} C(t, \theta) \]

\[ \theta = \{ W_{r \in R}, W_v, W_e, b \} \]
Activation Functions

- Sigmoid
  \[ \frac{1}{1 + e^{-x}} \]

- Tanh (hyperbolic tangent)
  \[ \frac{1 - e^{-2x}}{1 + e^{-2x}} \]

- Rectified linear unit (ReLU)
  \[ \max(0, x) \]

- Normalized Tanh
  \[ f(x) = \frac{\tanh(x)}{\|\tanh(x)\|} \]
Assignment

• Quiz bowl questions and answers
• Dependency parsed and named entities identified
  – othello
  – john_adams
  – battle_of_midway
Assignment

["he", "nsubj", 2],
["began", "root", 0],
["his", "poss", 5],
["congressional", "amod", 5],
["career", "dobj", 2],
["as", "prep", 2],
["a", "det", 8],
["warhawk", "pobj", 6],
[null, null, null],
["urging", "partmod", 2],
["president", "nn", 12],
["james_madison", "dobj", 10],
["towards", "prep", 12],
["war", "pobj", 13]
....
], "henry_clay"]
Grading (1/2)

1. Logistic regression – 1 pt

2. Format data – 5 pts

3. Activation function – 2 pts

\[ f(x) = \frac{\tanh(x)}{\|\tanh(x)\|} \]
Grading (2/2)

4. Recurrence relation and cost function – 5 pts

\[ h_n = f(W_v \cdot x_n + b + \sum_{k \in K(n)} W_{R(n,k)} \cdot h_k) \]

5. Experiments with word2vec – 5 pts

6. Visualization of answers – 1 pt

7. Add 3 features to classifier – 5 pts
NLP