Neural networks

Natural language processing - recursive network training
**Topics:** recursive neural network (RNN)

- Idea: recursively merge pairs of word/phrase representations

- We need 2 things
  - a model that merges pairs of representations
  - a model that determines the tree structure

---

**Figure 1.** Illustration of our recursive neural network architecture which parses images and natural language sentences. Segment features and word indices (orange) are first mapped into semantic feature space (blue) and then recursively merged by the same neural network until they represent the entire image or sentence. Both mappings and mergings are learned.

---

Socher, Lin, Ng and Manning, 2011
RECURSIVE NEURAL NETWORK

Topics: training algorithm

• Let \( y \) be the true parse tree and \( \hat{y} \) be the predicted parse tree
  
  ‣ we would like the score \( s(y) \) of \( y \) to be higher than the score \( s(\hat{y}) \) of \( \hat{y} \) (unless \( \hat{y} \) is actually \( y \))

• To update the recursive network
  
  ‣ infer the predicted parse tree \( \hat{y} \)
  
  ‣ increase the score \( s(y) \) and decrease the score \( s(\hat{y}) \) by doing an update in the direction of the gradient
    
    \[
    \nabla_\theta s(y) - \nabla_\theta s(\hat{y})
    \]

- these gradient can be computed by backpropagating through the recursive network structured according to the parse trees \( y \) and \( \hat{y} \)
RECURSIVE NEURAL NETWORK

**Topics:** training algorithm

- The nodes of a parse tree are also labeled
  - noun phrase (NP), verb phrase (VP), etc.
  - can add softmax layer that predict the label from each node representation
  - this is an additional gradient to backpropagate, for the true parse tree $y$
**Topics:** training algorithm

- The nodes of a parse tree are also labeled
  - noun phrase (NP), verb phrase (VP), etc.
  - can add softmax layer that predict the label from each node representation
  - this is an additional gradient to backpropagate, for the true parse tree $y$
**Topics:** training algorithm

- Other details
  - word representations are pre-trained using Collobert and Weston’s approach and fine-tuned while training the recursive network
  - training is actually based on a margin criteria: $s(y) > s(y^*) + \Delta(y, y^*)$
    - score of the true parse tree $y$ trained to be larger than score of any other tree $y^*$ plus its number of incorrect spans $\Delta(y, y^*)$
    - a simple modification to the beam search finding the best tree (see Socher et al. for details)
Topics: experimental comparison

• Parsing F1 performance
  - recursive neural network: 90.29%
  - Berkeley parser: 91.63%

• Nearest neighbor phrases based on RNN representation

Fujisawa gained 50 to UNK
1. Mead gained 1 to 37 UNK
2. Ogden gained 1 UNK to 32
3. Kellogg surged 4 UNK to 7

The dollar dropped
1. The dollar retreated
2. The dollar gained
3. Bond prices rallied

Socher, Lin, Ng and Manning, 2011