Learning hypernymy of distributed word vectors via a stacked LSTM network

Irving Rodriguez

Stanford University

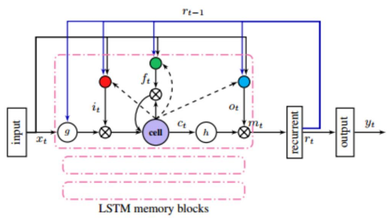
Hypernymy and Word Vectors

- Distributed word vectors learn semantic information between words with similar contexts.
- **Hypothesis: Hypernymy (and other semantic relationships) are distributed across the dimensions of the learned vectors.**

Example of hypernymy and its asymmetry and transitivity.

LSTM Network Architecture

- **Goal:** Learn hyponym-hypernym vector mapping using a stacked LSTM network.



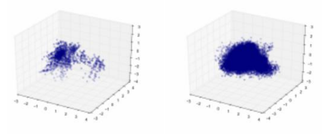
$$i_t = \sigma(W_i h_{t-1} + W_i w_t + b_i)$$
$$f_t = \sigma(W_f h_{t-1} + W_f w_t + b_f)$$
$$o_t = \sigma(W_o h_{t-1} + W_o w_t + b_o)$$
$$c_t = tanh(W_c h_{t-1} + W_i w_t + b_c)$$
$$C_t = f_t \times C_{t-1} + (1 - f_t) \times c_t$$
$$h_t = o_t \times tanh(C_t)$$

Right: Update rules for LSTM cell. h denotes the predicted hypernym, w the input hyponym, and C the cell state.
Left: Visualization of LSTM cell with input, forget, output, and activation (c) gates.
***Source: Sak et. al,** I"Long Short-Term Recurrent Neural Network Architectures for Large Scale Acoustic Modeling"*

- Hypernymy may be distributed in complex, non-uniform ways.
- As such, use LSTM cells with unified input-forget ("replacement") gate to update weights differently for different hypernym "types"
  - ➢ apple – fruit vs. apple – food vs. apple - company

Data

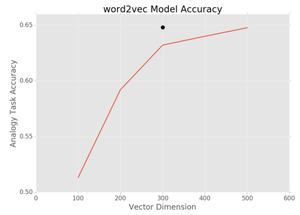
- Datasets used in literature (number of hyponym-hypernym pairs):
  - BLESS (1.4k), Linked Hypernym Datasets (3.7M)
  - Pair examples: (chris_cristie, politician), (duathlon, event)
- Previous models use BLESS to cluster vector differences, then learn linear projection for each cluster (piecewise-projection).

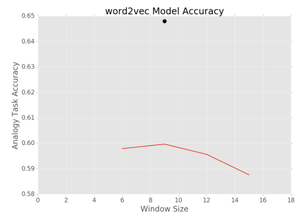


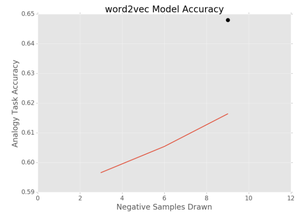
Top 3 principal components of the vector difference between pairs in each dataset.
Left: BLESS. Right: LHD.

Training and Hyperparameter Tuning

1) Train word vectors on English July 2015 Wikipedia dump (~5M articles, ~1.4B tokens) with word2vec module







Final Word Vector Model Parameters

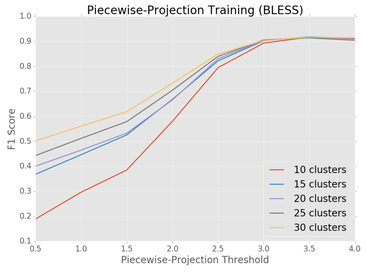
Vocabulary Size: 250k

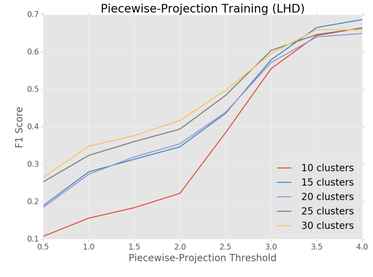
Dimension	Context Window Size	Minimum Token Appearance	Negative Samples
300	9	50	9

2) Train piecewise-projection classifier on BLESS and LHD sets, use as baseline accuracy for LSTM network.

- Classify word pairs whose projected difference norm is under some threshold *(Fu et. al):*

$$\| \Phi_k x - y \|^2 < \delta$$





Final Piecewise-Projection Model Parameters

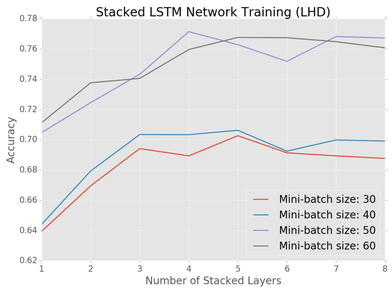
Number of clusters, BLESS	Threshold, BLESS	Number of clusters, LHD	Threshold, LHD
30	3.5	15	4.0

3) Train stacked LSTM model to learn mapping from hyponym vector to hypernym vector.

- Minimize quadratic loss between predicted and label hypernym:

$$J = \frac{1}{2} \Sigma_{i=1}^{m} ||h_i - w_{e_i}||_2^2$$

Stacked LSTM Results

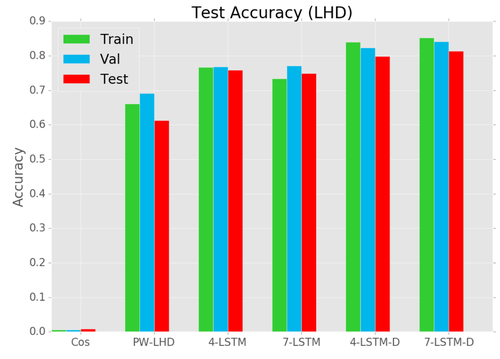


Final Stacked LSTM Model Parameters

Mini-batch SGD Size	Number of LSTM Layers
50	4

4) Add dropout to the top layer of the two best-performing LSTM models.

Model	Training Accuracy (%)	Validation Accuracy (%)	Test Accuracy (%)
PW-PROJ-LHD	66.1	69.1	61.3
4-LSTM	76.7	76.8	75.9
7-LSTM	73.4	77.1	75.0
4-LSTM + D	84.0	82.4	79.8
7-LSTM + D	**85.2**	**84.1**	**81.3**



References

- Mikolov et al. "Distributed Representations of Words and Phrases and their Compositionality". http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf
- Baroni et. al. "How we BLESSed distributional semantic evaluation." http://dl.acm.org/citation.cfm?id=2140491
- Kliegr. "Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery
- Fu et. al. "Learning Semantic Hierarchies via Word Embeddings." http://ir.hit.edu.cn/~jguo/papers/acl2014-hypernym.pdf
- Google. "word2vec, tool for computing continuous distributed representations of words." https://code.google.com/p/word2vec/
- Abadi et. al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems." http://tensorflow.org/