Unsupervised Recovery of Syntactic Trees from Learned Language Representations

Benjamin Newman

blnewman@stanford.edu

Stanford
CS Department

Intro

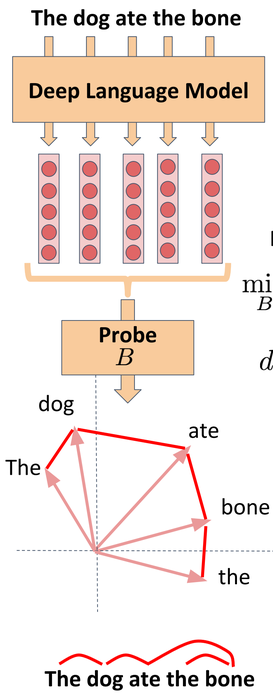
- Current deep language models like ELMo, BERT, XLNet appear to learn syntactic information with explicit syntactic signal.
- Most searches for syntax train supervised "probing" models on top of deep representations, but these incorporate external syntactic information.
- We seek an *unsupervised* probe for syntactic structure to see what structure is contained within/between the representations themselves.

Question

- Can we predict syntactic structure of sentences by looking only at the differences between vector representations for the sentence's words?

Background

- Structural Probe (Hewitt and Manning, 2019):



Start with a sentence

$$[w_1, w_2, w_3, w_4, w_5]$$

Encode it using ELMo to get model representations

$$[h_1, h_2, h_3, h_4, h_5]$$

Find the B (using SGD) such that:

$$\min_B \sum_{i,j} \left| d_B(h_i, h_j) - d(w_i, w_j) \right|$$

where

$$d_B(h_i, h_j) = ||Bh_i - Bh_j||_2^2$$

This create a distance matrix:

	The	*dog*	*ate*	*the*	*bone*
The	0.0	0.9	2.2	5.1	2.9
dog	0.9	0.0	1.2	3.4	1.8
ate	2.2	1.2	0.0	2.0	0.8
the	5.1	3.4	2.0	0.0	0.7
bone	2.9	1.8	0.8	0.7	0.0

Finally, find the tree that best fits the distance matrix

Experiment 1: Tree from Distance Matrix

Data

- 125 distance matrices, *D*, computed from trees, *T*, on 5 elements
- We right *D* as $D = \sum_i^n \lambda_i T_i$ for three values of *n*
  - $n = 1, \lambda = 1$
  - $n \in \{2, 5\}, \lambda_i \in [0, 1]$

Models

- Decoding methods - convert *D* to *T*.
  - Find the Minimum Spanning Tree of Metric (MST)
  - Solve Local Integer Linear Program (ILP)

$$\min_x \sum_{i,j} x_{ij}(d_B(h_i, h_j) - 1)$$

s.t. edge indicators $x_{i,j} \in \{0, 1\}$transitivity, tree flow

  - Agglomerative Hierarchical Clustering (AGG)

Methods

- Question: How do we find the tree distance matrix, *T*, that best fits a given distance matrix, *D*?
- We evaluate fit in two ways:
  - Weighted Edge Accuracy: $\frac{\sum_i \lambda_i |\text{gold}_i \cap \text{predicted}|}{n \sum_i \lambda_i}$
  - Frobenius Norm: $||D - T||_F$

Results

Weighted Edge Accuracy

	single tree	two trees	five trees
MST	1.000	0.400	0.124
ILP	1.000	0.324	0.124
AGG	-	-	-

Average Frobenius Norm

	single tree	two trees	five trees
MST	0.000	3.695	12.666
ILP	0.000	3.914	12.666
AGG	9.673	9.108	7.116

Experiment 2: Unsupervised Probe

Data

- Penn Treebank - news text and parses (Marcus, 1999)
  - Training set: ~40,000 sentences
  - Development set: ~1,700 sentences
  - Test Set: ~2,400 sentences
- ELMo layer 1 representations (Peters et al., 2018)

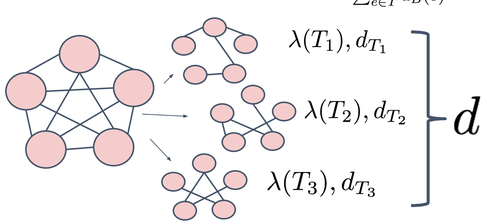
Objectives

Minimum Spanning Tree:

$$\sum_{i,j} \left| d_B(h_i, h_j) - d_{MST(d_B)}(w_i, w_j) \right|$$

Sampling Random Walks:

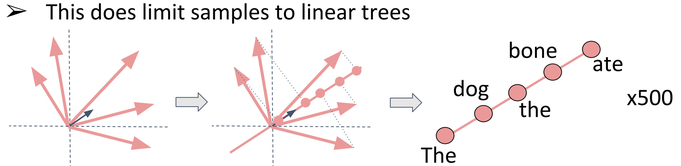
$$\sum_{i,j} \left| d_B(h_i, h_j) - \sum_{T \sim RW(d_b)} \lambda(T) d_T(w_i, w_j) \right| \qquad \lambda(T) \propto \frac{1}{\sum_{e \in T} d_B(e)}$$



Methods

- Question: Where to get distances, *d*, in optimization objective in the unsupervised setting?
- Supervised:
  - Gold Labels (Hewitt and Manning, 2019)
- Unsupervised: compute labels from d_B
  - Minimum Spanning Tree (Prim's algorithm)
  - Sampling Random Walks
  - Sampling Trees (FRT)
- Use these unsupervised "labels" to compute losses the same way you would for EM.
- Evaluation Metric:
  - Edge Accuracy (compared to gold syntax labels)

- Naïvely, this is **slow**, so approximate random walks by using random projections
  - Project mean-centered probe outputs onto random unit vectors 500x
  - Sort projections to get a random walk
  - Probe outputs that are close will more often be close after random projections as well
  - This does limit samples to linear trees

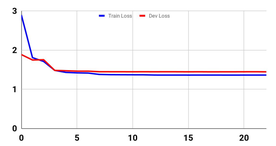


Results

Probe	Average Edge Accuracy
Untrained Probe	31%
Unsupervised MST	39%
Unsupervised Random Walk	24%
Supervised Probe	77%

MST Loss Curve

Random Walk Loss Curve



Discussion/Future Work

- Investigate the failure more closely, it looks like the probe learned something, but what?
- Get this probe to work with Fakcharoenphol, Rao, Talwar 2004's O(log(n))-distortion tree-sampling procedure.
- Speed up sampling, it's quite slow.
- Extend sampling procedure to decompose distance metrics into multiple trees.

Conclusion

- When decoding a tree from a distance matrix, using the minimum spanning tree performs is truer to the distance matrix than a local ILP (and it also is a lot faster).
- Training a probe with this EM-like procedure is a challenge and doesn't seem to find syntactic information in the models.

References

- Fakcharoenphol, Jittat, Satish Rao, and Kunal Talwar. "A tight bound on approximating arbitrary metrics by tree metrics." *Journal of Computer and System Sciences* 69.3 (2004): 485-497.
- Hewitt, John, and Christopher D. Manning. "A structural probe for finding syntax in word representations." *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019.
- Marcus, Mitchell, et al. Treebank-3 LDC99T42. Web Download. Philadelphia: Linguistic Data Consortium (1999).
- Peters, Matthew E., et al. "Deep contextualized word representations." *arXiv preprint arXiv:1802.05365* (2018).