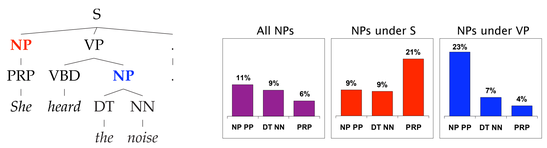
Generative and Discriminative Latent Variable Grammars

Slav Petrov
Google Research, New York

Motivation

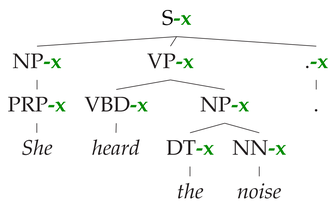
Grammar Learning

The observed treebank categories are too coarse because the rewrite probabilities depend on context.



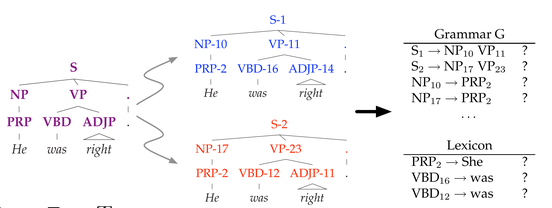
Grammars with Latent Variables

Given a treebank over a set of categories learn an optimally refined grammar for parsing.



Automatic Grammar Refinement

Refine the observed trees with latent variables and learn subcategories.



Parse Tree T Sentence w | Derivations $t_i : T$ | Model Parameters θ

$$\mathrm{P}_\theta(t|w) = \frac{1}{Z(\theta, w)} \prod_{X \rightarrow \gamma \in t} e^{\theta_{X \rightarrow \gamma}} = \frac{1}{Z(\theta, w)} e^{\theta^\top f(t)}$$

Training

Generative Parameter Estimation

Maximize the joint likelihood:

$$\mathcal{L}_{joint}(\theta) = \log \prod_i \mathrm{P}_\theta(T_i, w_i) = \log \prod_i \sum_{t:T_i} \mathrm{P}_\theta(t, w_i)$$

$$\theta^* = \operatorname*{argmax}_\theta \left(\log \prod_i \sum_{t:T_i} \mathrm{P}_\theta(t, w_i) \right)$$

The parameters can be learned with an Expectation Maximization algorithm. The E-Step involves computing expectations over derivations corresponding to the observed trees. These expectations are normalized in the M-Step to update the rewrite probabilities:

$$\phi_{X \rightarrow \gamma} = \frac{\sum_T \mathbb{E}_\theta[f_{X \rightarrow \gamma}(t)|T]}{\sum_{\gamma'} \sum_T \mathbb{E}_\theta[f_{X \rightarrow \gamma'}(t)|T]}$$

Computing expectations over derivations corresponding to the observed trees can be done in linear time (in the number of words).

Discriminative Parameter Estimation

Maximize the conditional likelihood:

$$\mathcal{L}_{cond}(\theta) = \log \prod_i \mathrm{P}_\theta(T_i|w_i) = \log \prod_i \sum_{t:T_i} \mathrm{P}_\theta(t|w_i)$$

$$\theta^* = \operatorname*{argmax}_\theta \left(\log \prod_i \sum_{t:T_i} \mathrm{P}_\theta(t|w_i) \right)$$

The parameters can be learned with a numerical gradient based method (e.g. L-BFGS). Computing the gradient involves calculating expectations over derivations corresponding to the observed trees, as well as over all possible trees:

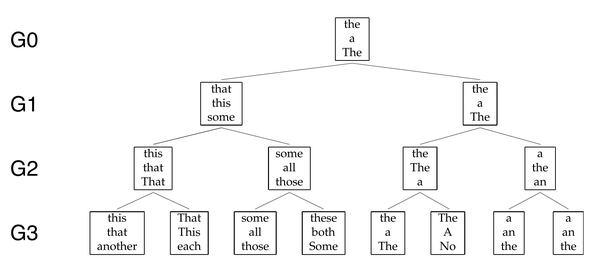
$$\frac{\partial \mathcal{L}_{cond}(\theta)}{\partial \theta_{X \rightarrow \gamma}} = \sum_i \left(\mathbb{E}_\theta \left[f_{X \rightarrow \gamma}(t)|T_i \right] - \mathbb{E}_\theta[f_{X \rightarrow \gamma}(t)|w_i] \right)$$

Computing expectations over derivations corresponding to all possible trees involves parsing the training corpus, which requires cubic time (in the number of words).

Controlling Complexity

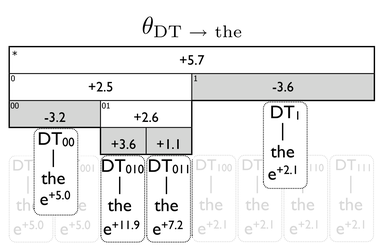
Split & Merge Grammars

Use split & merge heuristic with likelihood ratio criterion. Explicitly model number of subcategories. Split each category in two, and merge back the least useful half of the splits.



Multi-Scale Grammars

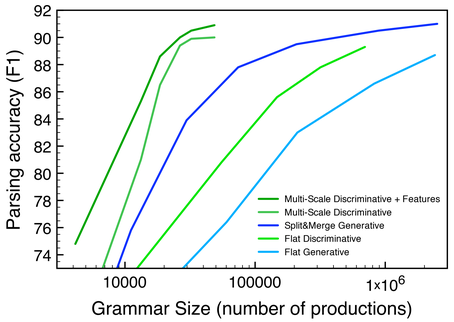
Use hierarchical features and L1 regularization. Allow each production to reference categories at different levels of granularity.



Results

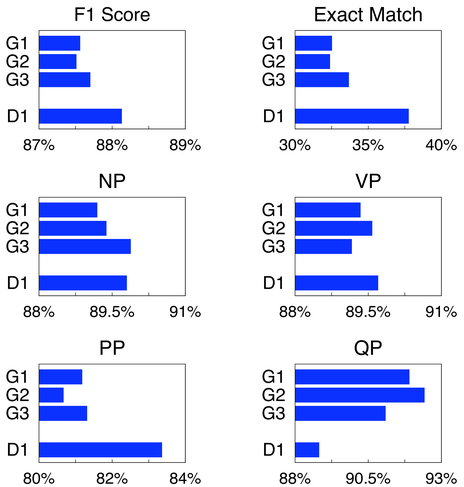
Overall Parsing Accuracy

Grammars were trained on the Wall Street Journal section of the Penn Treebank (1M words in 40K sentences).



Detailed Breakdown

There are significant differences in the errors that generative and discriminative models make.



Note also the large variance among the different generative grammars.

Google Research