Combining the Sparsity and Unambiguity Biases for Grammar Induction

Kewei Tu

Departments of Statistics and Computer Science, University of California, Los Angeles

Overview

We describe our participating system for the dependency induction track of the PASCAL Challenge on Grammar Induction. Our system incorporates two types of inductive biases:

- The *sparsity* bias, which favors a grammar with fewer grammar rules.
- The *unambiguity* bias, which favors a grammar that leads to unambiguous parses, motivated by the observation that natural language is remarkably unambiguous in the sense that the number of plausible parses of a natural language sentence is very small.

Sparsity Bias

We employ two different approaches to inducing sparsity.

Dirichlet Prior (Dir)

A Dirichlet distribution can be used as the prior of the grammar rule probabilities θ

$$P(\theta;\alpha_1,\ldots,\alpha_K)=\frac{1}{B(\alpha)}\prod_{i=1}^{K}\theta_i^{\alpha_i-1}$$

where $\alpha=(\alpha_1,\ldots,\alpha_K)$ are the hyperparameters, and $B(\alpha)$ is the normalization constant. If the values of the hyperparameters are less than 1, then the Dirichlet prior assigns larger probabilities to vectors that have more elements close to zero, therefore encouraging parameter sparsity.

Sparsity-inducing Posterior Regularization (SPR)

Gillenwater et al. (2010) proposed to induce sparsity in dependency grammars by adding a regularization term to the posterior of the grammar that penalizes the number of unique dependency types in the parses of the training data. Their objective function is:

$$J(\theta) = \log p(\theta|\mathbf{X}) - \min_q \left(\mathbf{KL}(q(\mathbf{Z})||p_\theta(\mathbf{Z}|\mathbf{X})) + \sigma_s \sum_{cp}\max_i \mathbf{E}_q[\phi_{cpi}(\mathbf{X},\mathbf{Z})]\right)$$

where θ is the parameter of the grammar, $\mathbf{X}$ is the training data, $\mathbf{Z}$ is the dependency parses of the training data $\mathbf{X}$, σ_s is a constant that controls the strength of the regularization term, c and p range over all the tags of the dependency grammar, i ranges over all the occurrences of tag c in the training data $\mathbf{X}$, and $\phi_{cpi}(\mathbf{X},\mathbf{Z})$ is an indicator function of whether tag p is the dependency head of the i-th occurrence of tag c in the dependency parses $\mathbf{Z}$.

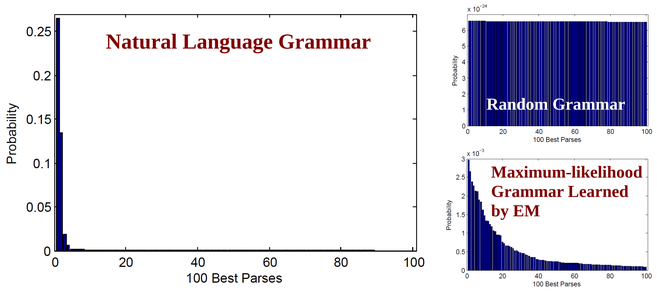
Gillenwater et al. (2010) optimize this objective function using a variant of the expectation-maximization algorithm (EM), which contains an E-step that optimizes the auxiliary distribution q using the projected subgradient method.

Unambiguity Bias

The unambiguity bias favors a grammar that leads to unambiguous parses on natural language sentences (Tu and Honavar, 2012).

Motivation

Natural language is remarkably unambiguous in the sense that the number of plausible parses of a natural language sentence is very small in comparison with the total number of possible parses.



On the right we show the probabilities of the parses of a typical natural language sentence. The natural language grammar (approximated by the Berkeley parser) is highly unambiguous compared with the random grammar and the maximum-likelihood grammar.

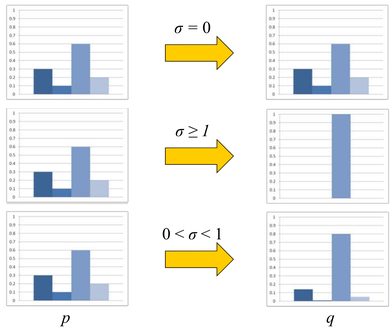
Unambiguity Regularization (UR)

We add into the objective function a regularization term that penalizes the entropy of the parses given the training sentences, based on the posterior regularization framework (Ganchev et al., 2010).

$$J(\theta) = \log p(\theta|\mathbf{X}) - \min_q \left(\mathbf{KL}(q(\mathbf{Z})||p_\theta(\mathbf{Z}|\mathbf{X})) + \sigma \sum_i H(z_i)\right)$$

- Log posterior of the grammar given the training sentences
- An auxiliary distribution
- KL-divergence between q and the posterior distribution of the parses
- A constant that controls the strength of regularization
- Entropy of the parses based on q

This objective function can be optimized using coordinate ascent. It can be shown that the behavior of the algorithm is controlled by the value of the parameter σ.



- When $\sigma = 0$, our approach reduces to standard EM: $q^*(z_i) = p_\theta(z_i|x_i)$ in the E-step.
- When $\sigma \geq 1$, our approach reduces to Viterbi EM:
$$q^*(z_i) = \begin{cases} 1 & \text{if } z_i = \arg\max_{z_i} p_\theta(z_i|x_i) \\ 0 & \text{otherwise} \end{cases}$$
- When $0 < \sigma < 1$, in the E-step we have: $q^*(z_i) = \alpha_i p_\theta(z_i|x_i)^{\frac{1}{1-\sigma}}$, which is a softmax function of p_θ, where α_i is the normalization factor. To compute q, we can simply raise all the rule probabilities of the grammar to the power of $\frac{1}{1-\sigma}$ and then run the normal E-step of EM. We refer to the algorithm in this case as the *softmax-EM* algorithm.

The choice of the value of σ is important in unambiguity regularization. To avoid choosing a fixed value of σ, we can anneal its value: starting learning with a large value of σ (e.g., $\sigma = 1$) to strongly push the learner away from the highly ambiguous initial grammar; then gradually reducing the value of σ, possibly ending with $\sigma = 0$, to avoid inducing excessive unambiguity in the learned grammar.

Combining Sparsity and Unambiguity Biases

Dir + UR

To incorporate Dirichlet priors into unambiguity regularization, we derive a mean-field variational inference algorithm, which alternately optimizes $q(\theta)$ and $q(\mathbf{Z})$. In optimizing $q(\theta)$, we obtain a set of weights summarized from $q(\theta)$. The optimization of $q(\mathbf{Z})$ is similar to the E-step of unambiguity regularization: when $0 < \sigma < 1$, we raise all the weights to the power of $\frac{1}{1-\sigma}$ before running the normal step of computing $q(\mathbf{Z})$; and when $\sigma \geq 1$, we use the weights to find the best parse of each training sentence and assign probability 1 to it.

SPR + UR

We employ a simplistic approach that optimizes the sparsity and unambiguity regularization terms separately in the E-step. First we ignore the sparsity regularization term and optimize $q(\mathbf{Z})$ by the E-step of unambiguity regularization. The result is an intermediate distribution $q'(\mathbf{Z})$. Then we ignore the unambiguity regularization term and optimize $q(\mathbf{Z})$ to minimize the sparsity regularization term as well as the KL-divergence between $q(\mathbf{Z})$ and $q'(\mathbf{Z})$.

Implementation and Experiments

Our system was built on top of the grammar induction system from Gillenwater et al. (2010). We preprocessed the corpora to remove all the punctuations as denoted by the universal POS tags. We trained our system on the fine POS tags, except for the Dutch corpus (on which we used the coarse POS tags). Harmonic initialization is used as in previous work.

We tuned the following parameters by coordinate ascent on the development set: the maximal length of sentences used in training, the valence and back-off strength of the E-DMV model, the hyperparameter α of Dirichlet priors, the type and strength σ_s of the sparsity-inducing posterior regularization, and the strength σ of unambiguity regularization. Dir+UR and SPR+UR were each found to be the better approach for five of the ten corpora. The sparsity bias was found to be beneficial (i.e., $\alpha < 1$ if Dirichlet priors were used, or $\sigma_s > 0$ if sparsity-inducing posterior regularization was used) for six of the ten corpora. The unambiguity bias was found to be beneficial (i.e., $\sigma > 0$) for seven of the ten corpora. This implies the usefulness of both types of inductive biases in grammar induction.

References

Jennifer Gillenwater, Kuzman Ganchev, Joao Graca, Fernando Pereira, and Ben Taskar. 2010. Sparsity in dependency grammar induction. In ACL 2010.

Kuzman Ganchev, Joao Graca, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. Journal of Machine Learning Research.

Kewei Tu and Vasant Honavar. 2012. Unambiguity regularization for unsupervised learning of probabilistic grammars. In EMNLP-CoNLL 2012.