Contrasting objective functions for CYK chart decoding

Aaron Dunlop and Brian Roark
Center for Spoken Language Understanding
Oregon Health & Science University
Portland, OR
[aaron.dunlop,roarkbr]@gmail.com

1 Introduction

Context-free inference is a standard part of many NLP pipelines. Most approaches use a variant of the CYK dynamic programming algorithm to populate a chart structure with predicted nonterminals over each span. We can extract a parse tree from this chart in several ways. In this work, we compare two commonly-used decoding approaches (*Viterbi* and *max-rule*) with a minimum-bayes-risk (MBR) method which has not been widely used. We find that the latter approach is competitive with and in some cases superior to the standard decoding methods.

2 Inference Methods

Viterbi decoding, the simplest and most common decoding method, finds the most probable complete tree according to the grammar. *Max-rule* decoding, first presented in Petrov and Klein (2007), optimizes instead the number of expected correct *rules*. The argmax is performed over grammar rules. Petrov reports an improvement of approximately 1.5 points F-score over Viterbi decoding; our experiments showed a similar increase for latent-variable grammars, but a *decrease* for smaller grammars.

Goodman (1996) proposed a MBR decoding method which maximizes expected recall of labeled nodes. He demonstrated that this *max-recall* method is able to produce parse trees which recover more correct nodes than the Viterbi parse, even if the tree in its entirety is not permitted by the grammar. He demonstrated closely related metrics maximizing precision or balancing the two.

We consider a tree T as a set of labeled spans. Given the posterior probability of a labeled span, $\gamma(X)$ and λ to select the operating point, we can produce the desired $\hat{T}$ using the maximization:[1]

$$\hat{T} = \operatorname*{argmax}_{T \in \tau} \sum_{X \in T} (\gamma(X) - \lambda)$$

Most grammars of interest encode split nonterminal spaces. That is, many (or all) elements of the nonterminal set are annotated with additional information beyond the training corpus node labels. For instance, a parent-annotated grammar splits each nonterminal to encode the parents with which it occurs in the training data; latent-variable grammars Petrov et al. (2006) add latent annotations (e.g., *NP* might be split into *NP_0, NP_1, ... NP_47*. We can perform the MBR argmax over the split nonterminals or while summing over those split states. We refer to these approaches as MBR-Max and MBR-Sum and present results for each.

3 Results and Discussion

We parsed section 22 of the Penn Treebank with each of the methods described, using a variety of grammars.[2] Table 1 shows summary statistics of each grammar and the results of exhaustive inside-outside inference with the Markov-0, Markov-2, and Parent-annotated grammar, and a near-exhaustive beam search with the latent-variable grammars. We also include results for 2 pruning approaches — the Berkeley parser's multi-level coarse-to-fine (Petrov et al. (2006)) and beam-width prediction (Bodenstab et al. (2011)). We found that

[1] $\lambda = 0$, maximizes recall and $\lambda = 1$ precision. $\lambda = 0.5$ balances the two equally, which is closely related to maximizing F1. See Appendix A of Hollingshead and Roark (2007) for the full derivation of this criteria and Goodman (1996) for the maximization algorithm.

[2] Most of these grammars are described in more detail in Dunlop et al. (2010). The SM5 grammar not included in that discussion is a Berkeley latent-variable, similar to the SM6 grammar, but trained for 5 split-merge cycles instead of 6.

	Grammar	Nonterminals	Rules	Viterbi	Max-Rule	MBR-Max	MBR-Sum
1	Markov-0	99	56485	61.1	60.1	65.4 / 0.25	65.4 / 0.25
2	Markov-2	3092	65902	70.9	71.2	73.6 / 0.25	73.6 / 0.25
3	Parent-annotated	6965	77928	78.3	79.4	78.8 / 0.25	80.0 / 0.35
4	SM5	1121	4.1m	88.4	90.1	89.7 / 0.2	89.6 / 0.4
5	SM6	1134	4.3m	88.8	90.4	89.8 / 0.2	89.7 / 0.4
6	SM6 Beam-prediction	1134	4.3m	88.8	88.1	87.1 / 0.15	86.6 / 0.4
7	Berkeley Parser (SM6 CTF)	1134	4.3m	-	90.4	-	-

Table 1: Parsing accuracy on WSJ section 22, and the peak λ values for MBR methods.

the MBR methods perform very similarly to each other (although they achieve peak accuracy at different operating points). MBR decoding improves accuracy considerably over Viterbi search for all grammars, although not as much as max-rule on the largest latent-variable grammars.

Bodenstab et al.'s beam-width prediction (Bodenstab et al., 2011) pruning is intended to keep the Viterbi 1-best parse in the beam (and in fact, it achieves identical accuracy to exhaustive Viterbi inference), but it impacts *max-rule* and the MBR approaches negatively. We consider this a particularly interesting result, and plan to explore the effects of other pruning approaches as well.

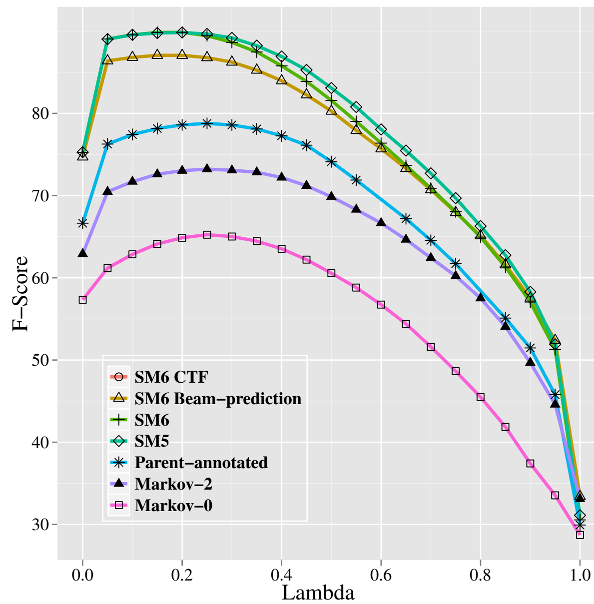

Figure 1: MBR-max parsing accuracy on WSJ section 22 at various operating points.

References

Nathan Bodenstab, Aaron Dunlop, Brian Roark, and Keith Hall. 2011. Beam-Width prediction for efficient Context-Free parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 440–449, Portland, Oregon, June.

Aaron Dunlop, Nathan Bodenstab, and Brian Roark. 2010. Reducing the grammar constant: an analysis of CYK parsing efficiency. Technical report CSLU-2010-02, OHSU.

Joshua Goodman. 1996. Parsing algorithms and metrics. *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, page 177183.

Joshua Goodman. 1998. *Parsing Inside-Out*. Ph.D. thesis, Harvard University, May.

Kristy Hollingshead and Brian Roark. 2007. Pipeline iteration. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, page 952959, Prague, Czech Republic, June. ACL.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York, April. ACL.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 433–440, Sydney, Australia. ACL.