Einführung in die Computerlinguistik

Data-Driven Parsing

Laura Kallmeyer
Heinrich-Heine-Universität Düsseldorf

Sommersemester 2013

Overview

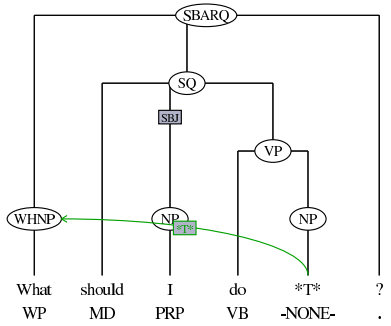
1. Treebanks
2. Grammar Extraction
3. Evaluation

Treebanks (1)

- Treebanks are corpora (i.e., collections of texts) where each sentence is annotated with a syntactic structure.
- The syntactic structure can be a constituency structure or a dependency structure.
- Constituency-based data driven parsing is usually done by learning a grammar (in most cases a PCFG) from a constituency treebank and using this grammar for parsing.
- Dependency-based data driven parsing is usually done by learning a dependency parser (e.g., a classifier) from the treebank.

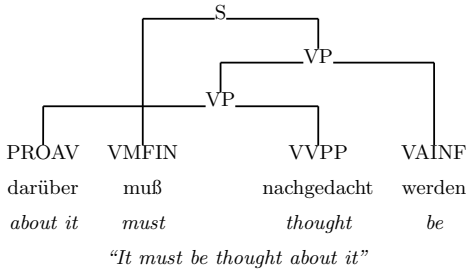
Treebanks (2)

Sample trees from the Penn Treebank (PTB):



Treebanks (3)

Sample tree from NeGra:



Grammar Extraction (1)

Having a treebank, in order to extract a latent PCFG,

- we first do some preprocessing (removal of traces, of crossing branches, ...).
- Then, we binarize the trees, i.e., we make sure all rhight-hand sides have length 2.
- For all $A \to \alpha \in P$, the estimated probability $p(A \to \alpha)$ is

$$p(A \to \alpha) = \frac{count(A \to \alpha)}{count(A)}$$

  where $count(A \to \alpha)$ is the number of occurrences of the production in the treebank and $count(A)$ the number of A-nodes in the treebank.

This is called a Maximum Likelihood Estimator.

Grammar Extraction (2)

Problem with such grammars: Independence assumptions are too strong.

Therefore, a series of techniques for grammar refinement have been proposed:

- Lexicalization of PCFGs [Collins, 2003]
- Markovization: Instead of using unique new non-terminals during binarization, we always use the same X, attaching some vertical and horizontal context to it [Klein and Manning, 2003]
- Category splitting and merging: whenever a single category A behaves differently in different context, we split it into several new categories, depending on context. This can be done automatically [Petrov et al., 2006]

Evaluation (1)

- In order to judge the performance of a parser, one must be able to assess the quality of its output (the parsed test data) with respect to the desired output (the gold data).
- The most widely used technique for this task consists of comparing for each parsed sentence the set of bracketings produced by the parser with the set of gold bracketings from the manual treebank annotation.
- A bracketing is a pair of indices on the input string denoting the start and the end of the span dominated by a certain non-terminal. The bracketing is called labeled if the label is included; if it is just the index pair, it is called unlabeled.

Evaluation (2)

Commonly, bracket scoring is defined as follows. Let $\mathbb{O}$ be the set of bracketings from the parser output, and let the set of bracketings from the treebank annotation be $\mathbb{G}$.

- Precision is then computed as $\frac{|\mathbb{O} \cap \mathbb{G}|}{|\mathbb{O}|}$,
- recall as $\frac{|\mathbb{O} \cap \mathbb{G}|}{|\mathbb{G}|}$, and
- F-score F_1 as $\frac{2 * precision * recall}{precision + recall}$.

Best F-score for English is 90.2, with the Penn Treebank:
[Petrov et al., 2006] with automatic category splitting and merging.

References

[Collins, 2003] Collins, M. (2003). Head-Driven Statistical Models for Natural Language Parsing. *Computational Linguistics*, 29(4):589–637.

[Klein and Manning, 2003] Klein, D. and Manning, C. D. (2003). Fast exact inference with a factored model for natural language parsing. In *In Advances in Neural Information Processing Systems 15 (NIPS)*. MIT Press.

[Nederhof, 2003] Nederhof, M.-J. (2003). Weighted Deductive Parsing and Knuth's Algorithm. *Computational Linguistics*, 29(1):135–143.

[Petrov et al., 2006] Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 433–440, Sydney.