

CONSTRAINT GRAMMAR AS A FRAMEWORK FOR PARSING RUNNING TEXT

Fred Karlsson

1. Outline

Grammars used in parsers are often directly imported from autonomous grammar theory and descriptive practice that were not exercised for the explicit purpose of parsing. Parsers have been designed for English based on e.g. GB, GPSG, and LFG. We present a formalism to be used for parsing where the grammar statements are closer to real text sentences and more directly address some notorious parsing problems, especially ambiguity.

The descriptive statements, constraints, don't have the ordinary task of defining the notion 'correct sentence in L'. They are less categorical in nature, closely tied to morphological features, and more directly geared towards the basic task of parsing: to infer structure from a stream of concrete tokens in a basically bottom-up mode.

Our tokens are morphologically analyzed word-forms. The central idea is to maximize the use of morphological information for the purpose of pruning ambiguities and determining syntactic structure. Another central idea is to have all relevant structure assigned directly via lexicon, morphology, and simple mappings from morphology to syntax. The task of the constraints is basically to discard as many alternatives as possible, the optimum being a fully disambiguated sentence with one syntactic reading only.

A good parsing formalism should satisfy at least the following requirements: the constraints should be declarative rather than procedural, they should be clearly separated from the program code, the formalism should be language-independent, and it should be reasonably easy to implement (optimally as finite-state automata).

2. Breaking up the problem of parsing

The problem of parsing running text may be broken up in five subproblems: preprocessing, morphological analysis, morphological disambiguation (consisting of local disambiguation and context-dependent disambiguation), determination of intrasentential clause boundaries, and assignment of surface-syntactic functions.

Real texts are full of idiosyncracies in regard to headings, paragraph structure, interpunctuation, use of upper and lower case, etc. Such phenomena must be properly normalized. The existence of an adequate preprocessor is here taken for granted. We concentrate on morphological analysis, clause boundary determination, disambiguation, and syntactic function assignment. Viewing the problem of parsing in turn from one or another of these angles clarifies many

intricacies. The subproblems take more manageable proportions and make possible a novel type of modularity.

Morphological analysis is relatively independent. Our syntactic parsers (treating ambiguities, clause boundaries, and syntactic functions) are always supplied with adequate morphological input. The morphological analyzers are designed according to Koskenniemi's (1983) two-level model. Currently our Research Unit has available morphological analyzers for English (37,000 lexicon entries), Finnish (34,000 entries), and Swedish (41,000 entries). Here are two morphologically analyzed English words ("a" has one reading, "move" four; reading lines start by the base-form; upper-case elements are morphological features except those with an initial "#" which are syntactic functions, also emanating from the lexicon; "#DN/" = modifier of the next N to the right, "#+FMAINV" = finite main verb, "#-FMAINV" = non-finite main verb as member of verb chain, "#NOM-FMAINV" = main verb as postmodifier of nominal):

```
a
a * DET CENTR ART INDEF #DN/"
move
move * N NOM SG "
move * V SUBJUNCTIVE #+FMAINV "
move * V IMP #+FMAINV "
move * V INF #-FMAINV #NOM-FMAINV "
```

By disambiguation is here especially meant reducing morphological ambiguities, optimally down to one. Sense disambiguation is not included (presently our lexical items have no sense descriptions). Selecting the proper syntactic tag when several are available (cf. the infinitive above) may in fact also be regarded as disambiguation.

The subproblems of disambiguation, clause boundaries, and syntax are interrelated. E.g., for optimal disambiguation it is useful to know the boundaries of the current clause, and to know as much as possible about its syntactic structure. An important aspect of the general problems is to work out the precise relations between these modules.

My paper (to be presented in Finnish) outlines the formalism and gives a demonstration of it as applied to English in collaboration with Aro Vuolilainen, Juha Heikkilä, and Arto Anttila.

KTP 17,
1990