

An Efficient, Primarily Bottom-Up Parser for Unification Grammars

Neil K. Simpkins and Peter J. Hancox
Applied Mathematics
Aston University
Aston Triangle
Birmingham, B4 7ET

Abstract

The search for efficient parsing strategies has a long history, dating back to at least the Cocke/Younger/Kusami parser of the early sixties. The publication of the Earley parser in 1970 has had a significant influence on context-free (CF) parsing for natural language processing, evidenced by the interest in the variety of chart parsers implemented since then. The development of unification grammars (with their complex feature structures) has put new life into the discussion of efficient parsing strategies, and there has been some debate on the use of essentially bottom-up or top-down strategies, the efficacy of top-down filtering and so on.

The approach to parsing described here is suitable for complex category, unification-based grammars. The concentration here is on a unification grammar which has a context-free backbone, Lexical-Functional Grammar (LFG). The parser is designed primarily for simplicity, efficiency and practical application.

The parser outlined here results in a high-level, but still efficient, language system without making a requirement on the grammar/lexicon writer to understand its implementation details. The parsing algorithm operates in a systematic bottom-up (BU) fashion, thus taking earliest advantage of LFG's concentration of information in the lexicon and also making use of unrestricted feature structures to realize LFG's Top-Down (TD) predictive potential. While LFG can make special use of its CF backbone, the algorithm employed is not restricted to grammars having a CF backbone and is equally suited to complex-feature-based formalisms.

Additionally, the algorithm described (which is a systematic left-to-right (left corner) parsing algorithm) allows us to take full advantage of both BU and TD aspects of a unification-based grammar without incurring prohibitive overheads such as feature-structure comparison or subsumption checking. The use of TD prediction, which in the Earley algorithm is allowed to hypothesize new parse paths, is here restricted to confirming initial parses produced BU, and specializing these according to future (feature) expectations.

Such a system can serve as the basis of a practical, linguistically-biased system using LFG and is a prerequisite for an effective set of grammar writing and debugging tools which operate at the level of LFG itself. The parsing algorithm described here is intended for use in an automatic speech understanding (ASR) system project (currently funded by the Royal Signals and Radar Establishment), where a data driven strategy coupled with a strong TD predictive capability is highly desirable.