Comments on Bertsch and Nederhof

Aravind K. Joshi
Department of Computer and Information Science
Room 555 Moore School
University of Pennsylvania
Philadelphia, Pa 19104 USA
joshi@linc.cis.upenn.edu

I will summarize below my main points which will be the basis of my oral comments at the workshop.

1. The authors have introduced a truly novel idea. They have considered the regular closure (i.e., closure under concatenation, union and Kleene star) of deterministic context-free languages. This closure includes many inherently ambiguous languages, in particular, the well-known language L which is a union of two deterministic languages L1 and L2. L1 contains strings of a's followed by strings of b's followed by strings of c's, with the requirement that the number of b's equals the number of c's. L2 is similar to L1 except that the requirement is that the number of a's equals the number of b's.

The main idea (and a very significant one) of the authors is to characterize this closure by a two-level parser. The first level is a finite-state automaton whose edges are labeled with nonterminals associated with languages at the second level. The key result is that the linear time recognition and parsing result for deterministic languages extends to the regular closure of these languages.

As the authors correctly claim that this result is significant because this closure includes some inherently ambiguous languages, as noted above. Since many constructions in natural languages are 'inherently' ambiguous, this result acquires added significance.

2. So now the question is whether the inherent ambiguities that seem to appear in natural languages are included in this closure. It is not clear to me (at least right now) how one would answer this question. Let me make a beginning.

The formal language example of an inherently ambiguous language given above (and similar examples in the literature) have the property that inherent ambiguity arises because of a 'counting' argument. The two different analyses a string correspond to two different ways of arriving at the same 'count'. I am not aware of a formal language example of an inherently ambiguous language which does not depend in some way on a counting argument.

Inherent ambiguities in natural languages seem to arise due to alternative ways of structuring a string, which does not depend on counting symbols. Thus for any reasonable grammar we can think of the string ' We enjoy visiting relatives' will need to have two parses. Similarly, for ' I saw the man in the park with a telescope' will need to have at least two parses. By the very nature of the enterprise, natural language grammars that we write are underspecifications and therefore for almost all sentences we have more than one parse and these ambiguities do not depend on some counting consideration.

It would be nice if we could formalize this notion of inherent ambiguity (not involving a counting argument). Perhaps it is obvious to others how to do this. I do not see it right now. If we could formalize this notion then we could raise the question whether this kind of inherent ambiguity, which is more relevant to natural languages, is included in the regular closure of deterministic language. Perhaps the authors already know the answer!

As I have said before, this paper makes a truly novel contribution by extending the class of linear time recognizable context-free languages. I have raised some questions that need to be addressed in the context of applications to natural language parsing.

Proceedings of the ECAI 96 Workshop
Extended Finite State Models of Language
Edited by A. Kornai.