Session 1: Spoken Language Systems I

Wayne Ward, Chair
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pa 15213

The papers in this session addressed issues in combining speech recognition with natural language systems. The first three papers concern the use of grammars. Speech recognizers and Natural Language parsers make different requirements of language knowledge. Recognizers need efficient methods for constraining the search space, while parsers need detailed analytical knowledge. One solution to the problem of integrating speech recognizers with NL processors is to use different language constraints in the two modules. This in effect means using different grammars for recognizing and parsing. The recognizer may use no grammar or simple, efficient grammars, while the parser uses a more complete representation of the language. This means that the recognizer can overgenerate, or produce strings not acceptable to the parser. In this case, a recognition error can lead to a failure to parse the utterance. One solution to this problem is to use an N-Best recognizer. Such a recognizer produces the N (where N is preset) best scoring hypotheses for an utterance. These hypotheses are passed to the parser which can then pick the overall best one.

Rich Schwartz from BBN and Frank Soong from AT&T both presented efficient algorithms for generating the N best recognition strings for an utterance. In contrast to previous N-Best algorithms, both of these algorithms require only a small amount of additional computation to produce N-Best instead of the single best hypothesis. Both systems use a time-synchronous forward search to find the best hypothesis, and a backward pass to generate the N best. Information accumulated in the forward pass is used to score paths in the backward search. The BBN algorithm uses a beam search on the backward pass where the AT&T system uses a tree search.

In order to be implemented efficiently, recognition grammars often overgenerate, but they should not also undergenerate. That is, they should not reject strings acceptable to the parser. Finite-state approximations of phrase structure grammars have been described in the past. Typically the FSA is generated by limiting rule expansions to a preset depth. This method has the disadvantage that the FSA generated is not a strict superset of the language generated by the PSG. Some strings are rejected by the FSA that are acceptable to the PSG. Fernando Pereira of AT&T presented an algorithm to generate a finite state approximation for any context-free grammar where the approximation is a superset of the language accepted by the grammar. This guarantees that no string acceptable to the parser will be precluded during recognition. Thus the FSA may be implemented to provide efficient constraints for a recognizer while the full CFG is used by a parser for analysis.

The final paper in the session concerns a different type of knowledge, prosodic information. Mari Ostendorf described a system which uses prosodic phrase breaks to disambiguate parses. "Break indices" are computed between each pair of words in an utterance. These indices are automatically assigned using a Hidden Markov Model of relative duration of phonetic segments. The break index gives an indication of the relative tightness of coupling between adjacent words. The parser uses this information to decide between alternative ways of grouping words into phrases. Results for a small test set of sentences were encouraging. Fourteen sentences with prepositional ambiguities were tested. The use of the break indices significantly reduced (by about 25%) the number of parses produced for these sentences without eliminating any correct parses.