Towards Unrestricted Syntax for Speech Understanding

2Q−9

Nigel Ward[1]

Mech-Info Engineering, University of Tokyo,

1 Motivation

Traditional models of syntax assume each utterance can be given a complete parse tree. This view is not well suited for understanding spoken language utterances where the speaker speaks in a fragmented, spontaneous style. To make a speech-friendly parser, there have been many efforts to modify traditional parsers; but I have taken a different approach: to radically re-examine the syntactic aspects of speech understanding. The goal is an "unrestrictive parser", meaning a syntactic mechanism suitable for understanding all utterances, thereby removing the need to restrict the user to speak grammatically.

The first issue is dealing with the inherent uncertainty of speech input, especially for noisy inputs. Any recognizer produces large numbers of word hypotheses. So that the parser can do its work, these are traditionally reduced somehow into one or a few "sentence hypotheses" (typically this reduction relies on a separate "language model"). Doing so can lose a lot of information; thus syntax is a bottleneck. To eliminate this problem, syntax should work directly from the raw word hypotheses, numerous as they are.

The second issue is feedback. Because purely bottom-up speech recognition is hard, effective mobilization of "higher-level knowledge" to aid the recognizer is a major goal in speech understanding research. Traditionally semantic constraints have not been applied here (with two exceptions, 1. when syntactic and semantic processing is integrated into one module, as in semantic grammars, and 2. when semantic constraints are used for filtering complete interpretations.) The difficultly has been the lack of a syntactic mechanism which can, given some semantic feedback, compute the implications of that feedback for the word hypotheses.

The figure roughly indicates some of the kinds of information flow needed for speech understanding.

2 Role of Syntax

To meet these needs, syntax needs to be given a role somewhat different from its traditional role, as discussed in (Ward 1993). This section summarizes briefly.

For speech, the semantic interpretation of an spoken utterance is necessarily underdetermined by the input (Bates *et al.* 1993). Thus, a parser need not output complete interpretations. Rather, its job is

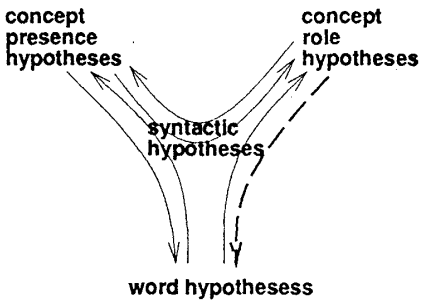

Figure 1: Some Desired Paths of Data Flow

to provide evidence for "conceptual hypotheses" of three types: 1. presence information, e.g., "this input involves *john*", "this involves a *beneficial action*", 2. linkage-information, e.g. "in this input *john* is related to *kiss*", and 3. relational information, e.g. "*john* is *active* in this input". Note that this inventory decomposes traditional case relations, such as "*john* is the *agent* of *kiss*", into one conceptual hypotheses of type 2 and several of type 3 (e.g, *john is .4 active*, *john is .2 responsible* ...) (Ward 1991; Ward 1994). This is done to make the individual conceptual hypotheses small and simple (and thus easy to relate to syntactic hypotheses and easy to score).

3 Mechanisms for Syntax

To handle inputs with many word hypotheses, and to transmit feedback from semantics to the word spotter, a parser for speech should be based on syntactic hypotheses which are considered in parallel, scored, and independent (Ward 1993). This view is very different from the traditional view of parsing as a symbolic (non-numeric process) whose goal is produce a single complete parse tree for each input.

In my system, syntactic hypotheses take the form "constituent X of construction Y was present over time span Z", for example, "the first constituent of the Subject-Predicate Construction was present over frames 17 to 25 of the input". Such "construction hypotheses" have the advantages of being simple, being suited to consideration in parallel, being easily scorable based on word hypothesis scores, and relating directly to semantics.

A construction hypothesis is spawned when there is a good match between the constituents of the construction and some of the word hypotheses in some time range.

A construction hypothesis which spans a certain time range is used for interpreting that part of the input. For example, suppose that: A. there is a hypothesized occurrence of the Subject-Predicate Construction for which the first constituent spans the time span from the 10th to the 22nd frame, and B. an occurrence of the word *"John"* is hypothesized in the time span from the 11th to the 19th frame. From this, since the time spans overlap, there is evidence for *john* being the subject, and hence *active* etc. Thus construction hypotheses provide evidence for conceptual hypotheses. Conversely, for feedback, semantic rescoring of such a conceptual hypotheses directly causes rescoring of the associated construction hypothesis (or hypotheses).

To avoid the complications involved in binding words to the constituents of constructions, the system uses "timelines". For example, if there is the hypothesis that "the part of the input from frame 11 thru frame 18 corresponds to direct-object", then there is evidence for *(affected .8)*, and this evidence is stored on positions 11 thru 18 of the timeline for *(affected .8)*. If there is also the hypothesis that "the word *"John"* appeared from frame 10 thru frame 16", then, by using the information on these timelines, the evidence regarding the degree of affectedness of *John* can be easily computed. Evidence from many construction hypotheses (etc.) is summed onto each timeline. As there is only a relatively small number of timelines, this technique is a relatively fast way to relate large numbers of word hypotheses, construction hypotheses, and conceptual hypotheses.

4 The Overall System

The system runs on a Sun SparcStation. Speech is input using a good microphone and the built-in 8Hz A/D converter. A simple template-matching word-spotter produces a lattice of word hypotheses, each consisting of word name, start point, end point, and score.

The system is set up as a question answerer: the overall task is to listen to "stories" and answer simple questions. There is no interesting semantics to this domain. Therefore semantic feedback is currently provided by human intervention. A window interface enables the developer to adjust the scores of the various conceptual hypotheses, to open feedback pathways, and to observe how the effects propagates through the system.

5 Results

Regarding the first issue raised in §1, this parser can, even when given many fairly corrupt word hypotheses, come up with fairly reasonable conceptual hypotheses.

Regarding the second issue, the system exhibits rich use of feedback. For example: Suppose A. the conceptual hypotheses *john-is-active* is highly ranked, and B. there is a hypothesized occurrence of the Subject-Predicate Construction where the first constituent is positioned around frame 12. Then there is evidence for any hypothesis that the word *"John"* appears in the input near frame 12 (using the fact that the first constituent of the Subject-Predicate Construction specifies the partial semantic role *active*.) (This sort of feedback is the dashed line in the figure.)

As another example, suppose conceptual reasoning leads to the hypothesis that "the word John probably appears in the input". This can be evidence for certain word hypotheses, which can lead to revised syntactic hypotheses, which can affect various other conceptual hypotheses.

6 Plans

– apply the system to spontaneous speech situations, where semantic feedback via syntax will provide invaluable help to the word spotter, for example, talking to a simple dog-sized mobile robot.

– quantify the advantage to providing semantic feedback via syntax to the word spotter, rather than than simply using it to rescore the n-best sentence hypotheses.

– build a fast word spotter. Since the system can cope with a plethora of word hypotheses, false hits are no problem. Thus I am developing an indiscriminate, but fast, word spotter.

– build an evidential semantic interpreter, that can make sense of the various scored conceptual hypotheses output by the parser, and can also provide feedback to the parser and recognizer in the form of re-scoring these conceptual hypotheses. I plan to explore abductive, preference-based, and connectionist approaches.

References

Bates, Madeline, Robert Bobrow, *et al.* (1993). The BBN/Harc Spoken Language Understanding System. In *1993 IEEE ICASSP*, pp. II–111–114.

Ward, Nigel (1991). Decomposing Deep Cases. In *43rd National Conference*, pp. 3:157–158. Information Processing Society of Japan.

Ward, Nigel (1993). On the Role of Syntax in Speech Understanding. In *Proceedings of the International Workshop on Speech Processing*, pp. 7–12. also Gijutsu Hokoku SP93-76, IEICE, Tokyo, 1993.

Ward, Nigel (1994). *A Connectionist Language Generator.* Ablex.

[1]supported in part by the Artificial Intelligence Research Promotion Foundation