Session 3: Natural Language Evaluation

Lynette Hirschman, Chair
Unisys Defense Systems
Center for Advanced Information Technology
Paoli, PA 19301

The session on Natural Language Evaluation focused on methods for evaluating text understanding systems. Beginning with the first Message Understanding Conference (MUCK-1) in 1987, there has been increasing focus on how to measure and evaluate text understanding systems. The MUCK-1 conference required developers to port their system to a common domain of Navy intelligence messages; MUCK-2 (May 1989) developed the first scoring system for evaluation based on template fill (also on a domain of Navy messages). MUCK-3 (to take place in late 1990 and early 1991) will refine that process and include an automated scoring procedure to grade quality of template fill. Meanwhile, a second evaluation effort has started, related to the MURASAKI multi-lingual text understanding project. It also uses the notion of evaluating systems doing template fill.

Both Murasaki and MUCK-3 were discussed during the Natural Language Evaluation session. The first (and only) paper of the session, – *Evaluating Natural Language Generated Database Records*, given by Rita McCardell (DoD), described a detailed scoring algorithm proposed for the Muraski project. The evaluation proposes to score template fills based on correctness, completeness and semantic proximity. This paper was followed by an informal presentation of plans for the third Message Understanding Conference, MUCK-3, by Beth Sundheim (NOSC).

During the discussion, several important points came up. Scoring algorithms for both Murasaki and MUCK-3 are complex and involve difficult issues, such as how to score template slots requiring multiple fills, whether and how much to penalize for incorrect answers, whether all slots count the same, etc. One area of concern was "how to score the scoring algorithm" -- that is, how to decide if the algorithm emphasizes the right things with respect to an intended application and how to calibrate the various weighting factors in light of that intended application. To date, "applications" have seemed fairly artificial, so that developers had little guidance as to the relative importance of precision (roughly how many mistakes the system makes) vs. recall (how many slots the system can fill in). In general, there was consensus that simpler scoring methods were preferable. Other suggestions included asking system developers to provide a range of results showing how precision varied with recall, and evaluating systems by giving them a multiple choice "reading comprehension" exam.