MCDONNELL DOUGLAS ELECTRONIC SYSTEMS COMPANY: MUC-3 Test Results and Analysis

David de Hilster and Amnon Meyers

Advanced Computing Technologies Lab
McDonnell Douglas Electronics Systems Company
1801 East Saint Andrew Place
Santa Ana, California 92705-6520
e-mail: vox@young.mdc.com
phone: (714)566-5956

RESULTS

INLET scored 25% recall and 35% precision after interactive correction. Non-interactive scores were 22% and 32%, respectively. Relative to other sites, we ranked 7th in recall and 9th in precision.

Because we are in transition from one system (VOX) to another (INLET), our scores reflect the performance of a very new and incomplete analyzer, as well as a small vocabulary and knowledge base. To be able to participate in MUC3, we implemented only a skimming capability, rather than a full-fledged syntax-driven language analyzer. Considering the status of our project, we feel INLET's performance is highly commendable. We ourselves were surprised by the success of skimming alone in performance on the MUC3 task, especially considering the preliminary nature of our work in this area.

TEST SETTINGS

Like many other developers of NLP systems, we have had no reason, up to now, to parameterize the trade-off between recall and precision. At various points in the past few months, we had tested heuristics that increased recall while reducing precision.

Participation in MUC3 has led us to examine such a parameter more seriously. In particular, a *confidence* metric, whereby the system assesses its own confidence in understanding a text, seems to be a useful component of NLP system development. We can attach such confidence factors to every action that the system takes in processing text, in extracting information from text, and in correlating the information to produce output.

As an NLP system improves and its knowledge of linguistics and the domain become more complete, confidence factors can be raised. Additionally, an arsenal of heuristics could be made available, depending on the confidence threshold assigned to the language processing task.

Assignment of confidence ratings can be assisted by empirical data. For example, we can assign an overall system confidence in identifying the incident type by scoring performance for the entire MUC3 corpus.

EXPENDITURE OF EFFORT

Once the skimmer was completed (April 22), we spent approximately 2 man-months customizing INLET to the MUC3 domain and task. A substantial portion of that time was spent in implementing code for filling the various slots in the MUC3 template and in developing heuristics to improve recall and precision. Somewhat less time was spent in building and applying generic and domain-specific grammar rules. System testing was relatively painless, because INLET typically processed 100 messages in 45 minutes, and several Sun Sparcstations were typically available for running tests. Unfortunately, we allowed too little time for vocabulary addition, so that many important words and phrases were omitted (e.g., "machinegun", "automatic weapons").

LIMITING FACTORS

Our main problems resulted from working with a new and incomplete system. Too often we had to devote our time to fixing bugs or making improvements in the skimmer, in the graphic representation tools, and in the knowledge addition tools. Starting with a small vocabulary and little linguistic and domain knowledge was disadvantageous. Adding a lot of knowledge to the system over a short period of time caused many problems to surface (e.g., initializing the system became a time-waster, system tables overflowed several times). Lack of an internal representation for information extracted from the text was yet another limitation on development.

TRAINING

We used the entire development corpus, including the key templates, for gathering domain information. For example, we used the key templates to get fairly complete lists of perpetrators, targets, instruments, and so on. Similarly, we searched the corpus for keywords, temporal, locative, and other patterns. Many of our domain-specific grammar rules were crafted using the results of such searches.

The first 100 messages of the development set served as a primary development and testing vehicle. TST1 messages were run occasionally in order to gauge progress on unseen message sets.

In order to shake out bugs in the system, we processed half the development set in batches of 100 messages several days before the testing deadline.

STRENGTHS AND WEAKNESSES

In general, skimming worked much better than expected. Based merely on our initial results for MUC3, we conclude that skimming is a powerful adjunct to deeper processing of text. We feel that with several months' work, continued development of skimming techniques, combined with knowledge base and vocabulary development, would substantially raise our MUC3 score.

Skimming provides extremely fast, simple, and robust text processing. While keyword and pattern-based methods for NLP have usually met with scorn, we feel a review of these methods is called for.

On the other hand, we are aware of the limitations of any approach that doesn't analyze text as deeply as possible. In order to segment incidents with great accuracy, linguistic context

as well as script-level understanding of the text are required. Many reference resolution problems also require such knowledge.

In the near future, we will merge our skimming capability with a bottom-up syntactic analysis mechanism, and also incorporate script-based understanding mechanisms.

The INLET customization tools have proved their worth by supporting hierarchy, grammar rule, and vocabulary addition. Even our qualified success would have been impossible without the effectiveness of the knowledge addition framework.

HITS AND MISSES

Our system is fairly good at determining the incident type, using a hierarchy of key words and patterns. With just a few specialized rules, the system is able to process appositives to find perpetrators, perpetrator organizations, physical targets, and human targets. An extensive temporal grammar was developed, though not much correlation of multiple temporal references has been implemented. A similar situation holds for locative phrases.

Simple gaps in knowledge and vocabulary caused many misses on the TST2 messages. Missing vocabulary (e.g., "killings"), missing domain rules (e.g., "explosion caused damage"), missing generic rules (e.g., "actor participated in action on object"), and missing mechanisms of various kinds led to substantially lower performance than we would expect of a more mature INLET system. Missing mechanisms include lack of threat handling, lack of any inferencing capability, lack of spelling correction, and lack of rejection of incidents for even simple reasons (e.g., an abstract object such as the "economy" is attacked).

PORTABILITY

Because INLET is a customization shell, portability of the specific knowledge added to the system is not a major concern. In 2 man-months, we were able to achieve a 25% recall and 35% precision score with a relatively immature INLET system. When the system is completed, we expect similar customization time to result in a better system for the particular domain and task.

The skimmer framework and knowledge addition framework are generic, as is the core knowledge base and vocabulary. On top of this layer is a substantial body of domain-specific code and knowledge, which would necessarily have to be replaced for a new domain and task.

LESSONS LEARNED

We have demonstrated that the INLET knowledge addition framework and skimmer can quickly support customization to a new domain and task. We have found that the graphic interface for knowledge addition has speeded up customization over a system like VOX. Finally, we have found that skimming is a critical adjunct to deeper NLP.

Our participation in MUC3 has shown us the high value of formal testing and comparison with other NLP efforts. We intend to continue using the MUC3 corpus and testing system for our system development, test, and evaluation.