THE SRI TIPSTER II PROJECT

Steven Maiorano
Office of Research & Development
Washington, D.C. 20505
E-mail: smaioran@ord.gov
Telephone: 703-351-2701

Introduction

The SRI TIPSTER Phase II program focused on supporting the development of an integrated architecture by helping to define the TIPSTER architecture and improving the portability of data extraction applications by enabling users to define and tailor their own information needs.

Architecture Support

SRI participated in the Architecture Working Group (AWG) meetings and aided in the design, testing, and implementation of the TIPSTER document manager architecture. Their contributions concerned input on the nature of basic entities, such as documents and text segments, and ways of communicating information from extraction modules to other modules in order to allow extraction and detection modules to work together. Subsequent to this design work, SRI made their FASTUS information extraction system compliant with the TIPSTER architecture, and integrated it with the New Mexico State University implementation of the architecture that was demonstrated at the Tipster 12-month workshop in May 1995. Also incorporated into this demo were three processing modules of the generic FASTUS system: name finder, phrase finder, and table recognizer.

The Generic System

Generic information extraction (IE) systems today adequately identify in text basic entities such as companies, personal names, and places. This performance promises that information extraction technology will becomes more widely utilized in real-world environments [1]. The main obstacle to deployment immediately is that generic systems must improve to the point where they can recognize other entities as specified by users who are not developers or computational linguists. SRI has been focusing its research efforts on developing the domain portability tools necessary for users to customize generic IE and capture previously unidentified entities in text.

The MUC-6 evaluation in November 1995 provided a good framework for testing the effectiveness of SRI's customization tools. SRI's excellent results indicate that their approach is on the right track. On both the Named Entity and Coreference Resolution Tasks, the FASTUS system was one of the top performers. The MUC-6 Scenario Template Task provided and even more rigorous test of SRI's domain transportability strategy since participants had only a month to achieve a high level of performance [2].

Support of Transportability

Two of the key elements in support of transportability are major SRI research areas: FastSpec and metarules [3].

The first area of research in support of transportability and user customization was the development of the pattern specification language FastSpec. This language allows one to define patterns for the FASTUS system in a convenient way. SRI also succeeded in accelerating FastSpec's compile time which had positive impact on development time in implementing domain-specific applications such as the MUC-6 dry run and test scenarios of labor negotiations and management succession. FastSpec is one of the influences on standardizing a community-wide pattern specification language.

The second key area of research in support of transportability was the implementation of compile-time transformations. These are metarules that allow the user to specify the simple domain-relevant subject-verb-object (S-V-O) patterns and have the system expand them into all the linguistic variants with the same semantic content including passives, gerunds, infinitives, and relative clauses. E.g., variations such as "cars are manufactured by GM" and "GM is to manufacture cars" can be generated automatically by the simple active S-V-O pattern "GM manufactures cars." Implementation of these metarules enabled SRI to bring the FASTUS system up to a high level of performance in a matter of days for the MUC-6 evaluation.

Ongoing Work and Future Directions

FastSpec and metarules constitute the basis for ongoing efforts to improve system transportability and customization further. SRI is developing an architecture for learning rules by example where, in a highly developed grammar, a constrained version of the rule -- or metarule -- is generated from a user's annotations; for relatively undeveloped grammars, simple S-V-O patterns are recognized in annotations and corresponding variant patterns are generated. In short, the goal is to have the system "observe" users and learn from their annotations rules of maximum generality from the minimum number of examples.

References

[1] Hobbs, J. R., Appelt, D. E., Bear, J., Israel, D., Kameyama, M., Stickel, M. and Tyson, M.; "*FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text*;" in Roche, E. and Schabes, Y., eds.; Finite-State Devices for Natural Language Processing; MIT Press; Cambridge, Mass.;1996.

[2] Appelt, D. E., Hobbs, J. R., Bear, J., Israel, D., Kameyama, M., and Tyson, M.; "*SRI International FASTUS System MUC-6 Results and Analysis*;" in Proceedings of Sixth Message Understanding Conference (MUC-6); Columbia, Maryland; November 1995.

[3] Hobbs, J. R., Appelt, D. E., Bear, J., Israel, D., Kameyama, Kehler, A., M., Stickel, M. and Tyson, M.; "*SRI's Tipster II Project*;" in this volume.