DOMAIN-INDEPENDENT NATURAL LANGUAGE INTERFACES: SESSION INTRODUCTION

Aravind K. Joshi
Department of Computer and Information Science
R. 268 Moore School
University of Pennsylvania
Philadelphia, PA 19104

In my comments,I will try to summarize briefly the six papers in this session, pointing out some key aspects of each contribution, listing some common themes,and raising some questions, not for the purpose of criticizing one or another contribution, but rather for the purpose of prodding the authors to discuss these questions in their presentations.

The six papers in the session are(alphabetically)

1. [GJM] J.M. Ginsparg, A robust and portable natural language data base interface.

2. [GHF] R. Grishman, L. Hirschman, and C.Friedman, Isolating domain dependencies in natural language interfaces.

3. [GRO] B.J. Grosz, Team: A transportable natural language interface system.

4. [TB] M. Templeton and J. Burger, Problems in natural language interfaces to DBMS:a retrospective view of the EUFID system.

5; [TT] B.H. Thompson and F.B. Thompson, Introducing ASK, a simple knowledgeable system.

6. [GT] G. Guida and C. Tasso, IR-NLI: an expert natural language interface to online data bases.

I. BRIEF SUMMARIES

[GIN]:A robust NL data base interface has been described. The system consists of two parts:a natural language processor for building a formal representation and an application program for building a query in an augmented relational algebra. Portability is achieved by providing (i) definitions of new concepts for the NL processor and (ii) the connection between the relations in the data base and NL processor concepts.

[GHF]:A robust portable NL interface is described, the portability evaluted with respect to two application domains. The initial processing is based on the Linguistic String Parser. The domain dependent information is characterized by domain information schema which specify the correspondences between information structures in the texts and the various internal representations. Modularity of design is emphasized for achieving portability and managing complexity.

[GRO]:The major theme is the construction of transportable NL interfaces. A unique feature of the TEAM system is the interactive facilty for acquiring information for adapting NL interfaces to data bases for which they were not handtailored. The DIALOGIC system constructs a logical form for the query and the data base access system translates the logical form into a data base query. Transportability is achieved by factoring the domain dependent and domain independent information. Some acquisitional and control strategies are also discussed.

[TB]: This is a retrospective view of the EUFID system,tracing its develpoment from the initial design specifications, including various stages of intermediate implementations and experience with different applications, and finally,ending with a list of problems to be solved. TB conclude that robust NL systems do not exist and provide some guidelines for the design of such systems.

[TT]:The NL interface(ASK) is meant for a user who wants to create, test change, augment, and, of course, use her/his knowledge base. The user interface is a limited subset of English. Fast response time is a major goal. Transportability is achieved via a dialogue in ASK which drives the Bulk Data Input Capability. Dialogues in ASK can also be used to design dialogues. A unique feature of the system is that it has been already implemented on a desktp computer,HP 9846.

[GT]:An NL interface, IR-NLI, is described, which is meant for non-professional users for accessing on-line data bases. The basic components of the system are described in general terms. An illustrative example is presented in which , in addition to the user-system dialogue, the internal representations and the search strategies are reported in the Euronet DIANE EUROLANGUAGE. Several future directions are suggested,including the system taking more initiative and providing justification of its mode of operation.

II. SOME QUESTIONS FOR DISCUSSION

1. All authors have dealt with the issue of domain independence, an important aspect of portability and robustness. The main strategy for achieving this behaviour is to factor the system into two parts, one based on domain independent knowledge and the other on the domain dependent knowledge. At least two systems[GRO and TT] talk about interactively acquiring the information needed for adaptation. GRO and TT should discuss the similarities and differences in their approaches to acquisition.

2. GIN uses the preference information in the data base connection part of his system to induce coercions(e.g., ´a doctor within 200 miles" meaning ´a ship with a doctor on it within 200 miles). GRO in TEAM achieves the same effect by inference rules. It is not clear just what the difference is between GRO´s approach and GIN´s approach, i.e., between inducing coercions and making the inferences. GRO nad GIN should discuss the realtive merits of their approaches.

3. The domain information schema of GHF specify the correspondence between information structures as they appear in the text and the various internal representations of information in the system. The system described by GHF is the only one in this session which derives its domain dependent knowledge from the texts instead of domain experts. GHJ should discuss how successful they are with this approach. Also perhaps they should discuss how their representations compare with those in TEAM[GRO].

4. TB have reviewed the history of EUFID and ended up on a negative note. They feel robust systems do not really exist yet. Since all the other authors have described ´robust´ systems, perhaps TB should discuss their conclusions in some detail and comment on the claims of these authors.

5. TT have empahsized the speed of response as a major feature of their system. What are the trade-offs between speed and the modularity of the design ? TT should comment on this aspect as well as other related aspects due to their concern for bringing up the system on a desktop computer.

6. Most of the issues discussed by GT in the first four sections of their paper pertain to making formal query language(FL)interfaces also portable. In fact, many of the issues about portability that the other authors have addressed to are also related to FL interfaces. It would be very useful if some(or all) of the authors discuss some specific problems about portability that are special to NL interfaces in contrast to FL interfaces.

Does the need for having flexible and portable interfaces for data bases necessarily force us towards NL interfaces ? We hope so ! Perhaps, some of the authors will comment on this issue. Their specific experience with their own systems would be very relevant here.

7. A question that is worth discussing by all the authors is whether portability and robustness can be helped if the design of the data base itself is determined by the consideration that an NL interface will be hooked to it ?