

The FINITE STRING Newsletter

Computational Linguistics Research at the University of Pennsylvania

Faculty: Tim Finin, Aravind Joshi, Anthony Kroch, Ellen Prince, Gillian Sankoff, Bonnie Webber, Scott Weinstein, Ralph Weischedel (visiting).

Students: Debby Dahl (post-doc in Cognitive Science), Anny Ewing, Julia Hirschberg, Sitaram Lanka, Eric Mays, Kathy McCoy, Gopalan Nadathur, Susan Pintzuk, Martha Pollack, Robert Rubinoff, Ethel Schuster, David Weir, Amy Zwarico.

1. Introduction

This is a brief, informal and, because of Christmas holidays, regrettably partial survey of current research in computational linguistics at the University of Pennsylvania. Any inaccuracies can be blamed on the departmental egg nog.

2. Extending the Range of Interactive Behavior

Perhaps the most activity here in computational linguistics is aimed at extending the kinds of behavior that can be supported in interactions with data base and expert systems. Elsewhere we have argued that such systems have to do more than retrieve and present appropriate facts and conclusions if they are to satisfy their users' real needs (Pollack, Hirschberg, and Webber 1982). Out of this conviction have already come systems able to recognize and respond to two types of presupposition failures (Kaplan 1982, Mays 1980) and a system able to describe in Natural Language what it knows about various entities (McKeown 1982). The following snippets indicate our current efforts in this area.

2.1. Recognizing and Responding to Belief Discrepancies

One reason for unsuccessful interactions is that the participants fail to realize they hold different beliefs about the world. Or if they do realize it, they fail to do anything about it. As a result, each leaves the interaction with very different ideas about what was communicated. Such discrepancies differ widely with respect to how easy they are to recognize and square away. A disparity revealed on the surface by some utterance or act may nevertheless be deeply embedded in a person's whole system of beliefs and hence potentially expensive to square away.

In the case of user-system interactions, as difficult as the general problem may be, certain discrepancies between the user's beliefs and those of the system may be clearly revealed in the user's utterances to the sys-

tem and easily squared away. This has been a focus of previous research here (Kaplan 1982, Mays 1980, Webber and Mays 1983) and is currently the focus of some new research on recognizing and responding to object-related discrepancies (McCoy 1983). There are several manifestations of these object-related discrepancies

1. the user's utterance describes an object in terms of a class it doesn't belong to;
2. the utterance incorrectly attributes some property to an object that it doesn't have; or
3. the utterance ascribes an impossible value to some property the object does have.

For example,

S: Do you have any liquid assets?

U: I have a \$5k money market certificate.

S: A money market certificate isn't a liquid asset.

Your money is tied up for several years in a money market certificate. Do you mean a money market account?

U: What's the interest rate on this stock?

S: Stocks don't have an interest rate. They may pay a dividend periodically.

Our work to date in this area has concentrated on factors involved in

- ▶ constructing appropriate responses (that is, what information to include, even if the system cannot establish for certain what the underlying discrepancy is), and
- ▶ representing the system's beliefs and its model of the user's beliefs in a well-motivated and well-structured way.

This work is being done by Kathy McCoy, with supervision from Bonnie Webber and Aravind Joshi.

2.2. Interacting with "Dynamic Data bases"

Most data bases are subject to change in the form of "updates". In the past, following an update, the previous information was either lost, archived or merged into some summary information. Now, optical disk technology can enable an organization to keep this information accessible. Such a data base has been called a "transactional" or "historical data base" (Clifford and Warren 1983). Very often, there are constraints on possible changes to the data base. However, this knowledge of "possible futures" is generally only available as mechanically applied update constraints – the system cannot reason with it. In contrast, we have been working on enabling a system to reason about how its data base has changed up to *now* and how it can change in the future. We have termed

this type of historical data base system a “dynamic data base system” and have described its implementation using a branching time temporal logic (Ari, Manana, and Pueuli 1981) in Mays (1982, 1983).

In the context of dynamic data bases, we are developing support for several valuable interactive functions, including enabling the system

- ▶ to recognize and square away beliefs that reflect ignorance of some event or its consequences (Webber and Mays 1983) – for example,

U: Is John registered for CSE220?

S: No, he can't be, because he already has advance placement credit for it.

- ▶ to offer to monitor for additional information it will provide to the user if and when it learns of it (Mays 1983) – for example,

U: Did John take CSE110 last term?

S: No. Do you want me to let you know if he does take it?

U: Did John pass CSE110?

S: No, the semester hasn't ended, so he hasn't received a grade yet. Shall I let you know then if he has passed CSE110?

To perform these useful services, a system must be able to recognize what sequences of events are possible or what additional information it might really acquire. Otherwise, it might not distinguish between the following two situations, in which the same response is clearly not appropriate:

U: Is the JFK within 15 miles of San Francisco?

S: No, but shall I let you know when it is?

U: Is Santa Cruz within 15 miles of San Francisco?

S: No, but shall I let you know when it is?

This work is being done by Eric Mays, with supervision from Aravind Joshi, Bonnie Webber, and Scott Weinstein.

2.3. “Expert Answers”

To date, expert systems and help systems alike have been banking on an assumption that the user knows what advice he/she needs and knows how to ask for it. As the audience for such systems grows, this assumption becomes less and less warranted: people ask for information they believe will help them, when in point of fact, it won't (or it won't do so efficiently). What a real expert does under such circumstances is answer the question the user should have asked, possibly after interacting with the user to establish what that question should be. For example, consider someone using one of the DEC-20 mail systems. Part way through creating a message, he accidentally types a Control-Z, whose effect is to end message creation and return to prompt level. He then asks someone

how to delete a control character, but is told that he can't – control characters are not treated as text. So he proceeds to recreate his message.

If this user had consulted a real expert, the expert would probably have recognized from his request – “How do you delete a Control-Z?” – that the user had typed a Control-Z unintentionally and wanted to continue what he was doing, probably creating a message. While there is no way to delete a Control-Z, there is a way to return to composing a message – that is, by editing it. The expert's response would inform the user that to continue composing a message the user can enter the editor from where he is and add directly to the message fragment.

The goal of our work here, as reported in Pollack (1983), is to enable an expert system or help system to deduce, from an incomplete or inappropriate query, what advice is actually needed and thereby generate appropriate responses to their users. Thus far, we have mapped out the sequence of processes involved in generating such responses and have experimented with at least one representation system (a type of dynamic logic, Rosenschein 1981) in which to do the reasoning involved. This work is being done by Martha Pollack, with supervision from Bonnie Webber and Aravind Joshi.

2.4. “Expert Questions”

One aspect of user-expert system interactions involves the system attempting to get from the user the information it needs to help solve his/her problem. The most commonly used way of getting information is via “menus” – essentially, multiple choice questions. However, there are several problems with relying on menus:

- The user may not understand either the question or the menu options.
- The user may be influenced by the options – that is, he/she assumes one of them must be appropriate to his/her case, so he/she bends the facts to fit the options.
- The user may not be satisfied with any of the options – that is, none seems appropriate to his/her case.
- The user may want to qualify his/her response – that is, he/she may feel that simply agreeing to a particular option will be misconstrued.

In all these cases, the reliability of the user's response is called into question.

Our research in this area attempts to provide users with as much freedom in responding to questions as other systems provide users in asking them. This involves at least the following:

1. providing the user with more help when he/she doesn't know how to respond.
2. allowing users more leeway in how they provide the requested information, along with any additional

information they believe relevant and want to convey as well.

Work on the first area is just getting started, with the implementation of a Prolog-based expert system that can reason about whether to ask for some information or try to deduce it. If it asks, but the user cannot answer, then the system has recourse to deducing the information or – and this is speculation – finding a way around it. This work is being done by Robert Rubinoff under the supervision of Tim Finin.

The next section describes briefly some research under way that is both of general interest for question-answering systems and of particular application for allowing the user more leeway in responding to questions.

2.5. Comprehension and Use of Scalar Implicature in Question Answering

A major problem in Natural Language processing is capturing the fact that hearers derive more from an utterance than its syntax and semantics encode. Indirect responses to yes/no questions, for example, often permit inference of both the direct response and additional implicit information. For example, in the following Q may infer that R's direct response to her query

Q: Has Jones taken all his medication?

R: He's had some of it.

is either 'no' or 'I don't know'. She will also infer that R believes there may be some prescribed medicine Jones has not taken.

Models of formal reasoning cannot explain such inferences: In standard logic, if there is some medicine that John has taken, it does not imply that it is not all his medicine that he has taken. Studies of indirect responses have generally sought to explain these inferences in terms of particular higher goals of speaker and hearer (Hobbs and Robinson 1979). However, linguistic pragmatics provides a more general explanation in the concept of scalar implicature. Our goal in this area is the formalization of scalar implicature to facilitate the interpretation and generation of cooperative responses in human-machine interaction (Hirschberg 1984).

The concept of conversational implicature comes from Grice (1975): An utterance conversationally implicates a proposition P when it conveys P by virtue of the Conversational Principle. Following Grice, Horn (1972) observed that, when an utterance employs a scalar x on some scale m defined by semantic entailment [W semantically entails T iff T is true whenever W is.], such that y is less than x and x is less than z , then x represents the highest value on m consistent with S 's observance of Grice's Maxim of Quality: "Be truthful." Any proposition formed by substituting in P some z that is higher on m than x is there-

by marked by S either as not believed to be the case or believed not to be the case. Any proposition formed by substituting in P some y lower on m than x will be true by entailment. Gazdar (1979) later termed this phenomenon "Scalar Quantity Implicature" (SQI).

We have found Horn's definition is too limited for our purposes. Utterances referencing entities, attributes, events, or states that may be viewed as ordered by some metric, such as set/set-member, process stages, or ISA hierarchy, permit similar implicatures:

Q: Has Jones registered yet?

R: He's filled out the insurance forms.

Here *registration* and *filling out insurance forms* are successive stages in a hospital admissions process. By his response, R conveys implicitly that either he doesn't know the direct answer to the question or the direct answer is no.

By expanding the definition of scale to include additional metrics, extending the notion of scalar implicature to define implicatures resulting from the denial of a scalar, and recognizing the correspondence between responses referencing a higher value and responses referencing a lower value, one can develop a powerful tool for deriving implicit information from scalar assertions (Hirschberg 1984). Systems may then use scalar implicature to derive inferences from user assertions and to avoid conveying unwanted inferences themselves. This work is being done by Julia Hirschberg, with supervision from Bonnie Webber and Aravind Joshi.

2.6 Avoiding false inferences

In his 1982 paper on Mutual Belief, Joshi shows that one of Grice's (1975) maxims, the Maxim of Quality:

Be truthful

doesn't go far enough towards accounting for cooperative conversational behavior vis-a-vis conveying information. He proposes the following revision:

Do not say anything which may imply for the hearer something that you, the speaker, believe to be false.

We are now working on applying this revised principle to question answering. In particular, the goals of the research are:

1. to characterize tractable cases in which the system as respondent (R) can anticipate the possibility of the user/questioner (Q) drawing false conclusions from its response and can alter or expand its response so as to prevent it happening;
2. to develop a formal method for computing the projected inferences that Q may draw from a particular response, identifying those factors whose presence or absence catalyzes the inferences;

3. to enable the system to generate modifications of its response that can defuse possible false inferences and that may provide additional useful information as well.

Thus far, we have been focussing on one class of false inferences – ones that follow from over-generalizing the situation to which the response is applicable. We have been attempting to identify ways in which the questioner might take the response too generally and to characterize the circumstances under which he/she might do so. This then leads to a characterization of the appropriate information to include in a response, to block any false conclusions that might so follow. What we see is that, if a system is to be able to anticipate and prevent false inferences that might be drawn from its response, it must be able to reason about, *inter alia*, shared assumptions, the goals of the user that underlie the question, “overloaded terms”, and expectations raised by Grice’s Cooperative Principle and its maxims.

This work is being done by Aravind Joshi, Bonnie Webber and Ralph Weischedel.

3. Aids in Creating Data Models

The goal of this research is to understand how structural descriptions can be learned – in particular, how they can be hypothesized from input data consisting of a finite set of Natural Language queries. One advantage of looking at queries is that they give implicit cues about what form the structure should take to provide an adequate response. For example, consider the query,

Which employees work on hardware and software projects?

The relation between employee and project can be either 1-1 or 1-many, information that must be encoded in the structural description to provide an adequate response. If the relation is 1-1, then the response will contain all employees who work on hardware projects and all employees who work on software projects. If it is 1-many, then the response will contain all employees who work on both hardware and software projects simultaneously.

Thus the cues to data base structure that queries provide relieve the user of the burden of explicitly mentioning all the nitty-gritty details. A second advantage to acquiring the data base structure from expected queries is that often the user starts off with a fuzzy notion of the domain but with a slightly better grasp on the requirements the data base schema should satisfy – that is, the range of queries it should be able to provide responses to. Hence, a collection of forecasted queries justifies as the input data.

We are developing a learning procedure (LP) that maps the incoming input data onto a set of consistent

hypotheses. To come up with a single hypothesis that best accounts for the data, LP will be guided by notions such as simplicity and naturalness, and above all, by feedback from the user. The tack here is for LP to narrow the number of possibilities as much as it can from the available information. When it requires further information to proceed, the user is called upon to provide that information. In this manner the learning procedure succeeds in arriving at the hypothesis that best fits the data.

This work is being done by Sitaram Lanka under the supervision of Aravind Joshi and Scott Weinstein. A technical report is in preparation.

4. Aids for Second Language Learning

People often rely heavily on their previous knowledge when learning a new skill. This previous knowledge can sometimes lead to misconceptions that hinder learning. By modelling users’ previous knowledge and resulting expectations, a cooperative Computer Assisted Instruction (CAI) system can better predict errors and thereby detect and correct them more easily. In particular, when tutoring a user in a new Natural Language, a CAI system might compare the user’s native language with the one to be learned to identify analogous patterns. These patterns can then be used to predict, detect and re-mediate errors.

We are building an interactive system, called 2WORD, that exploits this technique in teaching English as a second language. 2WORD concentrates on the acquisition of English two-word verbs, such as “turn up”, “dream about”, “talk over”, etc. These verbs comprise a lexical verb and a modifier, used together to produce a sentence that is both syntactically and semantically different from a sentence containing only the lexical verb. In addition, English two-word verbs do not map simply to two-word verbs in other Natural Languages. Even more insidious is the fact that a particular verb and modifier may not always function as a two-word verb or the same two-word verb. For example, in (a) “turn” is the verb and “up the road” is just a prepositional phrase like “to the left”. In (b), “turn up” is an intransitive two-word verb (wherein “up” is not a movable particle), while in (c) it is transitive - “up” can appear after the object, as in (d).

- (a) Raquel turned up the road.
- (b) Raquel turned up at our house this morning.
- (c) Raquel turned up the cuffs of her trousers.
- (d) Raquel turned the cuffs up on her trousers.

The consequence is that two-word verbs are particularly difficult for non-native speakers to acquire, thus providing a rich area for CAI. This work is being done by Ethel Schuster, with supervision from Bonnie Webber and Tim Finin.

5. A Formal Account of Intra-sentential Code-Switching

This project involves both an investigation into the syntactic constraints on intra-sentential switching from one language to another in the speech of bilinguals – for example,

Khob gered Idish in Shanghai TOO.
I spoke Yiddish in Shanghai too.

and work on a formal framework for the production and recognition of such sentences. Little is known about their syntax, production, or comprehension. A number of formal theories have been proposed, but none is based on a corpus of naturally-occurring tokens of the phenomenon. In the study of syntactic constraints, we have been testing hypotheses against statistical analyses of a corpus of naturally-occurring code-switching tokens in transcripts of interviews with Yiddish-English, Spanish-English, Polish-English, and Yiddish-Spanish bilinguals. The formal framework we have been working on consists of two grammatical systems and a mechanism for switching between the two. Syntactic constraints are then explained in terms of constraints on the switching mechanism. The faculty members involved in this study are Aravind Joshi, Ellen Prince and Gillian Sankoff. Students working with them include Anny Ewing, Susan Pintzuk, and Ethel Schuster.

6. A Unified Account of Referring Expressions in Discourse

This research is aimed at sorting out some of the confusion about the roles that syntactic, semantic and pragmatic factors play in the use and interpretation of definite descriptions and referring expressions in discourse. Out of this we hope will come a theoretical framework that can account for a variety of discourse phenomena in which the three interact – for example, various types of ellipses. This work is being done by Aravind Joshi and Scott Weinstein, together with Barbara Grosz from SRI International, and is reported in Grosz, Joshi, and Weinstein (1983). It brings together work on global coherence – the way larger segments of discourse fit together (Grosz 1982); local coherence – the way individual sentences bind together to form larger discourse segments (Sidner 1982); and centering – a local focusing process involving the single entity that an individual utterance most centrally concerns, that works to integrate an utterance into a discourse (Joshi and Weinstein 1981). The work is of particular significance for language generation, where one does not want to call upon powerful inference mechanisms to guide or bless the speaker's use of a particular referring phrase.

7. Grammar Formalisms – TAGs

During the last few years there has been vigorous activity in constructing highly constrained grammatical systems by eliminating the transformational component either totally or partially (Kaplan and Bresnan 1983, Pullum and Gazdar 1982, Peters and Ritchie 1982). There is increasing recognition of the fact that the dependencies that transformational grammars have tried to account for can be captured satisfactorily by classes of rules that are non-transformational and at the same time highly constrained in terms of the classes of grammars and languages that they define.

Two types of dependencies are especially important: subcategorization and filler-gap dependencies. In fact, one motivation for transformations was to account for unbounded dependencies. The so-called “non-transformational grammars” account for unbounded dependencies in different ways. In the formalism under study here (Joshi 1983) – tree-adjoining grammar (TAG) – unboundedness is achieved by factoring the dependencies and recursion in a novel and, we believe, linguistically interesting manner. All dependencies are defined on a finite set of basic structures (trees) that are bounded. Unboundedness is then a corollary of a particular composition operation, called adjoining, that preserves dependencies.

So far we have described TAGs, characterized some of their properties and compared them with other non-transformational grammars (Joshi 1983). Currently we are looking at the issue of linguistic relevance, both for language analysis and generation. This work is being done by Aravind Joshi and Tony Kroch.

8. Query-Driven Labelling of Objects in 3-D Scenes

If people are to interact comfortably and effectively with perceptual systems (visual, tactile, etc.), they must be provided with a better way of referring to objects and parts of scenes than with respect to a coordinate system – “the object centered at point $\langle x, y \rangle$ ”. A better means is in terms of labels: for example, in a street scene, one can label things as buildings, streets, cars, etc. – not only objects but parts of objects as well, like the median line down a street, the entrance to a building, etc. (Pointing is also a possibility but for scenes of any complexity must be augmented by a reference facility in the form of labels/descriptions.)

We are looking at the problem of assigning labels/descriptions and recognizing their referents in the context of a query driven image understanding system (being developed by faculty members Ruzena Bajcsy and Sam Goldwasser) permitting queries posed in Natural Language. In response to queries, the system will automatically generate strategies for recognizing observable objects and relationships. It is envi-

sioned as realizing a true interaction between a textual/conceptual data base and a pictorial data base.

If a system is to be able to label objects and refer to them correctly, it must be able to make use of its sensory data, together with knowledge of the consequences (for further interpretation and labelling) of a particular label choice. There are several problems we have begun to look at. The first has to do with the specificity of the labelling: an object labelled in response to one query may have to be relabelled in terms of its parts for another. On the other hand, separately labelled objects may later have to be relabelled as a single object. One should be able to take advantage of this, rather than starting from scratch each time. A second problem has to do with the fact that what is conceptually a part of an object is not necessarily something defined by natural segmentation, so a vision system must have other means of partitioning objects at its disposal. Finally, the appropriate way to describe objects is not simply a function of their geometric properties and relationships to one another. Correct usage depends in part on idealized properties of an object, once its identity is known.

This work is just getting off the ground; it is being done by Aravind Joshi and graduate student Amy Zwarico.

References

- Ben Ari, M.; Manna, Z.; and Pnueli, A. 1981 The Temporal Logic of Branching Time. In *Eighth Annual ACM Symposium on Principles of Programming Languages*. Williamsburg, Virginia.
- Clifford, J. and Warren, D. 1983 Formal Semantics for Time in Data bases. *ACM Trans. on Data Base Systems* 8(2): 214-254.
- Gazdar, G. 1979 A Solution to the Projection Problem. In Oh, C.-K. and Dinneen, D., Eds., *Syntax and Semantics*. Academic Press, New York, New York: 57-90.
- Grice, H.P. 1975 Logic and Conversation. In Cole, P. and Morgan, J.L., Eds., *Syntax and Semantics*. Academic Press, New York, New York.
- Grosz, B.J. 1982 Focusing and Description in Natural Language Dialogues. In Joshi, A.; Webber, B.; and Sag, I., Eds., *Elements of Discourse Understanding*. Cambridge University Press, Cambridge, England.
- Grosz, B.; Joshi, A.K.; and Weinstein, S. 1983 Providing a Unified Account of Definite Noun Phrases in Discourse. In *Proc. 21st Annual Meeting, Association for Computational Linguistics*. Cambridge, Massachusetts: 44-50.
- Hirschberg, J. 1984 Scalar Implicature and Indirect Responses in Question-Answering. In *Proc. CSCSI-84*. London, Ontario. (Submitted for presentation.)
- Hobbs, J. and Robinson, J. 1979 Why Ask? *Discourse Processes* 2.
- Horn, L.R. 1972 On the Semantic Properties of Logical Operators in English. Ph.D. thesis, University of California at Los Angeles.
- Joshi, A.K. 1982 Mutual Beliefs in Question Answering Systems. In Smith, N., Ed., *Mutual Belief*. Academic Press, New York, New York.
- Joshi, A.K. 1983 Factoring Recursion and Dependencies. *Proc. 21st Annual Meeting, Association for Computational Linguistics*. Cambridge, Massachusetts: 7-15.
- Joshi, A.K. and Weinstein, S. 1981 Control of Inference: Centering. In *Proc. 1981 Meeting, Int'l. Joint Conf. on Artificial Intelligence*. Vancouver, Canada: 385-387.
- Kaplan, J. 1982 Cooperative Responses from a Portable Natural Language Data Base Query System. In Brady, M., Ed., *Computational Models of Discourse*. MIT Press, Cambridge, Massachusetts.
- Kaplan, R. and Bresnan, J. 1983 Lexical Funal Grammar: A Formal System for Grammatical Representation. In Bresnan, J., Ed., *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge, Massachusetts.
- Mays, E. 1980 Failures in Natural Language Systems: Application to Data Base Query Systems. In *Proc. First National Conference on Artificial Intelligence (AAAI)*. Stanford, California.
- Mays, E. 1982 Monitors as Responses to Questions: Determining Competence. In *Proc. 1982 National Conference on Artificial Intelligence*. Pittsburgh, Pennsylvania.
- Mays, E. 1983 A Modal Temporal Logic for Reasoning about Change. In *Proc. 1983 Assoc. for Computational Linguistics Conference*. Cambridge, Massachusetts.
- McCoy, K. 1983 Correcting Misconceptions: What to Say. In *CHI'83 Conference Human Factors in Computing Systems*. Cambridge, Massachusetts.
- McKeown, K. 1982 Generating Natural Language Text in Response to Questions About Data Base Structure. Ph.D. thesis, University of Pennsylvania.
- Peters, S. and Ritchie, R. 1982 Phrase Linking Grammars. Technical Report, Department of Linguistics, University of Texas.
- Pollack, M.; Hirschberg, J.; and Webber, B. 1982 User Participation in the Reasoning Processes of Expert Systems. In *Proc. AAAI-82*. Carnegie-Mellon University, Pittsburgh, Pennsylvania. A longer version appears as Technical Report CIS-82-9, Department of Computer and Information Science, University of Pennsylvania.
- Pollack, M. 1983 A Framework for Providing Appropriate Advice. Technical Report CIS-83-28, Computer and Information Science, University of Pennsylvania, Philadelphia, Pennsylvania.
- Pullum, G. and Gazdar, G. 1982 Natural Languages and Context-Free Languages. *Linguistics and Philosophy* 4: 471-504.
- Rosenschein, S.J. 1981 Plan Synthesis: A Logical Perspective. In *Proceedings of the 7th Conference*. IJCAI: 331-337.
- Sidner, C.L. 1982 Focusing in the Comprehension of Definite Anaphora. In Brady, M., Ed., *Computational Models of Discourse*. MIT Press, Cambridge, Massachusetts.
- Webber, B. and Mays, E. 1983 Varieties of User Misconceptions: Detection and Correction. In *Proc. IJCAI-8*. Karlsruhe, Germany.