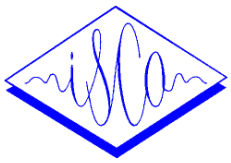


First International Conference on
Spoken Language Processing (ICSLP 90)
Kobe, Japan
November 18-22, 1990

Spoken Language System Integration and Development

Patti Price, Victor Abrash, Doug Appelt, John Bear, Jared Bernstein, Bridget Bly, John Butzberger, Michael Cohen, Eric Jackson, Robert Moore, Doug Moran, Hy Murveit, and Mitchel Weintraub

SRI International, Menlo Park, California 94025 USA

Abstract

SRI is developing a spoken language system (SLS) that should permit natural and efficient communication with an air travel information system. SLS development at SRI divides roughly into three areas: speech recognition, natural language processing, and human interface design.The paper presents an overview of SRI's development effort and an analysis of selected technical challenges in subparts of this effort, including the choice of initial domains for such technology, the architecture for the integration of the two technologies, the attributes of goal-directed spontaneous speech, and the evaluation of spoken language systems.

1.0 Introduction

Combining speech recognition and natural language understanding will vastly increase the number and range of potential applications for both technologies. Speech recognition without natural language results in a transcription of the words spoken; adding an interpretation of what those words mean opens a vast range of possibilities in human-machine interaction. Natural language technology without speech recognition requires typing skills and makes unnecessary demands on the eyes, the hands, and the brain. Freeing the eyes, hands, and brain of the user from the keyboard will allow for more efficiency, better use of visual displays and mouse interactions, interactive problem solving during hands-busy tasks, and flexible telephone applications. By using spoken natural language, the user can focus more on the problem to be solved and less on how to formulate it adequately for the computer.

A further motivation for the integration of speech recognition and natural language understanding is the belief that each technology could be improved by taking advantage of the other. Not every word can follow every other word. This is true in any language. Grammars are expressions of conditions on possible word sequences. Constraining the possible, or likely, sequences of words has had a major impact on large-vocabulary speech recognition because it effectively reduces the work done by the recognizer and eliminates many otherwise possible sources of confusion. Taking advantage of the grammatical constraints of a language could be important in improving speech recognition performance. With the exception of small domain-dependent grammars, such constraint to date typically comes from models of the statistical properties of word sequences. Such grammars have difficulty expressing constraints that are based on grammatical relations that may span an arbitrary number of words. Just as natural language constraints could improve speech recognition, information from speech could improve natural language understanding: Speech includes much information that is not indicated in the text, such as lexical, phrasal and contrastive stress, and prosodic groupings of words. Such information can aid lexical decisions (e.g., is the word "OBject" or "obJECT") as well as syntactic and semantic decisions.

The attempt to go beyond speech transcription and to go beyond text understanding by moving toward spoken language understanding opens an exciting new array of possibilities for human-machine interaction. It also opens a new array of issues not previously faced. The issues discussed in this paper include:

- The choice of initial domains for such technology
- The architecture for the integration of the two technologies
- The attributes of goal-directed spontaneous speech
- Evaluation of spoken language systems.

2.0 Domains

Spoken language understanding is a technology in its infancy. The first systems will be extremely limited, and we have little experience in the human factors issues of integrating the technology into an application. Spoken language understanding is an exciting area for human-machine interaction because people are used to solving problems interactively by voice. For this same reason, however, adding spoken language understanding to an interface may lead the user to believe the system has reasoning and understanding capabilities beyond current achievements.

Designing the human interface for inserting a new technology in an application is difficult, since we have no existing systems to observe. A promising technique for gaining the required data on human-machine interactions is the use of simulations of applications. Since variability across users in speech and language is quite large, initial systems should focus on applications in which a large population of potential users can be sampled. The data thus obtained can be used to develop initial systems and to develop methods for obtaining more such data efficiently for future systems.

The domain SRI has chosen for its first spoken-language, interactive, problem solving system is air travel planning. This domain has several important advantages as a first area:

- It takes advantage of an existing public domain real database, the *Official Airline Guide*, used by hundreds of thousands of people in the United States.
- It is a rich and interesting domain, including data on schedules and fares, hotels and car rentals, ground transportation, local information, airport statistics, trip and travel packages, on-time rates, and so on.
- A wide pool of users are familiar with the domain and can understand and appreciate problem solving in the domain. (This is crucial both for initial data collection for development and for demonstrating the advantages of a new technology to potential future users in a wide variety of domains.)
- The domain can be easily scaled with the technology, which is important for rapid prototyping and for taking advantage of advances in capabilities.
- The domain includes a significant amount that can be ported to other domains, such as generic database query and interactive problem solving.

3.0 Integration Architecture

3.1 Previous Approaches

A speech recognition component might communicate in several different ways with a natural language understanding component. Perhaps the most straightforward approach is a serial connection. In this scheme, the speech is input to the recognition system which, on the basis of the speech alone, outputs its best hypothesis to the natural language understanding system, which computes a meaning on the basis of text alone. There is no feedback in this scheme: the speech component does not have access to syntax and semantics in hypothesizing words, and the natural language component does not have access to, for example, the prosody of the speech for understanding contrastive stress. This approach has the advantage of being simple and of putting no additional effort into either of the two component technologies. It also has the advantage of requiring minimal communication between two culturally distinct groups: the engineers that dominate the speech recognition community and the artificial intelligence community that dominates natural language understanding.

Serial integration is, however, suboptimal because it does not take advantage of all the information available. A sentence that is misrecognized may have little hope of receiving a proper interpretation. We know that humans use a good deal of knowledge about syntax and semantics in interpreting what another person has said. A spoken language system should be able to take advantage of this information as well. Modifications to the strict serial architecture include sending a large lattice of words from the speech recognition component or a sequence of sentence hypotheses. This allows the syntax or semantics to explore more than just the best speech hypothesis. Sending a large lattice can reduce the error rate, provided the correct set of words is somewhere in the lattice or sentence list. Architectures of this type have been explored (Schwartz & Chow 1989; Paul 1989). However, a tighter integration should improve performance by allowing more communication among the components earlier in the process.

3.2 SRI's Frame-level Integration

More communication between the speech and the understanding components involve more complex architectures, but should improve both the speed and the accuracy of the spoken language system. SRI is investigating a unique frame-level integration (information between the two components is exchanged every 10 msec) that allows a computationally efficient use of natural language constraints in the speech recognition search. This system architecture allows for independent development yet integrated application of constraints from phonetics through semantics.

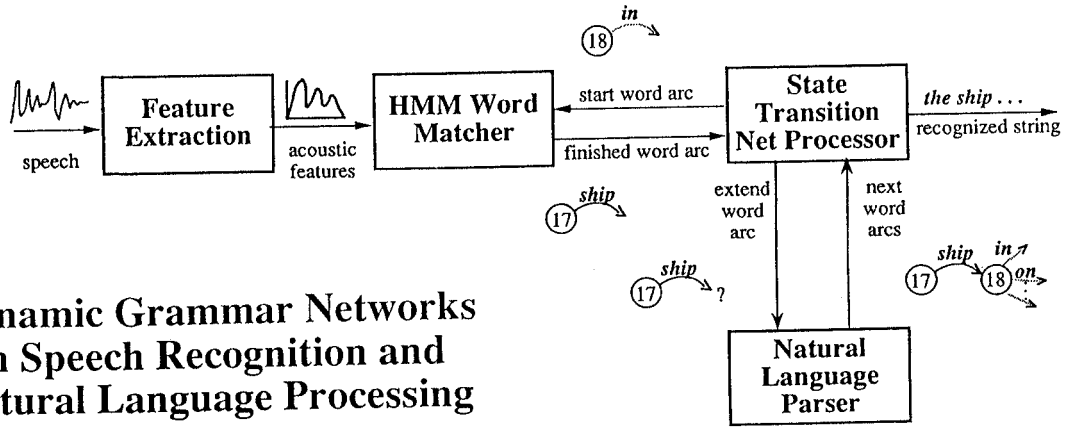
SRI's approach, called *dynamic grammar network* (DGN) generation (Murveit & Moore 1990), represents natural language knowledge in a state transition network, similar to finite-state language models used elsewhere for speech recognition systems. A straightforward implementation of this approach is not feasible, however, because typical NL systems would generate unmanageably large or infinite networks. Therefore, the network is generated on the fly, and only the portions of the network within a pruned search are expanded. Thus, the state-transition network generated for a particular spoken sentence will be relatively small, and different from that generated for a different utterance.

The approach is described graphically in Figure 1. The system runs as if it were a standard speech recognition system based on a hidden Markov model (HMM) using a language model based on a state-transition network. When the system is started up, the state-transition network contains an initial state, a list of the words that can leave that state (predictions), and markers indicating that the states that would be reached from these initial predictions are blocked--not yet included in the state transition network. The speech recognition (SR) system begins by searching for the words in the initial state's prediction list using a standard beam search. When a state is reached that is not in the network, the SR system calls the natural language processing (NLP) system which runs the parser, and creates the needed state. The SR system can then continue until it blocks again. The process of accepting the completion of a word from a state in the network and generating a new state is called a shift, as it corresponds to a shift in a *shift-reduce* natural language parser (Aho & Ullman 1979).

The shift process continues until the entire signal is exhausted. Words ending at the end of the signal are checked to see if they reach a final state--a state such that the hypotheses reaching that state are acceptable as complete utterances--and the most probable final-state hypothesis is chosen as the recognized sentence.

This approach allows a tight coupling of SR and NLP algorithms and has the following advantages:

- It brings all knowledge to bear as soon as possible so that extra work need not be done (for instance, the recognizer will not pursue hypotheses that can be ruled out by NLP and vice versa). In contrast to an equivalent system based on word lattices, a dynamic-grammar network system would not search portions of the signal that correspond to word-lattice entries that are unlikely according to previous acoustics or natural language.
- It allows for interactions between speech and NLP. For instance, an acoustic recognition model can be altered if the NLP system judges that the word should be emphasized given its syntactic or semantic position.



Dynamic Grammar Networks in Speech Recognition and Natural Language Processing

- In addition, this approach has the important advantage that, from the perspective of the recognition system, finite-state language constraints are used. Thus, all of the experience the speech recognition community has developed for dealing with finite-state-based speech recognition systems still applies to this system. For instance, a standard beam-search pruning technique is used in this system (Lowerre 1976).

4.0 Goal-Directed Speech

When a person is dictating to a system the goal is to communicate the words; the speaker is more likely to enunciate carefully and to focus on how the words are produced. When, however, a person is involved in interactive problem solving, the focus is not, or should not be, on the speech itself, but on the problem to be solved. This means that the speech is likely to be less careful and more casual. In particular, this means that there may be more variability in pronunciation, and that segments and syllables may be more likely to be reduced or deleted. It also means that more instances of "non-standard" grammatical forms will occur.

4.1 Phonological Variation

SRI has partially addressed the issue of phonological variation by incorporating detailed, statistically trained models of possible pronunciations for words (Cohen 1989, Cohen et al. 1990). The rules for pronunciation variations are created once for English and then can be applied to automatically generate a network of possible pronunciations for any new word. The likelihoods of the variants can also be automatically estimated on the basis of observations of the occurrences of similar instances in training data that need not contain the new words.

4.2 Grammatical Variation

The common production of non-standard grammatical forms brings into focus the trade-off between complete understanding of a given utterance and reliance on alternative techniques for interpretation. Even within a restricted domain, full understanding of any utterance, is difficult to accomplish. Language is productive, so new constructions appear frequently. Further, people often get distracted or change their minds in mid-sentence, which can result in wide deviations from "standard" language structure. Therefore, it seems useful to allow some flexibility in what the grammar will allow. However, accommodating more constructions typically requires more computation (and longer waiting time for the user), and also will provide less constraint (and thus make greater demands of accuracy on the recognition component). One solution to this problem is to bring more knowledge sources to bear, such as dialogue or plan models. However, a new domain has little data available on which to base a plan model, and poor models can perform worse than no model at all. At SRI we are exploring various combinations of tight and flexible grammars, trying to obtain the advantages of both. For the time being, SRI is pursuing the idea of cascading an analytical, linguistically-based grammar with a template-filler grammar so that the template filler can analyze those sentences that the analytical system cannot handle.

4.3 Template Grammar

In our initial work in this area, we have constructed a template-based grammar based on an analysis of frequently occurring patterns in the air travel planning domain.

We created templates corresponding to several common types of information that can be produced by the system (for example, schedules of flights, fares, seat availability, etc.) Templates are triggered based on the existence of keywords within a sentence, and multiple templates can be triggered for the same sentence. Templates contain slots such as the origin and destination of a flight in question. The slots are filled in from phrases following slot-keywords. Thus, for example, in the sentence "What flights leave San Francisco for Boston on Sunday?" the word flights will be a keyword triggering the "Flights" template, "leave" will cause the next phrase (if it is a city or airport) to be placed in the from-slot, "for" will cause the next phrase (if it is a city or airport) to be placed in the to-slot, and on (if the next phrase is a time) will cause the time slot for the flights question to be filled.

Template hypotheses are scored according to the percentage of content words used in filling the slots of the template. The template with the highest score is selected for interpretation. However, this grammar has a "cut-off" parameter for template scores that can be set to trade off wrong answers with no answers. That is, when the system is unsure, it can either guess, or admit that it doesn't know. Different applications would require different settings of this parameter. Our initial results with this system are very encouraging. On a fair test (testing on data not used in development) using DARPA standards for evaluation, we recently obtained the results shown in Table 1 for various settings of the cut-off parameter.

TABLE 1 PARSING PERFORMANCE AS A FUNCTION OF CUT-OFF

Cut-off	Right	Wrong	No Answer
0.0	55	13	22
0.833	42	4	44
1.0	37	2	51

These are very preliminary results, and much work remains to be done to combine the two grammars.

5.0 Evaluation

Progress can be measured and encouraged via standards for comparison and evaluation. Although qualitative assessments can be useful in initial stages, quantifiable measures of systems under the same conditions are essential for comparing results and assessing claims. Numbers are meaningless unless it is clear where they come from. The evaluation of any technology is greatly enhanced in usefulness if accompanied by documented standards for assessment. There has been a growing appreciation in the speech recognition community of the importance of standards for reporting performance. The availability of standard databases and protocols for evaluation has been an important component in progress in the field and in the sharing of new ideas. Progress toward evaluating spoken language systems, like the technology itself, is beginning to emerge. The following issues have been important in coming to agreement on standards for evaluation.

5.1 Independent Training and Test Sets

The importance of independent training/development data and testing data has been acknowledged in speech recognition evaluation for some time. The idea is less prominent in natural language understanding because, from a theoretical perspective, it may be important to work on a certain class of phenomena. In an application, however, the coverage of a certain class of phenomena must be weighed against the costs (how much larger or slower is the resulting system) and benefits (how frequently do the phenomena occur). The only fair test of coverage in this sense is a test on a sample of data similar to that to be used in the application, but not seen during development.

5.2 Black Box versus Glass Box Evaluation

Evaluating components of a system is important in system development, although not necessarily useful for comparing various systems, unless the systems evaluated are very similar, which is not often the case. Since the motivation for evaluating components of a system is for internal testing, there is less need to reach wide-spread agreement in the community on the measurement methodology. System-internal measures can be used to evaluate component technologies as a function of their design parameters; for example, recognition accuracy can be tested as a function of syntactic and phonological perplexity, and parser performance can be measured as a function of the accuracy of the word input. In addition, these measures are useful in assessing the amount of progress being made, and how changes in various components affect each other.

5.3 Quantitative versus Qualitative Evaluation

Qualitative evaluation (for example, do users seem to like the system) can be encouraging, but more convincing to those who cannot observe the system themselves are quantitative automated measures. Automation of the measures is important because we want to avoid any possibility of nudging the data wittingly or unwittingly, and of errors arising from fatigue and inattention. Further, if the process is automated, we can observe far more data than otherwise possible, which is important in research on language, where many units occur infrequently and where the variation across subjects can be large. For these measures to be meaningful, they should be standardized insofar as possible, and they should be reproducible.

5.4 Collecting Data for Evaluation

In order to collect the data we need for evaluating spoken language systems, we have developed a *pnambic* system (named after the line in the *Wizard of Oz*: "pay no attention to the man behind the curtain"). In this system a subject is led to believe that the interaction is taking place with a computer, when in fact the queries are handled by a transcriber wizard (who transcribes the speech and sends it to the subject's screen) and a database wizard who is supplied with a tool for rapid access to the on-line database in order to respond to the queries. The wizard is not allowed to perform complex tasks. The wizard may only retrieve data from the database or send one of a small number of other responses, such as "your query requires reasoning beyond the capabilities of the system." In general, the guidelines for the wizard are to handle requests that the wizard understands and the database can answer. The data must be analyzed afterwards to assess whether or not the wizard did the right thing.

5.5 Transcription Conventions

The session transcriptions, i.e., the sentences displayed to the subject, represent the subject's speech in a natural English text style. In order to perform automatic evaluation, we must agree on conventions for representing what the subject said, and we must implement procedures to ensure that these conventions are consistently used.

5.6 Canonical Answers and Scoring.

Canonical answers are, in general, the answer returned under the wizard's control. These answers will have to be cleaned up if the wizard makes an error, or if the answer given by the wizard was the (cooperative) context-dependent answer, which may differ from a context-independent answer, if it exists. Scoring is accomplished using standardized software, and conventions for inputs and outputs.

The process of coming to agreement on conventions for evaluation of spoken language systems, and implementing such procedures is difficult and time-consuming. However, the rewards of an automatic, common mechanism for system evaluation is worth the effort, and we believe that spoken language system development will benefit enormously from this effort.

6.0 Summary

In sum, workstations equipped with spoken language systems have the potential to increase user efficiency in interactive problem-solving. Natural language input allows the user to formulate more complex questions and commands more efficiently and more naturally. Spoken natural language can increase user efficiency, can reduce cognitive load, and can provide an alternate input modality to improve system robustness. SRI's research suggests that successful development of SLS technology requires an appreciation of the new challenges associated with acceptance of user input that cannot be defined beforehand. Furthermore, system integration design decisions can affect how well the system can deal with these new input forms.

References

A. Aho and J. Ullman (1979) *Principles of Compiler Design*. Addison-Wesley, Reading Mass.

M. Cohen (1989) "Phonological Structures for Speech Recognition," PhD Thesis, Computer Science Dept., University of California, Berkeley.

M. Cohen, H. Murveit, J. Bernstein, P. Price and M. Weintraub (1990) "The DECIPHER Speech Recognition System," *Proc. IEEE ICASSP-90*.

B. Lowerre (1976) *The Harpy Speech Recognition System*. PhD Thesis, Computer Science Dept., Carnegie Mellon U.

H. Murveit and R. Moore (1990) "Integrating Natural Language Constraints into HMM-based Speech Recognition," *Proc. IEEE ICASSP-90*.

D. Paul (1989) "A CSR/NLP Interface Specification," *Proc. of the DARPA Speech and Natural Language Workshop*, Oct. 1989.

R. Schwartz & Y-R Chow (1989) "The optimal N-Best Algorithm: An Efficient Procedure for Finding the Top N Sentence Hypotheses," *Proc. of the DARPA Speech and Natural Language Workshop*, Oct. 1989.

We gratefully acknowledge support from SRI internal funding, DARPA and NSF. Support from DARPA is through the Office of Naval Research contract N00014-90-C-0085. This material is based upon work supported by the National Science Foundation under Grant No. IRI-87204403. The government has certain rights in this material. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.