Evaluation of Spoken Language Systems: the ATIS Domain

P. J. Price

SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025

Abstract

Progress can be measured and encouraged via standards for comparison and evaluation. Though qualitative assessments can be useful in initial stages, quantifiable measures of systems under the same conditions are essential for comparing results and assessing claims. This paper will address the emerging standards for evaluation of spoken language systems.

Introduction and Background

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. This paper presents some background on the problem and outlines the issues and initial experiments in evaluating spoken language systems in the "common" task domain, known as ATIS (Air Travel Information Service).

The speech recognition community has reached agreement on some standards for evaluating speech recognition systems, and is beginning to evolve a mechanism for revising these standards as the needs of the community change (e.g., as new systems require new kinds of data, as new system capabilities emerge, or as refinements in existing methods develop). A protocol for testing speaker-dependent and speaker-independent speech recognition systems on read speech with a 1000-word vocabulary, (e.g., [6]), coordinated through the National Institute of Standards and Technology (NIST), has been operating for several years. This mechanism has inspired a healthy environment of competitive cooperation, and has led to documented major performance improvements and has increased the sharing of methodologies and of data.

Evaluation of natural language (NL) understanding is more difficult than recognition because (1) the phenomena of interest occur less frequently (a given corpus contains more phones and words than syntactic or semantic phenomena), (2) semantics is far more domain dependent than phonetics or phonology, hence changing domains is more labor intensive, and (3) there is less agreement on what constitutes the "correct" analysis. However, MUCK, Message Understanding Conference, is planning the third in a series of message understanding evaluations for later this year (August 1990). The objective is to carry out evaluations of text interpretation systems. The previous evaluation, carried out in March-June 1989, yielded quantitative measures of performance for eight natural language processing systems [4, 5]. The systems are evaluated on performance on a template-filling task and scored on measures of completeness and precision [7].

So far, we have discussed the evaluation of automatic speech recognition (i.e., the algorithmic translation from human speech to machine readable text), and of some aspects of natural language understanding (i.e., the automatic computation of a meaning and the generation, if needed, of an appropriate response). The evaluation of spoken language systems represents a big step beyond the previous evaluation mechanisms described. The input is spontaneous, rather than read, speech. The speech is recorded in an office environment, rather than in a sound-isolated booth. The subjects are involved in problem-solving scenarios. The systems to be tested will be evaluated on the answers returned from a common database. The rest of this paper focuses on the steps taken by the DARPA speech and natural language community to develop a common evaluation database and scoring software and protocols. The first use of this mechanism took place June 1990. However, given the greatly increased challenge, the first use of the mechanism is more a test of the mechanism than of the systems evaluated.

It has become clear in carrying out the evaluation mechanism that the needs of common evaluation are sometimes at odds with the needs of well-designed systems. In particular, the common evaluation ignores dialogue beyond a single query-response pair, and all interactive aspects of systems. A proposal for dialogue evaluation is included in [3], this volume.

Though the initial evaluation mechanism, described below, represents a major effort, and an enormous advance over past evaluations, we still fall short of a completely adequate evaluation mechanism for spoken language systems. Some forms of evaluation may have to be postponed to the system level and measured in terms of time to complete a task, or units sold. We need to continue to elaborate methods of evaluation that are meaningful. Numbers alone are insufficient. We need to find ways of gaining insight into differences that distinguish various systems or system configurations.

Issues

In this section we will outline the major evaluation issues that have taken up a good deal of our time and energy over the past several months, including: the separation of training and testing materials, black box vs. glass box evaluations, quantitative vs. qualitative evaluation, the selection of a domain, the collection of the data, transcribing and processing the data, documenting and classifying the data, obtaining canonical answers, and scoring of answers.

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. The focus in linguistics on competence rather than performance has meant that many developers of syntactic and semantic models have not traditionally evaluated their systems on a corpus of observed data. Those who have looked at data, have typically referred to a few token examples and have not evaluated systematically on an entire corpus. Still more rare is evaluation on an independent corpus, a corpus not used to derive or modify the theory or model. There is no doubt that a system can eventually be made to handle any finite number of evaluation sentences. Having a test suite of phenomena is essential for evaluating and comparing competing theories. More important for an application, however, is a test on an independent set of sentences that represent phenomena the system is likely to encounter. This ensures that developers have handled the phenomena observed in the training set in a manner that will generalize, and it properly (for systems rather than theories) focuses the evaluation of various phenomena in proportion to their likelihood of occurrence. That is, though from a theoretical perspective it may be important to cover certain phenomena, in an application, the coverage of those phenomena must be weighed against the costs (how much larger or slower is the resulting system) and benefits (how frequently do the phenomena occur).

Black Box versus Glass Box Evaluation

Evaluating components of a system is important in system development, though 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.

A useful means of evaluating system performance is the time to complete a task successfully. This measure cannot be used to compare systems unless they are aimed at completing the same task. It is, however, useful in assessing the system in comparison to problem solving without the spoken language system in question. For example, if the alternative to a database query spoken language system is the analysis of huge stacks of paperwork, the simple measure of time-to-complete-task can be important in showing the efficiency gains of such a system.

Time-to-complete-task, however, is a difficult measure to use in evaluating a decision-support system because (1) individual differences in cognitive skill in the potential user population will be large in relation to the system-related differences under test, and (2) the puzzle-solving nature of the task may complicate procedures that reuse subjects as their own controls. Therefore, care should be taken in the design of such measures. For example, it is clear that when variability across subjects is large, it is important to evaluate on a large pool of users, or to use a within-subject design. The latter is possible if equivalent forms of certain tasks can be developed. In this case, each subject could perform one form of the task using the spoken language system and another form using an alternative (such as examining stacks of papers, or using typed rather than spoken input, or using a database query language rather than natural language).

Quantitative versus Qualitative Evaluation

Qualitative evaluation (for example, do users seem to like the system) can be encouraging, rewarding and can even sell systems. 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 language, where the units occur infrequently and where the variation across subjects is large. For these measures to be meaningful, they should be standardized insofar as possible, and they should be reproducible. These are the goals of the DARPA-NIST protocols for evaluation of spoken language systems. These constraints form a real challenge to the community in defining meaningful performance measures.

Limiting the Domain

Spoken language systems for the near future will not handle all of English, but, rather, will be limited to a domain-specific sub-language. Accurate modeling of the sub-language will depend on analysis of domain-specific data. Since no spoken language systems currently have a wide range of users, and since variability across users is expected to be large, we are simulating applications in which a large population of potential users can be sampled.

The domain used for the standard evaluation is ATIS using the on-line Official Airline Guide (OAG), which we have put into a relational format. This application has many advantages for an initial system, including the following:

- It takes advantage of an existing public domain real database, the Official Airline Guide, used by hundreds of thousands of people.
- 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, and on-time rates.
- 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 good deal that can be ported to other domains, such as generic database query and interactive problem solving.

Related to the issue of limiting the domain is the issue of limiting the vocabulary. In the past, for speech recognition, we have used a fixed vocabulary. For spontaneous speech, however, as opposed to read speech, how does one specify the vocabulary? Initially, we have not fixed the vocabulary, and merely observed the lexical items that occur. However, it is an impossible task to fully account for every possible word that might occur, and it is a very large task to derive methods to detect new words. It is also a very large task to properly handle these new words, and one that probably will involve interactive systems that do not meet the requirements of our current common evaluation methods. However, there is evidence that people can accomplish tasks using a quite restricted vocabulary. Therefore, it may be possible to provide some training of subjects, and some tools in the data collection methods so that a fixed vocabulary can be specified and feedback can automatically be given to subjects when extra-lexical material occurs. This would meet the needs of spontaneous speech, of common evaluation and of a fixed vocabulary (where one could choose to include or exclude the occurring extra-lexical items in the evaluation).

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 online 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 the wizard did the right thing.

The subjects in the data collection are asked to solve one of several air travel planning scenarios. The goal of the scenarios is to inspire the subjects with realistic problems and to help them focus on problem solving. A sample scenario is:

> Plan a business trip to 4 different cities (of your choice), using public ground transportation to and from the airports. Save time and money where you can. The client is an airplane buff and enjoys flying on different kinds of aircraft.

Further details on the data collection mechanism is provided in [2] in this volume.

Transcription Conventions

The session transcriptions, i.e., the sentences displayed to the subject, represent the subject's speech in a natural English text style. Errors or dysfluencies (such as false starts) that the subject corrects will not appear in the transcription. Grammatical errors that the subject does not correct (such as number disagreement) will appear in the transcription as spoken by the subject. The transcription wizard will follow general English principles, such as those described in *The Chicago Manual of Style* (13th Edition, 1982). The tremendous interactive pressure on the transcription wizard will inevitably lead to transcription errors, so these conventions serve as a guide.

This initial transcription will then be verified and cleaned up as required. The result can be used as conventional input to text-based natural language understanding systems. It will represent what the subject "meant to say", in that it will not include dysfluencies corrected by the subject. However, it may contain ungrammatical input.

In order to evaluate the differences between previously collected read-speech corpera and the spontaneous-speech corpus, subjects will read the transcriptions of their sessions. The text used to prompt this reading will be derived from the natural language transcription while listening to the spoken input. It will obey standard textual transcriptions to look natural to the user, except where this might affect the utterance. For example, for the fare restriction code "VU/1" the prompt may appear as "V U slash one" or as "V U one", depending on what the subject said.

Finally, the above transcription needs to be further modified to take into account various speech phenomena, according to conventions for their representation. For example, obviously mispronounced words that are nevertheless intelligible will be marked with asterisks, words verbally deleted by the subject will be enclosed in angle brackets, words interrupted will end in a hyphen, some non-speech acoustic events will be noted in square brackets, pauses will be be marked with a period approximately corresponding to each elapsed second, commas will be used for less salient boundaries, an exclamation mark before a word or syllable indicates emphatic stress, and unusual vowel lengthening will be indicated by a colon immediately after the lengthened sound. Some of the indications will be useful for speech recognition systems, but not all of them will be included in the reference strings for evaluating the speech recognition output.

The various transcriptions are illustrated in the examples below, with the agreed upon file extensions in parentheses, where applicable:

- SESSION TRANSCRIPTION:
  Show me a generic description of a 757.
- NL TEXT INPUT (.nli):
  Show me a general description of a 757.
- PROMPTING TEXT (.ptx):
  Show me a general description of a seven fifty seven.
- SPEECH DETAIL (.sro):
  <list> show me: a general description, of a seven fifty seven
- SPEECH REFERENCE (.snr):
  SHOW ME A GENERAL DESCRIPTION OF A SEVEN FIFTY SEVEN

Data Classification

Once collected and processed, the data will have to be classified. Ambiguous queries will be excluded from the evaluation set only if it is impossible for a person to tell without context what the preferred reading is. Another issue is minor syntactic or semantic ill-formedness. Our guideline here is that if the query is interpretable, it will be accepted, unless it is so ill-formed that it is clear that it is not intended to be normal conversational English. All presuppositions about the number of answers (either existence or uniqueness) will be ignored, and these are the only types of presupposition failures noted to date. Any other types of presupposition failure that make the query truly unanswerable will no doubt also have made it impossible for the wizard to generate a database query, and will be ruled out on those grounds. Queries that are formed of more than one sentence will not automatically be ruled out. The examples observed so far are clearly interpretable as expressing multiple constraints that can be combined into a single query.

Evaluatable queries will be identified by exception, i.e., those that are none of the following:

1. context dependent,
2. vague, ambiguous, disambiguated only by context, or otherwise failing to yield a single canonical database answer,
3. grossly ill-formed,
4. other unanswerable queries (i.e., those not given a database by the wizard),
5. queries from a noncooperative subject.

Canonical Answers and Scoring

Canonical answers will, in general, be the corrected version of the answer returned under the wizard's control. These will have to be cleaned up in the case that 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. In the context of a database query system, the wizard is instructed to interpret queries broadly as database requests. Thus, we believe that "yes/no" questions will be in general interpreted as a request for a list, rather than the word "yes" or "no", as in "Are there any morning flights to Denver?" Other conventions involve treatment of strings for comparison purposes and case-sensitivity, the appearance of extra columns in tabular answers, and the inclusion of identifying fields (see [1] for details).

Scoring is accomplished using standardized software, and conventions for inputs and outputs. Comparing scalar answers simply means comparing values. Table answers are more interesting, since in general the order of the columns is irrelevant to correctness. For single-element answers, a scalar answer and a table containing a single element are judged equivalent, for both specifications and answers. For our first experiment with the new protocols, sites were only required to report results on the natural language component. The transcriptions were released a few days before the results were to be reported. One site, CMU, reported results on speech inputs. See [1] for further details on scoring.

Conclusions

The process of coming to agreement on conventions for evaluation of spoken language systems, and implementing such procedures has been a larger task than most of us anticipated. We are still learning, and sometimes it has been painful. However, the rewards of an automatic, common mechanism for system evaluation is worth the effort, and we believe the spoken language program will benefit enormously from this effort. There still is a good deal more work to do as we find ways to meet the constraints of evaluation in a way that makes sense for the development of spoken language systems.

Acknowledgements

This article is based on a perusing of the voluminous email and phone discussions involving numerous people from various sites, including BBN, CMU, MIT, MIT-LL, NIST, SRI, TI, and Unisys. The author gratefully acknowledges the important roles played by individuals from each of these sites. The program described is funded by DARPA, the particular contract that funded the writing of this paper is through DARPA under Office of Naval Research contract N00014-90-C-0085.

References

[1] L. Bates and S. Boisen, "Developing an Evaluation Methodology for Spoken Language Systems," this volume.

[2] C. Hemphill, J. Godfrey, and G. Doddington, "The ATIS Spoken Language Systems Pilot Corpus," this volume.

[3] L. Hirschman, D. Dahl, D. McKay, L. Norton, and M. Linebarger, "Beyond Class A: A proposal for Automatic Evaluation of Discourse," this volume.

[4] D. Pallett and W. Fisher, "Performance Results Reported to NIST," this volume.

[5] D. Pallett, chair, "ATIS Site Reports and General Discussion," Session 5, this volume.

[6] P. J. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition," *Proc. ICASSP,* 1988. Database available on CD-ROM.

[7] B. Sondheim, "Plans for a Task-Oriented Evaluation of Natural Language Understanding Systems," *Proc. of the DARPA Speech and Natural Language Workshop*, Feb. 1989.