CHALLENGES FOR SPOKEN DIALOGUE SYSTEMS

James R. Glass

Spoken Language Systems Group
MIT Laboratory for Computer Science
Cambridge, MA 02139
http://www.sls.lcs.mit.edu

ABSTRACT

The past decade has seen the development of a large number of spoken dialogue systems around the world, both as research prototypes and commercial applications. These systems allow users to interact with a machine to retrieve information, conduct transactions, or perform other problem-solving tasks. In this paper we discuss some of the design issues which confront developers of spoken dialogue systems, provide some examples of research being undertaken in this area, and describe some of the ongoing challenges facing current spoken language technology.

1. INTRODUCTION

The past decade has seen the development of a large number of spoken dialogue systems around the world, both as research prototypes and commercial applications. These systems allow users to interact with a machine to retrieve information, conduct transactions, or perform other problem-solving tasks. The architecture of these systems can vary significantly, ranging from interactive voice response (IVR) systems augmented with isolated word recognition (e.g., "Press, or say two"), to full-fledged natural language-based dialogue systems which allow for more unconstrained input from the user (e.g., "How may I help you?").

Ever since the creation of the VOYAGER urban navigation system in 1989 [22], researchers of the Spoken Language Systems group at MIT have been active in developing technology for spoken dialogue systems, and have produced a number of such systems over the years, including the JUPITER weather information system [66]. Rather than dwelling on the particular details of these systems however, in this paper we try to discuss some of the design issues which confront developers of spoken dialogue systems, provide some examples of research being undertaken in this area, and describe some of the ongoing challenges facing current spoken language technology.

The term "spoken dialogue system" has different meanings to different people, but generally implies an interactive system which operates in a constrained domain. One of the main ways in which systems differ is the degree to which the system takes an active role in the conversation. Systems which have a machine-directed dialogue will tend to ask the user a series of questions, much as an IVR system might interact with a user. Directed queries (e.g., "What is the departure city?") can result in shorter responses from the user, which will result in higher success rates. Many deployed systems have successfully used this strategy. This framework can also be enhanced by allowing knowledgeable (i.e., experienced) users to specify multiple constraints in a single utterance.

An alternative approach to dialogue is to employ a mixed-initiative strategy, whereby the system is more flexible in handling constraints from the user, and can typically process more complex linguistic inputs. These systems usually attempt to jointly negotiate with the user to help determine a set of constraints for the particular task. By allowing the user more flexibility however, these systems can have higher error rates, and can be more confusing to users unfamiliar with the technology. In order to reduce the severity of this problem, some systems deploy a hybrid approach, by backing off to a more constraining dialogue when the system detects it is having problems understanding the user (e.g., [50, 53]).

At a minimum, a spoken dialogue system requires an automatic speech recognizer (ASR) to perform speech to text conversion, some form of dialogue manager (controller) to control the interaction with the user, and a mechanism for conveying information to the user (e.g., text and/or speech generation). More complex systems will generally incorporate modules for ASR, natural language understanding (NLU), language generation, speech synthesis, and a mechanism for handling local discourse phenomena, in addition to the dialogue manager. Figure 1 shows a block diagram of a spoken di-

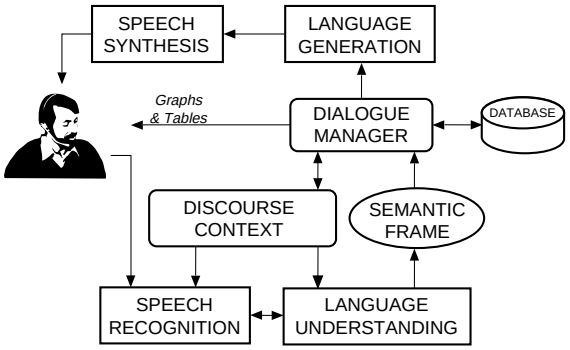

Figure 1: Block diagram of a spoken dialogue system.

alogue system used for MIT systems, which is similar to the structure of many other systems.

The vocabulary sizes of spoken dialogue systems can vary significantly depending on the domain, ranging from hundreds of words for simple command and control interfaces, to tens or even hundreds of thousands of words for applications which involve recognizing items from a list (e.g., call routing [7, 9, 31], stock quotes [5, 42]). Many mixed-initiative systems, such as JUPITER, have vocabularies on the order of thousands of words. It is interesting to observe that vocabulary growth rates (unique words vs. corpus size) for spoken dialogue systems are usually significantly lower than for other types of recognition tasks such as transcription [28]. This is because current dialogue systems operate in a constrained domain, which tends to naturally limit the range of queries from users, although out-of-domain queries will certainly occur.

2. RECENT PROGRESS

In the past decade there has been increasing activity in the the area of spoken dialogue systems, largely due to government funding in the U.S. and Europe. By the late 1980's the DARPA spoken language systems program was initiated in the U.S., focusing on understanding speech input. The research groups which took part each developed an understanding system for an air travel information service (ATIS) domain, undertook a joint data collection effort, and underwent regular common evaluations (e.g., [6, 24, 13, 65]). Since there was no common ground established for evaluating understanding based on meaning representation, systems were evaluated based on their answers from a static database [37]. Dialogue was not emphasized in this program, since different dialogue strategies from different research groups would generate different user responses, and there was no agreed upon mechanism to evaluate it across sites.

In the past year a new DARPA program has begun which emphasizes dialogue-based interfaces incorporating both speech input and output technologies. One of the properties of this program is that participants are using a common system architecture to encourage component sharing across sites [57]. Participants in this program are developing both their own dialogue domains, and a common complex travel task (e.g., [20]).

In Europe, there have been several large research programs which encouraged research in spoken dialogue systems. The Esprit SUNDIAL (Speech Understanding and DIALog) program sponsored research in four languages (English, French, German, and Italian) [46]. Participants developed systems for flight and train schedule information. This program promoted co-operation by requiring different sites to contribute components to a single multi-site system. In the more recent ARISE (Automatic Railway Information Systems for Europe) program, participants developed train timetable information systems covering three different languages (Dutch, French, and Italian) [16]. Groups explored alternative dialogue strategies, and investigated different technology issues. Four prototypes underwent substantial testing and evaluation (e.g., [12, 34, 54]).

In addition to the research sponsored by these programs, there have been many other independent initiatives as well. For example, the Office Manager system developed at CMU provides voice access to application programs for the office of the future [52]. The Berkeley Restaurant Project (BeRP) provided restaurant information in the Berkeley, California area [30]. The AT&T AutoRes system allowed users to make rental car reservations over the phone via a toll-free number [38]. Their "How may I help you?" system provides call routing services and information [27]. The WAXHOLM system provides ferry timetables and tourist information for the Stockholm archipelago [8]. In the past few years, several spoken dialogue systems have been commercially deployed in domains such as call routing, stock quotes, train schedules, and flight reservations (e.g., [4, 5, 42]).

Another active area of research for spoken dialogue is conversational agents, which typically interact with a user to plan some task, which the agent is then able to carry out independently (e.g., "call me when that flight is half an hour from the airport."). These systems are likely candidates to incorporate multi-modal inputs and outputs (e.g., [1, 11, 63]).

Customer			Agent	
Act	Freq.	Words	Freq.	Words
Acknowledge	47.9	2.3	30.8	3.1
Request	29.5	9.0	15.0	12.3
Confirm	13.1	5.3	11.3	6.4
Inform	5.9	7.9	27.8	12.7
Statement	3.4	6.9	15.0	6.7

Table 1: Statistics of human-human conversations in a movie domain (from Flammia [21]). Annotated dialogue acts are sorted by customer usage, and include frequency of occurrence and average word length.

3. DIALOGUE DESIGN

When creating a spoken dialogue system, one of the most basic design decisions is the structure of the dialogue manager itself. This section discusses some of the issues that can be considered during the design process.

3.1. Studying Human Conversations

An open question for designers is how much to model human-machine dialogue after observed human-human communication. The study of human-human dialogue is an active area of research itself; there have been many large corpora collected and analyzed (e.g., [1, 2, 21]). Human conversations contain large numbers of phenomena such as disfluencies, interruptions, confirmations, anaphora, and ellipsis. Many utterances simply cannot be properly understood except in the context in which they occurred, and with knowledge of the domain. Table 1 shows statistics of annotated dialogue acts computed from human-human conversations in a movie information domain [21]. These statistics show that nearly half of the customers' dialogue turns were acknowledgements (e.g., "okay," "alright," "uh-huh"). [1] They also show that customer queries were not especially long. In a study of 100 hours of conversation from more than 1000 interactions between customers and agents for several different information tasks, Flammia found that over 80% of user utterances contained fewer than 12 words, with over half being 4 words or less [21].

While it is clear that the study of human-human conversations can provide valuable insights into the nature of dialogue, it is still a matter of debate how "human-like" spoken dialogue systems should be. The ability to handle phenomena commonly used in human conversations could ultimately make systems more natural and easy to use by humans, but they also have the potential to make things more complex and confusing. This is part of the reason why there is a great diversity in dialogue strategies deployed by different systems.

[1] An average dialogue consisted of over 28 turns between the customer and the agent.

3.2. Modeling Dialogue Phenomena

Many human dialogue phenomena are being successfully modeled by some systems, however. Most systems must deal with spontaneous speech artifacts such as filled pauses and partial words, especially those which allow users more flexibility in expressing their queries. Systems with some natural language capability must deal with discourse phenomena such as anaphora and ellipsis, and must be capable of processing ungrammatical queries, which are common in spontaneous speech.

Many systems are also able to handle interruptions, by allowing for the user to "barge in" over the system output (e.g., [5, 42, 53]). In addition to the problem of being reliably able to detect barge-in (which can quickly degrade both a human-machine and human-human conversation if not done properly!), it becomes necessary to properly update the dialogue status to reflect the fact that barge-in occurred. The system must take into account where the interruption occurred during its response. For example, if the system was reading a list of flights, the system might need to remember where the interruption occurred - especially if the interruption was under-specified (e.g., "I'll take the United flight," "Tell me about that one").

Researchers are also beginning to study the addition of back-channel communication in spoken dialogue responses, in order to make the interaction more natural. Prosodic information from fundamental frequency and duration appear to provide important clues as to when back-channelling might occur [40, 65]. Intermediate feedback from the system can also be more informative to the user than silence or idle music when inevitable delays occur in the dialogue (e.g., "Hold on while I look for the cheapest price for your flight to London...").

3.3. Matching Expectations with Capabilities

One of the more difficult problems in creating spoken dialogue systems is accurately conveying the system capabilities to the user. This includes both the type of speech input which can be processed by the system, and the domain of knowledge of the system itself. Expert users are familiar with at least a subset of the system capabilities, but this is not the case for novices, who can have considerable difficulty if their expectations are not well-matched with the system capabilities. Machine-directed dialogues tend to avoid these

problems altogether by leading the user through a series of questions, typically worded to produce a short answer. These systems also can take advantage of the fact that many users are familiar with the use of IVR systems, and thus already have a good mental model of the system's input and output behavior.

The same is not true of more mixed-initiative dialogue systems, which provide more freedom to the user. Users often do not understand the scope of the domain. For example, our JUPITER system knows only short-term weather forecasts, yet users ask a wide-variety of legitimate weather questions (e.g., "What's the average rainfall in Guatemala in January?" or, "When is high tide tomorrow?") which are outside the system's capabilities (they also ask a wide variety of non-weather queries, but that is a slightly different issue). In order to assist users to stay within the capabilities of the domain, some form of "help" capability is required. However, designing help capabilities is not an easy task. Users do not know how to ask for help, so identifying help requests is a spoken language understanding task on its own. In addition, the user may not understand the help (e.g., "Sorry, I didn't understand you, please rephrase your query"), especially if they do not understand why the system was not working for them in the first place. In our experience, people often tend to mimic whatever example suggestions are given. For example, we have observed that whenever an article appears about our JUPITER system with a sample dialogue, people will call in and try the exact same dialogue. Clearly the power of suggestion applies to spoken dialogue systems!

In addition to not knowing the domain of knowledge of the system, the user does not necessarily know the range of knowledge within the domain. For example, JUPITER does not know all 23,000 cities in the United States, so it is necessary to be able to detect when a user is asking for an out-of-vocabulary city, and then help inform the user what cities the system knows without listing all possibilities. JUPITER currently has a hierarchical geography table which tries to help the user narrow down a question to a specific city that it knows.

Finally, even if the user knows the full range of capabilities of the system, they may not know what type of questions the system is able to understand. Thus, we have observed many different speaking styles in our data which range from using isolated words, (e.g., "San Francisco"), to speaking cryptically (e.g., "temperature, Boston"), to being extremely chatty (e.g., "hi Jupiter, I'm taking a vacation to Hawaii next week and ...").

3.4. Recovering from Errors

Another challenging area of research is recovering from the inevitable mis-understandings that a system will make. Errors could be due to many different phenomena (e.g., acoustics, speaking style, disfluencies, out-of-vocabulary words, parse coverage, or understanding gaps), and it can be difficult to figure out that there is a problem, determine what the problem is caused by, and convey an appropriate response to the user that will fix the problem.

Many systems incorporate some form of confidence scoring to try to identify problematic inputs (e.g., [5, 31]). The system can then try an alternative strategy to either help the user, or back off to a more directed dialogue and/or one that requires explicit confirmation. Based on our statistics with JUPITER however, we have found that when an utterance is rejected, it is highly likely that the next utterance will be rejected as well [44]. Thus, it appears that certain users have an unfortunate tendency to go into a rejection death spiral which can be hard to get out of! More precise feedback from the system about the type of error and possible solutions would help this situation.

Using confidence scoring to perform partial understanding allows for more refined corrective dialogue, (e.g., requesting re-input of only the uncertain areas). Partial understanding may also help in identifying out-of-vocabulary words.

3.5. Integrating Multiple Modalities

Spoken dialogue systems can behave quite differently depending on what input and output modalities are available to the user. In displayless environments such as the telephone, it might be necessary to tailor the dialogue so as not to overwhelm the user with information. When displays are available however, it may be more desirable to simply summarize the information to the user, and to show them a table or image etc. Similarly, the nature of the interaction will change if alternative input modalities, such as pen or gesture, are available to the user. Which modality is most effective will depend among other things on environment (e.g., classroom), user preference, and perhaps dialogue state [14].

3.6. Implementation Strategies

There are many ways dialogue managers have been implemented, and they cannot all be described here. Many systems use a type of scripting language as a general mechanism to describe dialogue flow (e.g., [10, 58, 60]). Other systems represent dialogue flow by a

graph of dialogue objects or modules (e.g., [5, 63]).

Another aspect of system implementation is whether or not the active vocabulary or understanding capabilities change depending on the state of the dialogue. Some systems are structured so that a user can ask any question at any point in the dialogue, so that the entire vocabulary is active at all times. Other systems restrict the vocabulary and/or language which can be accepted at particular points in the dialogue. The trade-off is generally one of increased user flexibility (in reacting to a system response or query), and one of increased accuracy, due to the constraints on the user input.

In most current dialogue systems, the design of the dialogue strategy is typically hand-crafted by the system developer. This can be a time-consuming process, especially for mixed-initiative dialogues, whose result may or may not generalize to different domains. There is some research however, exploring the use of machine learning techniques to automatically determine dialogue strategy [35].

4. SPOKEN LANGUAGE TECHNOLOGIES

In addition to the problem of creating a dialogue manager, researchers are faced with many challenges in other spoken language technologies. Aside from an overall constraint of real-time performance which applies to the system as an ensemble, there are many interesting research challenges for individual component technologies. In this section we describe some of the research issues in more detail.

4.1. Speech Recognition

Many of the issues for speech recognition in the context of spoken dialogue can be found in other ASR research areas as well. In addition to the challenges associated with spontaneous speech artifacts which have been mentioned previously, some of the other problems for spoken dialogue systems are listed here.

4.1.1. Channel Conditions

The acoustic input conditions naturally depend on the setup of the particular dialogue system. In recent years however, many systems have moved from using close-talking, noise-canceling microphones, to telephones, or possibly microphone arrays. Telephones provide access to a much wider audience, and provide a more challenging acoustic environment due to narrower channel bandwidths, weaker S/N levels, and greater variety in handset characteristics. Cellular phones provide an even greater challenge due to added noise levels, drop-out, and other artifacts.

4.1.2. Speaker Variation

When we made our systems publicly available for general data collection, we started to observe a much wider variety of dialects and non-native accents than we had previously been able to collect. The difficulty of these data varied widely depending on the strength of the accent, and present a challenge to speaker-independent recognition technology [36].

Another type of speech we had not seen before in large quantities was data from children. These speakers were often considerably harder because the system had not been trained on large amounts of their data previously, and because the telephone channel bandwidths eliminated a significant portion of their speech. Despite the higher error rates, children were often fascinated by the technology and were very patient with the system - although they tended to just repeat their query verbatim rather than rephrasing.

4.1.3. Adaptation

Adaptation is widely used to improve the performance on individual speakers (e.g., [19]). Traditional adaptation techniques require a lot of data however, and so are not well suited to domains where there are a small number of queries, and there is no user identification mechanism. Thus, very short-term adaptation will be important for these areas. Applications where the user identity is known however, would be able to make use of some form of user profile, including acoustic-phonetic characteristics, as well as pronunciation, vocabulary, language, and possibly domain preferences (e.g., user lives in Boston, prefers aisle seat).

4.1.4. Adding New Words

In unconstrained input situations the vocabulary is not specified to the user, so users are free to use words unknown to the system. In systems such as JUPITER, which does not recognize all possible cities in the world, users will sometimes try to help out the system by spelling the word (e.g., "I said B A N G O R Bangor"), or emphasizing the syllables in the word (which usually has poor consequences!). In addition to detecting the presence of out-of-vocabulary words, it would also be useful to be able to add new words (especially content words) to dynamically augment system capabilities (e.g., [3]).

4.2. Language Understanding

Once a dialogue system gets beyond isolated word inputs, it becomes necessary to have some capability to

process word sequences so that they can be appropriately used by the system. There have been a large variety of methods explored in this area, both as to how linguistic constraints should be modelled, and how they should be integrated into the search.

Some systems drive the recognizer with a formal grammar (e.g., context-free). These systems have the potential disadvantage of being overly constraining however. More flexible frameworks do not try to fully analyze an utterance, but resort to keyword and phrase spotting methods (e.g., [4, 65]). This approach was highly successful when used during the DARPA ATIS program, and has been deployed successfully in commercial applications. Another hybrid approach, such as that adopted at MIT, is to try to perform a complete analysis of the utterance, but to back off to robust parsing when no complete parse is found [56]. In all cases, some form of stochastic parsing is very useful to provide constraint, in addition to being a platform for understanding.

At this point it is not clear whether or not a complete linguistic analysis is necessary. Certainly for simple domains such as JUPITER, it would be quite possible to spot keywords or phrases. As the language gets more complex however, it is not clear whether the simpler approach will always be effective.

Another area which has been explored to some extent is to incorporate automatic learning methods into understanding [47, 39, 45]. These methods require an annotated corpus however, which can be a time-consuming task.

There has been much research exploring how linguistic constraints can be incorporated into the search process. The simplest form of integration strategy is to decouple the two processes completely, whereby the ASR computes the top-choice sentence hypothesis and passes it to the NLU component for subsequent analysis. Currently, many systems compute a set of N-best sentence hypotheses, or a word graph, which gets processed by the NLU unit. Researchers have also explored more tightly-coupled strategies, although these are less commonly used in systems with more complex grammars due to increased computation. In general, an ongoing research challenge is to incorporate as much linguistic constraint early into the search, without introducing search errors, or increasing computation time much beyond real-time constraints.

4.3. Language Generation

Once an utterance has been understood by the system, it becomes necessary to convey information to the user. For systems with a display, it might be satisfactory to simply display a table of information. In our experience however, some form of linguistic feedback is extremely valuable as it informs the user what the machine understood. As such, it will be easier to detect that the machine made an error, and possibly to correct it. The linguistic feedback is usually a terse form of summary and often complements the tabular information.

The generation aspects are far more difficult when the system has to present all of its information to the user via speech. For example, it cannot speak about a long list of flights in detail without overwhelming the user with details, and therefore a significant part of the generation modelling has to do with deciding how to summarize a set of items in a list. Thus, there are interesting issues in deciding how much to convey new vs. old information, etc.

Many researchers have observed that the precise wording of the response can have a large impact on the user response. In general, the more vaguely worded response will result in the larger variation of inputs [5, 50]. Which type of response is more desirable will perhaps depend on whether the system is used for research or production purposes. If the final objective is to improve understanding of a wider variety of input, then a more general response might be more appropriate. A more directed response, however, would most likely improve performance in the short-term.

The language generation used by most spoken dialogue systems tends to be static, using a constant response pattern with users. We have observed that introducing variation in the way we prompt users for additional queries (e.g., "Is there anything else you'd like to know?" "Can I help you with anything else?", "What else?") is quite effective in making the system appear less robotic and more natural to users. It would be interesting to see if a more stochastic language generation capability would be well received by users. In addition, the ability to vary the prosody of the output (e.g., apply contrastive stress to certain words) also becomes important in reducing the monotony and unnaturalness of speech responses.

A more philosophical question for language generation is whether or not to personify the system in its responses to users. Naturally, there are varied opinions on this matter. In many situations we have found that an effective response is one commonly used in human-human interaction (e.g., "I'm sorry"). Certainly, users do not seem to be bothered by the personification evident in our deployed systems.

4.4. Speech Synthesis

Depending on the nature of the feedback, systems will use some form of speech synthesis for the user - especially for telephone-based systems. The type of syn-

thesis varies widely; from pre-recorded phrases, to word and phrase concatenation, to general-purpose synthesizers. In general, as the synthesis method becomes more general purpose, naturalness degrades. It is interesting to observe that the speech synthesis component is the one that often leaves the most lasting impression on users - especially when it is not especially natural. As such, more natural sounding speech synthesis will be an important research topic for spoken dialogue systems in the future.

4.5. Prosody

Although prosody impacts both speech understanding and speech generation, it has been most widely incorporated into text-to-speech systems. However, there have been attempts to make use of prosodic information for both recognition and understanding [29, 43, 55], and it is hopeful that more research will appear in this area in the future. In the Verbmobile project, researchers have been able to show considerable improvement in processing speed when integrating prosodic information into the search component during recognition [41].

5. DEVELOPMENT ISSUES

Developing spoken dialogue systems is a classic chicken and egg problem. In order to develop the system capabilities, one needs to have a large corpus of data for system refinement and evaluation. In order to collect data that reflects actual usage, one needs to have a system that users can speak to. Typically, developers create an initial system with a small corpus of data, perhaps using a wizard-of-oz data collection method, and then use a system-in-the-loop mechanism to iteratively refine and evaluate the system components.

5.1. Data Collection

One of the things that has evolved considerably over the last decade is the means and scale of data collection for system development and evaluation. This is true for both the speech recognition and speech understanding communities, and can be seen in many of the systems in the recent ARISE project [16], and elsewhere. At MIT for example, the VOYAGER system was developed by recruiting 100 subjects to come to our laboratory and ask a series of questions to an initial wizard-based system [22]. In contrast, the data collection procedure for the more recent JUPITER consists of deploying a publicly available system, and recording the interactions [66]. There are large differences in the number of queries, the number of users, and the range of issues which the data provide. By using a system-in-the-loop form of data collection, system development and evaluation become iterative procedures. If unsupervised methods were used to augment the system ASR and NLU capabilities, it could become continuous (e.g., [32]).

We have found making our systems widely available to real users has helped make our systems considerably more robust, and has provided a wealth of data for research in spoken dialogue systems. However, in order to get users to actually use the system, it needs to be providing "real" information to the user. Otherwise, there is little incentive for people to use the system other than to play around with it, or to solve toy problem scenarios which may or may not reflect problems of real users.

5.2. Evaluation

One of the issues which faces developers of spoken dialogue systems is how to evaluate progress, in order to determine if they have created a usable system. Developers must decide what metrics to use to evaluate their systems to ensure that progress is being made. Metrics can include component evaluations, but should also assess the overall performance of their system.

For systems which conduct a transaction, it is possible to tell whether or not a user has completed a task. In these cases, it is also possible to measure accompanying statistics such as the length of time to complete the task, the number of turns etc. It has been noted however, that such statistics may not be as important as user satisfaction (e.g., [51]). For example, a spoken dialogue interface may take longer than some alternative, yet users may prefer it due to other factors (less stressful, hands free, etc). A better form of evaluation might be a measure of whether users liked the system, whether they called to perform a real task (rather than browsing), and whether they would use it again, or recommend it to others.

Component evaluations for ASR and NLU are fairly common, but are less common for generation and synthesis since they are less amenable to automatic evaluation methods where it is necessary to decide what is a correct answer. ASR evaluation is usually the most straightforward although, even here, there are a range of phenomena which are not necessarily obvious how to evaluate (e.g., cross talk, mumbling, partial words). NLU evaluation can also be performed by comparing some form of meaning representation with a reference. The problem with NLU understanding is that there is no common meaning representation among different research sites, so cross-site comparison becomes difficult. In the DARPA ATIS program for example, the participants ultimately could agree only on comparing to

an answer coming from a common database. Unfortunately, this necessarily led to the creation of a large document defining principals of interpretation for all conceivable queries [37]. In order to keep the response across systems consistent, systems were restricted from taking the initiative, a major constraint on dialogue research.

One way to show progress for a particular system is to perform longitudinal evaluations for recognition and understanding. In the case of JUPITER, we continually evaluate on standard test sets, which we can redefine periodically in order to keep from tuning to a particular data set [49, 66]. Since data continually arrive, it is not difficult to create new sets and re-evaluate older system releases on these new data.

Some systems make use of dialogue context to provide constraint for recognition, for example, favoring candidate hypotheses that mention a date after the system has just asked for a date. Thus, any reprocessing of utterances in order to assess improvements in recognition or understanding performance at a later time need to be able to take advantage of the *same* dialogue context as was present in the original dialogue with the user. To do this, the dialogue context must be recorded at the time of data collection, and re-utilized in the subsequent off-line processing, in order to avoid giving the original system an unwarranted advantage.

5.3. Portability

Creating a robust, mixed-initiative dialogue system can require a tremendous amount of effort on the part of researchers. In order for this technology to ultimately be successful, it must be made easier port existing technology to new domains and languages. Over time, researchers have made the technology more modular. For example, in our original VOYAGER system, the discourse and language generation components were completely intertwined with the domain "back-end." Over time, we have taken language generation and discourse out of the back-end in order to make it more domain independent, and to make it easier to port to new domains. To date however, the dialogue manager is still very much domain dependent, and it is an ongoing challenge to make this component more domain independent.

Over the past few years, different research groups have been attempting to make it easier for non-experts to create new domains. Systems which modularize their dialogue manager try to take advantage of the fact that a dialogue can often be broken down into a smaller set of sub-dialogues (e.g., dates, addresses), in order to make it easier to construct dialogue for a new domain (e.g., [5, 63]. For example, researchers at OGI have developed rapid development kits for creating spoken dialogue systems which are freely available [63], and have been used by students to create their own systems [62]. Much more research is needed in this area if we are to try and make systems with complex dialogue strategies generalize to different domains.

6. CONCLUSIONS

Research in spoken dialogue systems has been increasing steadily over the last decade due to growing interest and demand for human-machine interaction. Many of spoken dialogue systems are now successfully deployed, in some cases for commercial applications, and tend to use data collected from real users for system development and evaluation. There are a wide range of dialogue strategies employed by these systems, ranging from tightly controlled machine-directed dialogue, to more flexible, and complex mixed-initiative dialogue. Despite increasing success, there remain a number of needed spoken language technologies which are necessary for improved interaction with users. As always, much work remains to be done to generalize the knowledge gained from experience with one domain, to many others.

7. ACKNOWLEDGEMENTS

Joe Polifroni and Stephanie Seneff made many suggestions which greatly improved the overall paper.

8. REFERENCES

[1] J. Allen, et al., "The TRAINS project: A Case Study in Defining a Conversational Planning Agent," *J. Experimental and Theoretical AI,* 7, 7–48, 1995.

[2] A. Anderson et al., "The HCRC Map Task Corpus," *Language and Speech,* 34(4), 351–366, 1992.

[3] A. Asadi, R. Schwartz, and J. Makhoul, "Automatic Modelling for Adding New Words to a Large Vocabulary Continuous Speech Recognition System," *Proc. ICASSP*, 305–308, 1991.

[4] H. Aust, M. Oerder, F. Seide, and V. Steinbiss, "The Philips Automatic Train Timetable Information System," *Speech Communication,* 17, 249–262, 1995.

[5] E. Barnard, A. Halberstadt, C. Kotelly, and M. Phillips, "A Consistent Approach to Designing Spoken-Dialog Systems," *these proceedings*, 1999.

[6] M. Bates, R. Bobrow, P. Fung, R. Ingria, F. Kubala, J. Makhoul, L. Nguyen, R. Schwartz, and D. Stallard, "The BBN/HARC Spoken Language Understanding System," *Proc. ICASSP,* 111–114, 1993.

[7] R. Billi, R. Canavesio, and C. Rullent, "Automation of Telecom Italia Directory Assistance Service: Field Trial Results," *Proc. IVTTA,* 11–16, 1998.

[8] M. Blomberg, R. Carlson, K. Elenius, B. Granstrom, J. Gustafson, S. Hunnicutt, R. Lindell, and L. Neovius, "An Experimental Dialogue System: Waxholm," *Proc. Eurospeech*, 1867-1870, 1993.

[9] B. Buntschuh et al., "VPQ: A Spoken Language Interface to Large Scale Directory Information," *Proc. ICSLP,* 2863–2866, 1998.

[10] R. Carlson and S. Hunnicutt, "Generic and Domain-Specific Aspects of the Waxholm NLP and Dialogue Modules," *Proc. ICSLP*, 677-680, 1996.

[11] J. Cassell, "Embodied Conversation: Integrating Face and Gesture into Automatic Spoken Dialogue Systems," *to appear, Spoken Dialogue Systems,* Luperfoy (ed.), MIT Press.

[12] G. Castagnieri, P. Baggia, and M. Danieli, "Field Trials of the Italian ARISE Train Timetable System," *Proc. IVTTA,* 97–102, 1998.

[13] M. Cohen, Z. Rivlin, and H. Bratt, "Speech Recognition in the ATIS Domain using Multiple Knowledge Sources," *Proc. DARPA Spoken Language Systems Technology Workshop,* 257–260, 1995.

[14] P. Cohen, M. Johnson, D. McGee, S. Oviatt, J. Clow, and I. Smith, "The Efficiency of Multimodal Interaction: A Case Study," *Proc. ICSLP,* 249–252, 1998.

[15] P. Constantinides, S. Hansma, and A. Rudnicky, "A Schema-based Approach to Dialog Control," *Proc. ICSLP,* 409–412, 1998.

[16] E. den Os, L. Boves, L. Lamel, and P. Baggia, "Overview of the ARISE project," *Proc. Eurospeech,* 1527–1530, 1999.

[17] M. Denecke, and A. Waibel, "Dialogue Strategies Guiding Users to Their Communicative Goals," *Proc. Eurospeech,* 2227–2230, 1997.

[18] L. Devillers, and H. Bonneau-Maynard, "Evaluation of Dialog Strategies for a Tourist Information Retrieval System," *Proc. ICSLP,* 1187–1190, 1998.

[19] V. Digilakis et al., "Rapid Speech Recognizer Adaptation to New Speakers," *Proc. ICASSP,* 765–768, 1999.

[20] M. Eskenazi, A. Rudnicky, K. Gregory, P. Constantinides, R. Brennan, C. Bennett, and J. Allen, "Data Collection and Processing in the Carnegie Mellon Communicator," *Proc. Eurospeech,* 2695–2698, 1999.

[21] G. Flammia, "Discourse Segmentation of Spoken Dialogue: An Empirical Approach," Ph.D. Thesis, MIT, 1998.

[22] J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue, "Multilingual Spoken-Language Understanding in the MIT Voyager System," *Speech Communication*, 17, 1–18, 1995.

[23] J. Glass, J. Polifroni, and S. Seneff, "Multilingual Language Generation across Multiple Domains," *Proc. ICSLP*, 983–976, 1994.

[24] J. Glass, et al., "The MIT ATIS System: December 1994 Progress Report," *Proc. DARPA Spoken Language Systems Technology Workshop,* 252–256, 1995.

[25] D. Goddeau, "Using Probabilistic Shift-Reduce Parsing in Speech Recognition Systems," *Proc. ICSLP*, 321–324, 1992.

[26] D. Goddeau, H. Meng, J. Polifroni, S. Seneff, and S. Busayapongchai, "A Form-Based Dialogue Manager for Spoken Language Applications," *Proc. ICSLP,* 701–704, 1996.

[27] A. Gorin, G. Riccardi, and J. Wright, "How may I help you?," *Speech Communication,* 23, 113–127, 1997.

[28] L. Hetherington and V. Zue, "New words: Implications for Continuous Speech Recognition," *Proc. Eurospeech*, 475-931, 1991.

[29] J. Hirschberg, "Communication and Prosody: Functional Aspects of Prosody," *Proc. ESCA Workshop on Dialogue and Prosody,* 7–15, 1999.

[30] D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stolcke, E. Fosler, and N. Morgan, "The Berkeley Restaurant Project," Proc. ICSLP, 2139-2142, 1994.

[31] A. Kellner, B. Rueber, and H. Schramm, "Using Combined Decisions and Confidence Measures for Name Recognition in Automatic Directory Assistance Systems," *Proc. ICSLP,* 2859–2862, 1998.

[32] T. Kemp, and A. Waibel, "Unsupervised Training of a Speech Recognizer: Recent Experiments," *Proc. Eurospeech,* 2725–2728, 1999.

[33] L. Lamel, S. Bennacef, J.L. Gauvain, H. Dartigues, and J. Temem, "User Evaluation of the Mask Kiosk," *Proc. ICSLP,* 2875–2878, 1998.

[34] L. Lamel, S. Rosset, J.L. Gauvain, S. Bennacef, M. Garnier-Rizet, and B. Prouts, "The LIMSI ARISE System," *Proc. IVTTA*, 209–214, 1998.

[35] E. Levin, R. Pieraccini, and W. Eckert, "Using Markov Decision Process for Learning Dialogue Strategies," *Proc. ICASSP*, 201–204, 1998.

[36] K. Livescu, "Analysis and Modelling of Non-Native Speech for Automatic Speech Recognition," S.M. Thesis, MIT, 1999.

[37] L. Hirschman et al., "Multi-site Data Collection for a Spoken Language Corpus," *Proc. DARPA Workshop on Speech and Natural Language,* 7–14, 1992.

[38] S. Marcus, et al., "Prompt Constrained Natural Language – Evolving the Next Generation of Telephony Services," *Proc. ICSLP,* 857–860, 1996.

[39] S. Miller, R. Schwartz, R. Bobrow, and R. Ingria, "Statistical Language Processing Using Hidden Understanding Models," *Proc. ARPA Speech and Natural Language Workshop*, 278–282, 1994.

[40] H. Noguchi and Y. Den, "Prosody-Based Detection of the Context of Backchannel Responses," *Proc. ICSLP*, 487–490, 1998.

[41] E. Nöth, "On the Use of Prosody in Automatic Dialogue Understanding," *Proc. ESCA Workshop on Dialogue and Prosody,* 25–34, 1999.

[42] Nuance Communications, http://www.nuance.com

[43] M. Ostendorf, C. Wightman, and N. Veilleux, "Parse Scoring with Prosodic Information: An Analysis/Synthesis Approach," *Computer Speech & Language,* 7(3), 193–210, 1993.

[44] C. Pao, P. Schmid, and J. Glass, "Confidence Scoring for Speech Understanding Systems," *Proc. ICSLP,* 815–818, 1998.

[45] K. Papineni, S. Roukos, and R. Ward, "Maximum Likelihood and Discriminative Training of Direct Translation Models," *Proc. ICASSP,* 189–192, 1998.

[46] J. Peckham, "A New Generation of Spoken Dialogue Systems: Results and Lessons from the SUNDIAL Project," *Proc. Eurospeech*, 33-40, 1993.

[47] R. Pieraccini, and E. Levin, "Stochastic Representation of Semantic Structure for Speech Understanding," *Speech Communication,* 11, 283–288, 1992.

[48] R. Pieraccini, E. Levin, and W. Eckert, "AMICA: The AT&T mixed initiative conversational architecture," *Proc. Eurospeech,* 1875–1879, 1997.

[49] J. Polifroni, S. Seneff, J. Glass, and T. Hazen, "Evaluation Methodology for a Telephone-based Conversational System," *Proc. Int. Conf. on Lang. Resources and Evaluation,* 42–50, 1998.

[50] S. Rosset, S. Bennacef, and L. Lamel, "Design Strategies for Spoken Language Dialog Systems," *Proc. Eurospeech,* 1535–1538, 1999.

[51] A. Rudnicky, M. Sakamoto, and J. Polifroni, "Evaluating Spoken Language Interaction," *Proc. DARPA Speech and Natural Language Workshop,* 150–159, October, 1989.

[52] A. Rudnicky, J.M. Lunati, and A. Franz, "Spoken Language Recognition in an Office Management Domain," *Proc. ICASSP*, 829–832, 1991.

[53] A. Rudnicky, E. Thayer, P. Constantinides, C. Tchou, R. Shern, K. Lenzo, W. Xu, and A. Oh, "Creating Natural Dialogs in the Carnegie Mellon Communicator System," *Proc. Eurospeech,* 1531–1534, 1999.

[54] A. Sanderman, J. Sturm, E. den Os, L. Boves, and A. Cremers, "Evaluation of the Dutch Train Timetable Information System developed in the ARISE Project," *Proc. IVTTA,* 91–96, 1998.

[55] E. Shriberg et al., "Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech?," *Language and Speech,* 41, 439–447, 1998.

[56] S. Seneff, "Robust Parsing for Spoken Language Systems," *Proc. ICASSP*, 189–192, 1992.

[57] S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue, "GALAXY-II: A Reference Architecture for Conversational System Development," *Proc. ICSLP,* 931–934, 1998.

[58] S. Seneff, R. Lau, and J. Polifroni, "Organization, Communication, and Control in the GALAXY-II Conversational System,", *Proc. Eurospeech,* 1271–1274, 1999.

[59] S. Seneff and J. Polifroni, "A New Restaurant Guide Conversational System: Issues in Rapid Prototyping for Specialized Domain," *Proc. ICSLP*, 665–668, 1996.

[60] V. Souvignier, A. Kellner, B. Rueber, H. Schramm, and F. Seide, "The Thoughtful Elephant: Strategies for Spoken Dialogue Systems," *to appear, IEEE Trans. SAP.*

[61] J. Sturm, E. den Os, and L. Boves, "Dialogue Management in the Dutch ARISE Train Timetable Information System," *Proc. Eurospeech,* 1419–1422, 1999.

[62] S. Sutton, E. Kaiser, A. Cronk, and R. Cole, "Bringing Spoken Language Systems to the Classroom," *Proc Eurospeech*, 709–712, 1997.

[63] S. Sutton, et al., "Universal Speech Tools: The CSLU Toolkit," *Proc. ICSLP*, 3221–3224, 1998.

[64] N. Ward, "Using Prosodic Cues to Decide when to Produce Back-Channel Utterances," *Proc. ICSLP*, 1728–1731, 1996.

[65] W. Ward, and S. Issar, "Recent Improvements in the CMU Spoken Language Understanding System," *Proc. ARPA Human Language Technology Workshop*, 213–216, 1996.

[66] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen, and L. Hetherington, "Jupiter: A Telephone-Based Conversational Interface for Weather Information," *to appear, IEEE Trans. SAP.*