Flexible and Personalizable Mixed-Initiative Dialogue Systems

James Glass and Stephanie Seneff
Spoken Language Systems Group
Laboratory for Computer Science, MIT
Cambridge, MA, USA
{jrg,seneff}@sls.lcs.mit.edu

Abstract

This paper describes our vision for a future time when *end users* of mixed-initiative spoken dialogue systems will be able to dynamically configure the system to suit their personalized goals. We argue that spoken dialogue systems will only become a common utility in society once they can be reconfigured, essentially instantaneously, to support a new working vocabulary within a new domain or subdomain. For example, if a user is interested in restaurants in Seattle, the system would go off-line to gather information from resources such as the Web, and would infer from that knowledge an appropriate working vocabulary, language models, and dialogue control mechanism for a subsequent spoken conversation on this topic. In addition to painting this vision, the paper also discusses our recent research efforts directed towards the technology development necessary to realize this larger goal.

1 Introduction

Spoken dialogue systems are emerging as an effective means for humans to access information spaces through natural spoken interaction with computers. These systems are usually implemented with a static knowledge space, or one that is only augmented through manual intervention from the system developers. A significant enhancement to the usability of such systems would be the ability to automatically acquire new knowledge through interaction with its end users and its available knowledge resources. We believe, in fact, that the main barrier to wide acceptance of spoken dialogue systems is their current lack of flexibility and personalization.

Over the past decade, researchers in the Spoken Language Systems Group at MIT have been developing human language technologies for *mixed initiative* conversational systems, which are distinguished from the emerging deployed commercial systems in that the interaction is natural and flexible, modelled after the style of human-human dialogue (Zue and Glass, 2000). The development of the Galaxy Communicator architecture (Seneff et al., 1998) has greatly accelerated the pace at which we as experts can configure complex dialogue systems in a wide range of different domains. As the underlying technology components have matured, our research focus has evolved to include issues related to portability, modularity, and dynamic configurability of system components. We believe that the ability for naive system developers, and even end users, to reconfigure existing systems to manage their personal needs, will be crucial for the successful use of these technologies.

We see several different ways in which such flexible reconfiguration will become feasible in the near future. Perhaps most critical is the initial preparation of a new domain, where available on-line databases will be the catalyst for defining the vocabulary and language models of the domain, as well as the nature of the dialogue interaction needed to guide the user through the information space (Polifroni et al., 2003). However, the ability to dynamically reconfigure based on new information will also be extremely valuable. For instance, a hotel domain for the entire U.S. might initially restrict name recognition to the major chains such as Sheraton and Hyatt, since it would not be feasible to support all hotel names in the U.S. Once the user defines a specific geographic region, the system can enhance the level of detail of its working vocabulary, but particular to the user's narrower request, now supporting explicit mention of perhaps all hotels by name within that designated region. Users would be able to select hotels based on a number of attributes as guided by the hotel database, and to specify constraints in general world knowledge areas such as dates and prices. Users should be able to further personalize the system

User:	I'm interested in a restaurant in Boston's North End
Sys:	I know of 53 restaurants in the North End. Of these, 46 are Italian restaurants. 28 of them are located on either Hanover or Salem Street.
User:	Tell me about the ones on Hanover Street.
Sys:	There is one seafood restaurant on Hanover Street. The others are Italian.
User:	I'm interested in the Seafood restaurant.
Sys:	The Daily Catch is located on 323 Hanover Street. The phone number is 617-523-8567. The price range is between $12 and $18.

Figure 1: Illustration of a possible dialogue between a user and a system in a restaurant domain.

User:	Can you tell me the phone number of the Thaiku restaurant in Seattle?
Sys:	I may not know the name of the restaurant. Can you spell it for me?
User:	t h a i k u
Sys:	The phone number of Thaiku is 206-706-7807.

Figure 2: A sub-dialogue to enroll a new restaurant name.

by adding new words instantaneously to the working vocabulary via spoken dialogue. This might also include specifying the word's semantic class: "I want to add the name *John Doe* to my rolodex." When feasible, a user-specified named entity, such as a restaurant, would be verified against Web sources to improve the system's ability to understand their request.

In order for this vision to become a reality, a number of specific technology goals must be met. First and foremost, it is essential to develop tools that will enable rapid configuration of dialogue systems in new domains of knowledge, guided mainly from domain-dependent information sources. Our efforts in generic dialogue development represent a strong initiative toward that goal (Polifroni and Chung, 2002). Secondly, we need to be able to support incremental update of vocabularies and language models for speech recognition and understanding, in essentially instantaneous time (Schalkwyk et al., 2003; Seneff et al., 1998; Chung et al., 2003). This would allow great flexibility within a single dialogue where the user might ask about a named entity that is not yet known to the system. Third, while we can make use of a large lexical resource for pronunciation modeling, we must have available as well a high-performance letter-to-sound capability, integrating multiple knowledge sources such as a Web page, a spoken name, a spoken spelling of the name, and/or a key-padded name (Chung and Seneff, 2002). Fourth, we need to have intelligent knowledge acquisition systems, capable of populating a database from Web sources, and extracting and organizing key elements from the database (Polifroni et al., 2003).

These ideas can best be illustrated through a couple of example scenarios. In Figure 1, the user begins with a request for a restaurant in a neighborhood of Boston. The system then rapidly configures itself to support the appropriate sub-language, and is able to summarize lists of restaurants meeting the constraints of the user's subsequent queries, eventually leading to a unique selection.

For the scenario in Figure 2, the user has asked about the phone number for a restaurant they already know about. The system parses the name within a complete parse, but with a generic "unknown_word" as a stand-in for the restaurant name. It can at this point go to the Web and download a set of candidate restaurant names for Seattle, to form additional constraints on a solicited spelling. The integration of the spelling, the spoken pronunciation, and the Web listing, we argue, potentially provide enough constraint to solve the specific problem with high accuracy. The system can now retrieve the requested information from the Web.

2 Underlying Technologies

Over the past several years, we have been making advances on several fronts, directed toward the larger goal of the vision outlined above. In this section, we will highlight some of these, with pointers to the literature for an in-depth description.

SpeechBuilder: Over the past few years, we have been developing a set of utilities that would enable research results to be migrated directly into application development (Glass and Weinstein, 2001). Our goal is to enable natural, mixed-initiative interfaces similar to those now created manually by a relatively small group of expert developers. We make no distinction between the technology components of SpeechBuilder and those of our most sophisticated dialogue systems, such as the Mercury flight reservation domain (Seneff and Polifroni, 2000). SpeechBuilder employs a Web-based interface where developers type in the specifics of their domain, guided by forms and pull-down menus. Components such as recognition vocabulary, parse rules, and semantic mappings are created automatically from example sentences entered by the developer. In several recent short courses, naive developers have been able to implement a new domain and converse with it on the telephone in a matter of hours.

Language Modelling: Patchwork Grammars A serious limitation in today's technology to immediate deployment of a new system is the chicken-and-egg problem of the language model. System performance is critically tied to the quality of the statistical language model, which typically depends on large domain-dependent corpora that don't exist until the domain is actually deployed and widely used. We have initiated an effort to automatically induce a grammar for a new domain from related content of existing speech corpora for other domains combined with knowledge derived from the content provider for the new domain. For instance, our hotel domain can leverage from an existing auto classified domain to extract patterns for referring to prices, can induce a grammar for dates from a flight domain, and can make use of statistics of hotel counts to determine city probabilities. Parse rules for general sub-domains such as dates, times, and prices are organized into sub-grammars that are easily embedded into any application, along with libraries for converting the resulting meaning representations into a canonical format, such as “27SEP2003.”

Flexible Vocabulary: We have recently realized our goal of enabling users to automatically add a new word to an existing system through natural interaction with the system itself (Schalkwyk et al., 2003; Seneff et al., 1998; Chung et al., 2003; Chung and Seneff, 2002; Seneff et al., 2003). We have thus far applied this only to the enrollment of the user’s name as part of a personalization phase (Seneff et al., 1998; Chung et al., 2003), through a “speak and spell” mode. After confirmation, the system reconfigures itself to fully support the word such that it can now be understood in subsequent dialogue. A high quality sound-to-letter framework (Chung et al., 2003) and a new ability to automatically derive a class n-gram from an NL grammar have facilitated this process (Seneff et al., 2003). The recognizer update is currently implemented via full recompilation, which can take up to a minute of elapsed time, but efforts to support incremental recognizer updates (Schalkwyk et al., 2003) hold promise for essentially instantaneous new word addition.

Managing the Dialogue: One of the most time consuming aspect of dialogue system development today is the implementation of the dialogue manager. To reduce this development phase, we have been creating a set of domain-independent functions that can be specialized to a particular domain through passed parameters. These functions perform such tasks as checking a query for completeness, filtering the database results on user-specified constraints, or making decisions on fuzzy attributes such as “near” (Polifroni and Chung, 2002).

One common but important subgoal in dialogue planning is to generate a succinct description of a set of retrieved entries. Our recent research in this area has focused on organizing database retrievals into a summary meaning representation, by automatically clustering sets into natural groupings. In parallel, we are developing generation tools that will translate these summaries into fluent English. For instance, in the hotel domain, the result set is automatically partitioned into “cheap” or “expensive” differently depending upon the city. By basing such subjective categories on a content provider, we alleviate the burden of the system developer, while at the same time producing a more intelligent system.

3 Summary and Conclusions

While there is inadequate space here to properly cover such a large topic as flexible and rapidly reconfigurable mixed-initiative dialogue systems, we hope that we have managed to convey our long-term research goals adequately and to provide the excitement that we ourselves feel in our current efforts to turn this vision into a reality. In fact, important subgoals that we have had for many years, such as incremental vocabulary update, grammar development and training through recycled resources, and tools to enable rapid development of effective dialogue interaction, are now finally bearing fruit. We believe that this is a critical moment in the life of dialogue system research, and we anticipate exciting breakthroughs in the near future, leading to systems that are not only useful but also easy to use and accommodating, such that users will prefer them over alternative means of acquiring their information needs.

References

G. Chung and S. Seneff, “Integrating speech with keypad input for automatic entry of spelling and pronunciation of new words,” *Proc. ICSLP*, 2061–2064, Denver, CO, 2002.

G. Chung, S. Seneff, and C. Wang, “Automatic acquisition of names using speak and spell mode in spoken dialogue systems,” *Proc. HLT-NAACL ’03*, Edmonton, Canada, 2003.

J. Glass and E. Weinstein, “SPEECHBUILDER: Facilitating spoken dialogue system development,” *Proc. Eurospeech*, 1335–1338, Aalborg, Denmark, 2001.

J. Polifroni and G. Chung, “Promoting portability in dialogue management,” *Proc. ICSLP*, 2721–2724, Denver, CO, 2002.

J. Polifroni, G. Chung, and S. Seneff, “Towards automatic generation of mixed-initiative dialogue systems from web content,” *submitted to EUROSPEECH*, 2003.

J. Schalkwyk, L. Hetherington, and E. Story, “Speech recognition with dynamic grammars,” *submitted to EUROSPEECH*, 2003.

S. Seneff, G. Chung and C. Wang, “Empowering end users to personalize dialogue systems through spoken interaction,” *submitted to EUROSPEECH*, 2003.

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, Sydney, Australia, 1998.

S. Seneff and J. Polifroni, “Dialogue management in the MERCURY flight reservation system,” *Proc. ANLP-NAACL Satellite Workshop*, 1–6, Seattle, WA, 2000.

S. Seneff, C. Wang and T. J. Hazen, ”Automatic induction of N-gram language models from a natural language grammar,” *submitted to EUROSPEECH*, 2003.

V. Zue and J. Glass, “Conversational interfaces: Advances and challenges,’ *Proc. IEEE*, 88(8), 1166–1180, 2000.