Bridging Gaps for Spoken Dialog System Frameworks in Instructional Settings

Gina-Anne Levow
Department of Linguistics
University of Washington
Seattle, WA 98195 USA
levow@u.washington.edu

Abstract

Spoken dialog systems frameworks fill a crucial role in the spoken dialog systems community by providing resources to lower barriers to entry. However, different user groups have different requirements and expectations for such systems. Here, we consider the particular needs for spoken dialog systems toolkits within an instructional setting. We discuss the challenges for existing systems in meeting these needs and propose strategies to overcome them.

1 Introduction

A key need in the spoken dialog systems community is a spoken dialog system development framework. Such systems fulfill fundamental roles in lowering barriers to entry for development of spoken dialog systems, providing baseline systems for comparability, and supporting novel experimental extensions. There are many characteristics that are desirable for a shared spoken dialog system resource, including:

- Availability: Systems should be provided on an on-going basis, with continuing support, updates, and maintenance.
- Ease-of-use: Systems should be easy to use and provide an environment in which systems are easy to develop.
- Platform-independence: Systems be usable on a wide variety of architectures, if installed, or provided on an accessible platform, such as a website.
- Application access: Systems should provide a range of exemplar applications within the framework.
- Flexibility and extensibility: Systems should enable integration of diverse technology components and facilitate a wide range of experimental configurations.
- Robustness: Systems should enable state-of-the-art performance for diverse applications.
- Affordability: Systems should be free if possible, or provided at pricing that is not prohibitive for different user groups.

However, these systems also serve diverse groups of users, from senior research developers to students building their first spoken dialog systems. While these users share many requirements, their relative importance naturally varies. Research developers will likely place greater emphasis on system robustness, extensibility, and flexibility, for example to incorporate alternative speech recognizers, speech synthesizers, or dialog managers. Those using such systems in an instructional setting will place greater importance on ease-of-use, platform portability or independence, availability, affordability, and access to reference applications. Below, we will discuss some of the challenges for systems trying to meet these needs. Then we will describe two popular current solutions and how they satisfy the needs of these different groups. Lastly we will present some additional needs for spoken dialog systems frameworks to bridge gaps in dialog systems for instructional use.

A variety of systems have been developed that address many of these needs, but all suffer from signif-

NAACL-HLT 2012 Workshop on Future directions and needs in the Spoken Dialog Community: Tools and Data, pages 21–22,
Montréal, Canada, June 7, 2012.

icant limitations. Availability and affordability have posed some of the knottiest problems. For example, many of the Galaxy Communicator research systems, such as those by University of Colorado (Pellom et al., 2001), MIT, and CMU, were made available to the research community. However, many of the systems are no longer available, usable, or supported, as research groups have disbanded and systems architectures have changed. Maintaining systems over time requires group and community commitment, facilitated by an open-source framework. Other toolkits and frameworks have become problematic due to conflicts between availability and affordability. The long-popular CSLU toolkit (Sutton and Cole, 1997) has recently shifted to a commercial footing. Similarly, several industry platforms have provided free non-commercial VoiceXML hosting, as a simple spoken dialog development environment. However, at least one of these systems has recently shifted to a paid-only status. The environment changes rapidly. Of three freely available academic systems and five VoiceXML platforms listed in a 2009 survey (Jokinen and McTear, 2009), two have already gone to paid status as of late 2011.

Two frameworks have emerged in recent years as popular SDS frameworks: the Ravenclaw/Olympus framework (Bohus et al., 2007) and VoiceXML, hosted on one of the industrial platforms, such as Nuance's Cafe or Voxeo[1]. However, they do seem to address the needs of different user groups. Ravenclaw/Olympus has been more widely adopted in the research community: it is robust, flexible, extensible, open-source, provides diverse use cases, and has an active support and development community. In contrast, the VoiceXML platforms have proven popular in an instructional setting, as attested by the large number of online homework assignments employing VoiceXML. These VoiceXML frameworks offer very simple, easy-to-use environments that are largely platform-independent, include basic support and tutorials, and provide simple baseline applications. Given VoiceXML's extensive role in industry settings, they also provide an advantage in terms of direct practical experience for students and in terms of broad resources and support. In an instructional setting, Ravenclaw/Olympus' relative complexity, Windows platform and software dependence in instructional environments where linux has become predominant, and smaller resource base represent hurdles. While the VoiceXML platforms excel in these dimensions, their very simplicity and ease-of-use are limiting. Students are often looking for existing applications of moderate interesting complexity as a basis for extension and experimentation. Most typical example applications are simpler than those given for Olympus, and the platform is severely limiting for more advanced users and tasks. For example, many VoiceXML frameworks do not even support user-defined pronunciations. Lastly, these VoiceXML platforms rely on the generosity of the industrial teams, which can readily evaporate as has already happened with Tellme Studio.

Ideally, for instructional use, we would like to bridge the gap between the too-simple, restrictive VoiceXML frameworks and the more challenging but more flexible and powerful Ravenclaw/Olympus framework, to allow students and instructors to transition more smoothly from one to the other. On the VoiceXML side, a community-supported VoiceXML platform would reduce dependence on industry platforms. Access to VoiceXML applications of greater complexity, comparable to Let's Go! or Communicator tasks, would allow more interesting experiments within a course's limited span. Lastly, porting Ravenclaw/Olympus to linux would allow easier adoption in a wider range of academic programs.

[1] http://cafe.bevocal.com; http://www.voxeo.com

References

D. Bohus, Antoine Raux, Thomas K. Harris, Maxine Eskenazi, and Alexander I. Rudnicky. 2007. Olympus: an open-source framework for conversational spoken language interface research. In *Bridging the Gap: Academic and Industrial Research in Dialog Technology workshop at HLT/NAACL 2007*.

Kristiina Jokinen and Michael F. McTear. 2009. *Spoken Dialogue Systems*. Morgan & Claypool Publishers.

B. Pellom, W. Ward, J. Hansen, K. Hacioglu, J. Zhang, X. Yu, and S. Pradhan. 2001. University of Colorado dialog systems for travel and navigation.

Stephen Sutton and Ronald Cole. 1997. The cslu toolkit: rapid prototyping of spoken language systems. In *Proceedings of the 10th annual ACM symposium on User interface software and technology*, UIST '97, pages 85–86, New York, NY, USA. ACM.