

Building Practical Spoken Dialog Systems

Antoine Raux¹, Brian Langner², Alan W Black³, Maxine Eskenazi⁴

Language Technologies Institute

Carnegie Mellon University

Pittsburgh, PA, USA

{antoine, blangner, awb, max}@cs.cmu.edu

1 Abstract

This tutorial will give a practical description of the free software Carnegie Mellon Olympus 2 Spoken Dialog Architecture. Building real working dialog systems that are robust enough for the general public to use is difficult. Most frequently, the functionality of the conversations is severely limited - down to simple question-answer pairs. While off-the-shelf toolkits help the development of such simple systems, they do not support more advanced, natural dialogs nor do they offer the transparency and flexibility required by computational linguistic researchers. However, Olympus 2 offers a complete dialog system with automatic speech recognition (Sphinx) and synthesis (SAPI, Festival) and has been used, along with previous versions of Olympus, for teaching and research at Carnegie Mellon and elsewhere for some 5 years. Overall, a dozen dialog systems have been built using various versions of Olympus, handling tasks ranging from providing bus schedule information to guidance through maintenance procedures for complex machinery, to personal calendar management. In addition to simplifying the development of dialog systems, Olympus provides a transparent platform for teaching and conducting research on all aspects of dialog systems, including speech recognition and synthesis, natural language understanding and generation, and dialog and interaction management.

The tutorial will give a brief introduction to spoken dialog systems before going into detail

about how to create your own dialog system within Olympus 2, using the Let's Go bus information system as an example. Further, we will provide guidelines on how to use an actual deployed spoken dialog system such as Let's Go to validate research results in the real world. As a possible testbed for such research, we will describe Let's Go Lab, which provides access to both the Let's Go system and its genuine user population for research experiments.

2 Outline

Part 1

- 1.1 Introduction
- 1.2 Overview of current spoken dialog system architectures
- 1.3 Description of the Olympus2 dialog architecture
- 1.4 How to build an Olympus2 spoken dialog system

Part 2

- 2.1 Advanced Topics
 - a. Improving ASR
 - b. Improving TTS
 - c. Dealing with ASR Errors
 - d. Logs and Tools
- 2.2 Using Olympus2 for research and applications
- 2.3 Final summary

¹ <http://www.cs.cmu.edu/~antoine>

² <http://www.cs.cmu.edu/~blangner>

³ <http://www.cs.cmu.edu/~awb>

⁴ <http://www.cs.cmu.edu/~max>