DEMONSTRATION OF AT&T "LET'S GO": A PRODUCTION-GRADE STATISTICAL SPOKEN DIALOG SYSTEM

Jason D. Williams, Iker Arizmendi and Alistair Conkie

AT&T Labs – Research, Shannon Laboratory, 180 Park Ave., Florham Park, NJ 07932, USA

{jdw,iker,adc}@research.att.com

1. INTRODUCTION AND AIMS

This is a demonstration of the AT&T "Let's Go" bus timetable spoken dialog system. This system was entered in the 2010 Spoken Dialog Challenge [1], where the task is to provide bus timetable information for Pittsburgh, Pennsylvania.

Our primary aim in the challenge was to build a statistical spoken dialog system to comemrcial production standards, both in terms of *user interface*, and also in terms of *compatibility with commercial development practices*. AT&T Let's Go illustrates how this can be done, incorporating two statistical techniques: the AT&T Statistical Dialog Toolkit (ASDT) [2], which tracks a distribution over many dialog states in real time; and regression-based confidence scores, which are trained on a corpus of in-domain recognitions [3].

2. SYSTEM DESCRIPTION

System design and development largely followed common commercial practices. To start, we listened to about 100 calls with an existing system, and observed that most callers knew the bus line they needed, and usually wanted the next few buses rather than other buses in the future. We also observed that the audio quality was often poor, since many users were calling in noisy conditions, such as a bus stop.

Based on these observations, we decided to first ask the caller for the bus route (or say "I'm not sure"). When the route was known, route-specific language models are used when recognizing the origin and destination – a common approach in industry to maximizing recognition accuracy in noisy conditions. The system then asked if the caller would like times for the next few buses; if not, the user was asked for the date and (separately) time. Finally, the bus schedule that best matched the query was presented; users could navigate through the available times by saying "next", "previous", or "repeat". If there are repeated difficulties recognizing a bus stop, the system instead asks for the neighborhood.

This design was implemented using a simple state-based dialog control algorithm, similar to VoiceXML, consisting of about 25 dialog states. Each state requested or confirmed a particular piece of information (e.g., bus route). The dialog control implementation itself is relatively compact, consisting of about 1700 lines of Python code.

Dynamic content was rendered using the AT&T Natural Voices (TM) text-to-speech engine (TTS), and static prompts were recorded using the same voice talent used for the TTS. As in commercial systems, the prompt language was context-specific and carefully written – for example, the intonation used for asking questions the first and subsequent times was different. Since the system was active at night, special attention was paid to rendering times clearly – for example, "At twelve thirty AM earlier tonight...", "At twelve thirty AM later tonight...", and "At twelve thirty AM early in the morning of July thirtieth...". The pronunciations used by TTS for most stop names were checked and adjusted when necessary, consulting native Pittsburghers. Recognition was performed with the AT&T WATSON speech recognizer [4].

Two statistical techniques were incorporated. First, recognitions were scored using a regression-based confidence model [3], trained for each language model using a corpus of in-domain utterances provided to challenge participants. The regression model used 11 features in the regression, including score from garbage models, features of the lattice, and features of the word confusion network.

Second, a *belief state* (posterior distribution) was tracked over partitions of values for each of the 5 slots (bus route, origin, destination, date, and time), using the AT&T Statistical Dialog Toolkit (ASDT) [2]. The belief in a partition is the posterior probability of that partition containing the true user goal given priors over the user goals, a model of how the user behaves, and *all* of the system prompts and speech recognition results received over the entire dialog. The main benefit to tracking this distribution is better robustness to ASR errors, achieved by combining repeated low-confidence recognitions, synthesizing together ASR N-Best lists across multiple recognitions, and incorporating priors in each user goal. For example, in this system, priors for origin and destination were based on how many bus stop IDs the partition contained, so the prior of "Forbes Avenue" was higher than for "Forbes and Murray". Each belief state tracked a maximum of 15 partitions, and each update considered a maximum of 10 N-best list entries.

The belief state was used to decide when to reject ("Sorry,

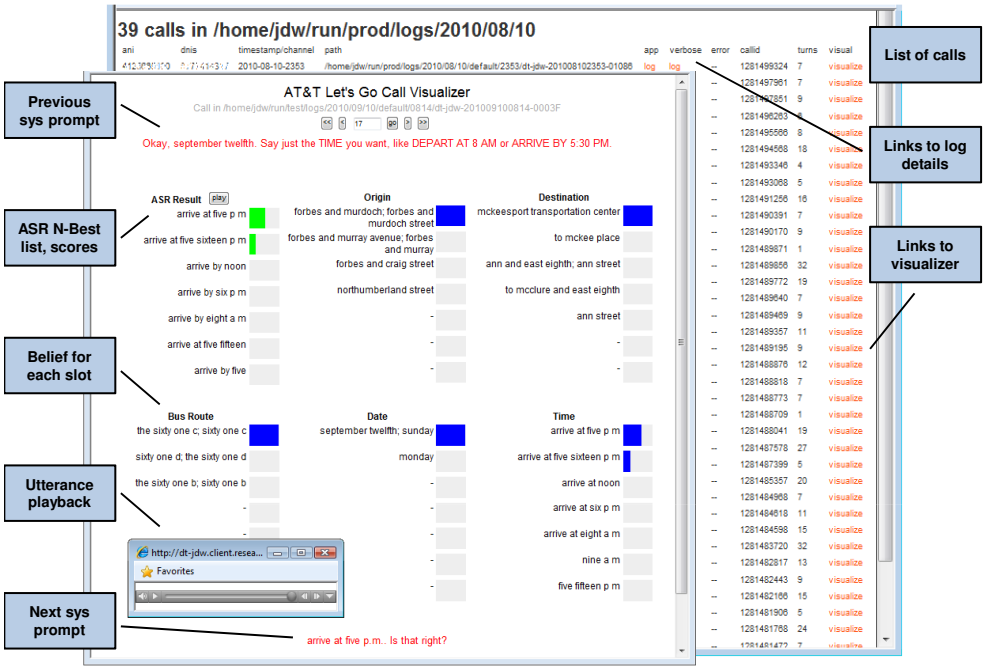

Fig. 1. Screenshot of the interactive call visualizer. The window in the background shows a database of all calls, with timestamps. Clicking on "visualize" shows the window in the foreground, which displays the system's output, ASR input, and the resulting belief state for each turn of the selected call. Clicking "play" plays the caller's utterance audio, in the inset window. The visualizer can also display calls currently in progress, in real-time.

where are you leaving from?"), explicitly confirm ("Leaving from McKeesport. Is that right?") or implicitly confirmed ("Ok, leaving from McKeesport. To change, say go back. Where are you going to?") a slot. Thresholds for each type of system action were set manually.

During the challenge, the system received approximately 100 calls from usability subjects, and 850 calls from real bus riders in Pittsburgh. As of the time of writing, transcriptions from real callers are not yet complete; in future work, we intend to report on the performance of the system.

3. DEMONSTRATION DESCRIPTION

The demonstration consists of two modes. First, a user can call the AT&T Let's Go dialog system, and attempt to complete a provided scenario, or a scenario of their own choosing. During the call, a real-time visualization is provided which shows the contents of the ASR N-Best list, the associated scores assigned by the regression model, and the belief state for all 5 slots. A poster which accompanies the demonstration includes a diagram of the state-based dialog design.

Second, a user can use the visualization to browse existing calls, including their own calls, and calls from users in the challenge. Similar to the real-time mode, the system action, ASR N-Best list, and belief state are shown at each turn of the call. In addition, callers' utterances can be played to check whether recognition was correct. Figure 1 shows a labeled screenshot.

By interacting with the demonstration, users can watch the operation of the regression-based confidence scores and belief state, and quickly gain an understanding of how these statistical techniques can be incorporated into a production-grade spoken dialog system.

4. REFERENCES

[1] AW Black, S Burger, B Langner, G Parent, and M Eskenazi, "Spoken dialog challenge 2010," in *Proc Workshop on Spoken Language Technologies (SLT), Spoken Dialog Challenge 2010 Special Session, Berkeley, CA*, 2010.

[2] JD Williams, *AT&T Statistical Dialog Toolkit*, 2010, `http://www.research.att.com/people/Williams_Jason_D`.

[3] JD Williams and S Balakrishnan, "Estimating probability of correctness for asr n-best lists," in *Proc SIGdial, London, UK*, 2009.

[4] V Goffin, C Allauzen, E Bocchieri, D Hakkani-Tur, A Ljolje, S Parthasarathy, M Rahim, G Riccardi, and M Saraclar, "The AT&T Watson speech recognizer," in *Proc ICASSP, Philadelphia*, 2005.