DeepListener: Harnessing Expected Utility to Guide Clarification Dialog in Spoken Language Systems

Eric Horvitz and Tim Paek

Microsoft Research
One Microsoft Way
Redmond, WA 98052 USA
{horvitz, timpaek}@microsoft.com

ABSTRACT

We describe research on endowing spoken language systems with the ability to consider the cost of misrecognition, and using that knowledge to guide clarification dialog about a user's intentions. Our approach relies on coupling utility-directed policies for dialog with the ongoing Bayesian fusion of evidence obtained from multiple utterances recognized during an interaction. After describing the methodology, we review the operation of a prototype system called DeepListener. DeepListener considers evidence gathered about utterances over time to make decisions about the optimal dialog strategy or real-world action to take given uncertainties about a user's intentions and the costs and benefits of different outcomes.

1. INTRODUCTION

Public enthusiasm for automated speech recognition (ASR) has been tempered by the common experience of frustrating and costly recognition errors. To enhance the performance of ASR, we have been exploring methods to leverage information about the stakes of real-world actions. The methods leverage knowledge about the probabilistic relationships between the output of the recognizer and the *intentions* of speakers, as well as a consideration of the *costs and benefits* of alternate actions taken under uncertainty. In this paper we focus on methods to guide and fuse the results of clarification dialog. This research on utility-directed clarification policies has been undertaken as part of the Conversational Architectures project at Microsoft Research, which seeks overall to the augment existing ASR systems with one or more layers of reflection [4][9][10][11]. These new layers facilitate context-sensitive decision-making about the intended target of an utterance, the intentions of the speaker and, ultimately, the best actions to execute, including both dialog and real-world actions.

Utility-directed clarification policies are especially useful in situations where only general purpose, untrained acoustic language models are available, and where we can expect poor or unreliable speech input. These situations are common in such popular applications as mobile telephony and desktop software. In these venues, inexpensive hardware and ambient noise can dramatically degrade the performance of speech recognition.

We first introduce the perspective of speech recognition as decision making under uncertainty. Then we describe representations of probabilistic relationships among adjacent utterances—including utterances that may be a response to system-initiated clarification. Finally, we describe the DeepListener system and review the clarification dialog policies employed in the system.

2. GOALS, UTTERANCES, AND UNCERTAINTY

Taking the perspective of speech recognition as decision making under uncertainty, we view dialog actions and real-world actions uniformly within an overarching expected-utility framework. The DeepListener system seeks to identify key uncertainties and to select actions with the highest expected utility given uncertainty about the intentions associated with an utterance. At the heart of the approach, we harness Bayesian graphical decision models called influence diagrams [6] and dynamic Bayesian networks [2][7]. Unlike recent work on the use of Markov decision processes in learning and modeling dialog [8], this work focuses on the use of rich, dynamic decision-making at each turn of the dialog, while performing inference about users' intentions from multiple acoustical signals over time. A spoken command and control system can refine its probability of a user's intention by fusing information from multiple utterances provided in a recognition session over time.

Figure 1 displays two time slices of a temporal probabilistic model, capturing key variables under consideration at two turns of an interaction. The ovals represent random variables and the arcs capture probabilistic dependencies among variables within and between time slices. Decision (square node) and value variables (diamond) in each time slice comprise a local decision problem that is used to identify local actions associated with the greatest expected utility, based on the inferred probability over a user's intentions. DeepListerner employs the model to reason about a user's spoken intentions, a key variable in the model. This variable has states representing the intention associated with different utterances or unrecognized acoustical information detected by the system.

As a testbed, we initially applied DeepListener to support the spoken command and control functionality of a software application called Lookout [3]. Lookout provides automated scheduling and calendar services, interoperating with the Microsoft Outlook system. As the application receives mail, it continually computes the likelihood that a user would like to receive assistance with accessing their calendar or scheduling a meeting based on the structure and content of the message at a user's focus of attention. The base speech capabilities of Lookout allow users to respond any way they would like to offers of assistance from Lookout. For Lookout and other command and control applications, DeepListener includes

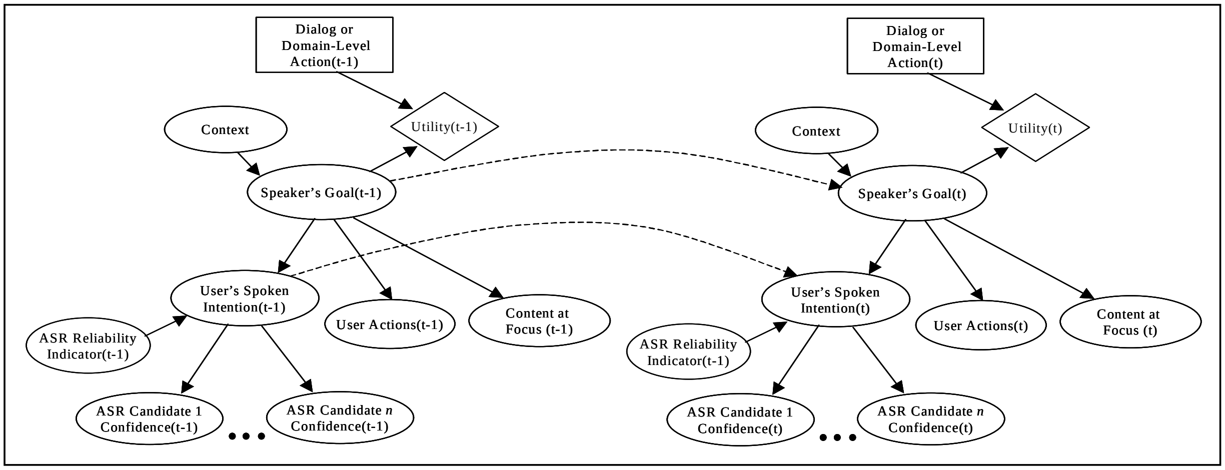

Dialog or Domain-Level Action(t-1)
Context
Utility(t-1)
Speaker's Goal(t-1)
User's Spoken Intention(t-1)
User Actions(t-1)
Content at Focus (t-1)
ASR Reliability Indicator(t-1)
ASR Candidate 1 Confidence(t-1)
• • •
ASR Candidate *n* Confidence(t-1)
Dialog or Domain-Level Action(t)
Context
Utility(t)
Speaker's Goal(t)
User's Spoken Intention(t)
User Actions(t)
Content at Focus (t)
ASR Reliability Indicator(t-1)
ASR Candidate 1 Confidence(t)
• • •
ASR Candidate *n* Confidence(t)

Figure 1: A temporal probabilistic model relating a user's intentions, utterances, and ideal actions over time.

intentions such as *acknowledgment* (user's spoken command was intended to communicate, "yes, do it"), *negation* ("no, don't do that"), *reflection* (responses like "hmm," "uh," etc. associated with a user's reflection about a desire for the service), *unrecognized signal* (system has heard an unrecognized noise or stream of words), and *no signal* (nothing is heard). At design time, developers define a set of candidate utterances the users might generate for each intention (e.g., "yes," "okay," "sure," "yeah," "alright," for acknowledgement).

The arcs in the dynamic network model indicate that a user's goals influence a user's spoken intentions, which in turn influence the likelihood that the speech engine will report different utterances. A variable labeled *context*, captures information that might be provided by an external user modeling system, such as Lookout's facility for assigning likelihoods to goals, given the text and user's focus of attention. As captured in the decision model, a user's goals directly influence actions that might be observed, including the content that a user is focusing on or creating. The dashed arcs between goals and intentions in adjacent time slices capture temporal dependencies between these variables. We also include a variable that captures potentially observed information about acoustical conditions that can influence the overall reliability of speech recognition, including levels of background noise. We assessed by hand the conditional probabilities encoded in a version of the user model which collapses the user's goals with their spoken intentions; for example, we assessed the likelihood of hearing different classes of response with specific values of confidence, given the intention represented by the user's actual utterance. The utilities about outcomes, captured in the value node, can be elicited from users through psychological experiments [9] or via graphical direct assessment tools provided in DeepListener. DeepListener deliberates about whether to ignore the detected utterance, respond to the signal with a relevant action, or engage in clarification dialog.

At run time, we observe evidence reported by the speech engine (recognized candidates and their confidence scores) and any user activity information and infer the likelihood over states of variables higher up in the model that we cannot observe directly, such as the user's intentions. The probability distribution over a user's intentions is used to compute the dialog action or real-world service that has the highest expected utility at each turn.

To highlight the ability of the temporal decision model to enhance the robustness of an interaction with potentially erroneous recognized speech, let us consider the typical use of DeepListener with an ASR system for command and control. A listening session is initiated by a prompt from the TTS system suggesting to the user that a service may be valuable in the current context. Following the initiation of a session, each recognition cycle, capturing a user's next turn in a dialog, is represented by a time slice in the model. At the end of each turn, the ASR system processes the acoustical signal and provides a set of recognized candidates and a measure of confidence for each candidate. A probability distribution over the spoken intentions of a user is then inferred from the utterances as well as other information that may be observed, including a user's recent interactions with a computer application. Inferring a user's goals, based on desktop or online activity and content, has been a recent focus of work on user modeling [1][5].

3. UTILITY-DIRECTED DECISIONS ABOUT DIALOG AND ACTION

DeepListener relies on Bayesian inference provided by components of the MSBNX Bayesian modeling and inference tool, developed at Microsoft Research. We coupled DeepListener with the basic command and control speech system standard in the Microsoft Agent package. Following the execution of a query to users by the Microsoft Agent's TTS system, the speech recognizer is activated and the acoustical signal that is detected within a time horizon is processed. To evaluate the utterance, a list of candidate commands and their confidence scores is retrieved by the system. A probability distribution is inferred over the classes of response represented by the processed acoustic signal and a local decision with maximum expected utility is identified. In one version of the system, actions include:

- Execute the service being offered
- Ask user to repeat the utterance

- Note a recognition of a user's hesitation and try again
- Note the hearing of a noise and inquire
- Try to get the user's attention
- Apologize for the interruption and forego the service
- Engage in troubleshooting about the overall dialog

We constructed a utility model by assessing the utility of different outcomes in the space of outcomes defined by the cross product of the classes of response under consideration and the actions listed above. To enhance the naturalness of interaction, we tailored the behaviors of an animated agent provided with the MS Agent package to create a set of appropriate utterances and gestures designed for each action.

4. DEEP LISTENER IN ACTION

DeepListener endows base-level ASR command and control system with additional flexibility by evaluating its listening and reacting in a manner consistent with both its current uncertainty about the intentions associated with one or more utterances and its preferences about outcomes. In typical situations of uncertainty, the system makes decisions in accordance with its understanding of the expected consequences of alternate actions. The experience of interacting with the system in noisy environments—or at a relatively long distance away from a microphone—appears to give users the impression of communicating with a person who is having difficulty hearing. A user can utter different words for acknowledging or accepting a service and can expect the system to have considered the entire recent history of interaction, rather than treating each utterance as an independent event. We are currently pursuing methodical tests and user studies to evaluate the behavior of DeepListener in different kinds of environments to see how well it performs in comparison with a more naive, low-level speech command and control system.

Figures 2 and 3 display graphs of the probabilities and expected utilities inferred over time from a sample interaction. For this interaction, the system was exposed to an ongoing background conversation that was paused briefly with a response of “yeah.” The prior probabilities of different intentions, shown in turn 0, are updated at turn 1. The most likely state of affairs at this time is *overheard*. The action with the maximum expected utility is the sharing of the inferred inference via a confused agent demonstrating its thinking, “…Was that meant for me?” appearing in a thought cloud. In the next turn, the user provides a muffled “yes.” The system recognizes “yes” with low confidence and a “yeah” with medium confidence, and updates all of its probabilities and expected utility calculations. Now, the most likely intention is “yes, perform the service.” However, given the utility model, the action with the highest expected utility is to ask the user to repeat the response. Following the receipt of a clarifying “sure,” DeepListener updates the probabilities and utilities of alternate actions and goes ahead and performs the service, as displayed in turn 3 of the session captured in Figure 3. Figure 4 displays the user experience at steps 2 and 3 of the interaction.

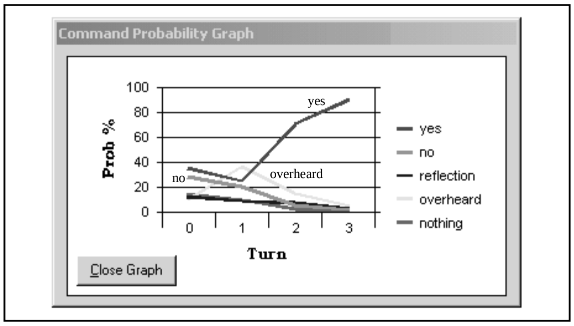


Figure 2: The inferred probability distribution over three steps of an interaction initiated by an offer of service and followed by clarification dialog.

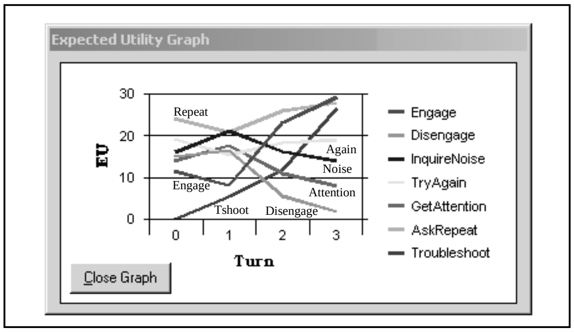
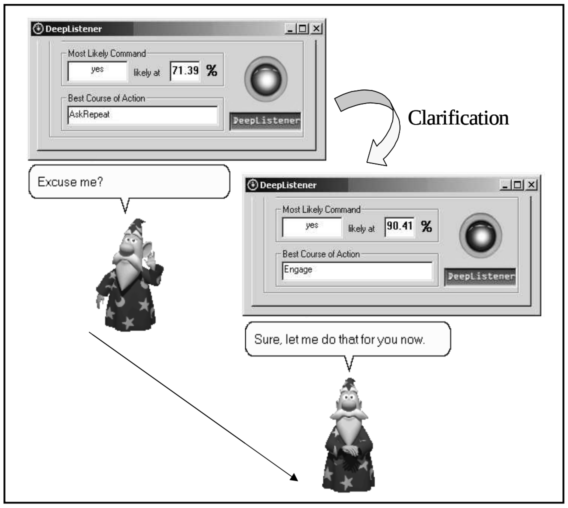


Figure 3: The expected utilities of different actions at each step computed from the probabilities displayed in Figure 2. DeepListener asks the user to repeat the intention in steps 1 and 2 and executes the service at step 3.


DeepListener

Most Likely Command: yes likely at 71.39 %

Best Course of Action: AskRepeat

DeepListener

Most Likely Command: yes likely at 90.41 %

Best Course of Action: Engage

Figure 4: The user experience at steps 2 and 3 of the sample case. A clarification dialog leads to a capture of additional evidence and execution of the scheduling service.

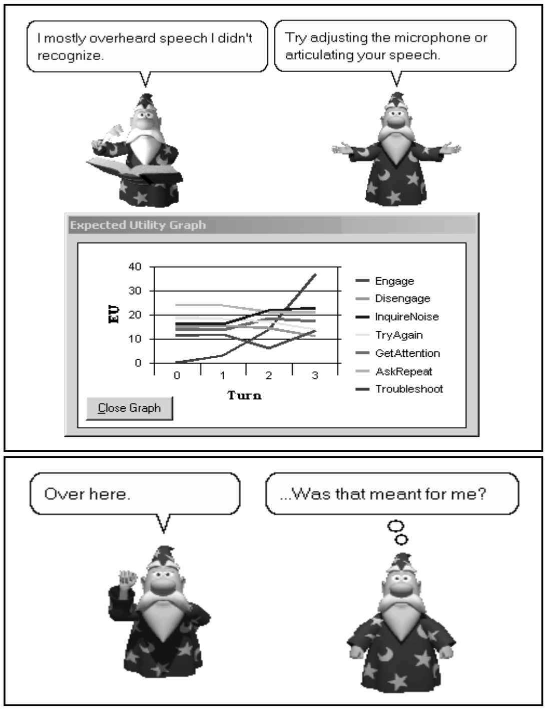


Figure 5: Other DeepListener behaviors. **Top**: At the third turn, the action with the greatest expected utility is troubleshooting the interaction. **Bottom**: DeepListener's behavior when it attempts to acquire the user's attention (left) and when it believes it is overhearing utterances targeted elsewhere (right).

Figure 5 displays a session where the expected value of troubleshooting dominates the other actions at step 3. In attempting to assist with troubleshooting, DeepListener provides a multi-step report, summarizing the history of the system's probabilistic inferences about the user's intentions during the session. In the lower portion of Figure 5, we display the system's behavior in cases where the system decides that it should acquire the user's attention and where the system believes it is overhearing utterances directed elsewhere.

We have generalized DeepListener into a development tool that can be used for multiple command and control domains. A set of preference assessment tools enable system builders or end users to assess utilities of real-world domain actions given each of the user's potential intentions. Using an ActiveX interface, an external system (such as Lookout) can supply probabilities about a user's goals based on its own analysis or observations. This interface is intended to allow an external user model to influence the prior probabilities of different user goals in a dynamic fashion.

6. SUMMARY

We have described principles for coupling spoken command and control systems with temporal probabilistic decision models that consider the costs and benefits of alternate actions. The approach centers on inferring key probabilities by pooling information gathered during one or more adjacent attempts to communicate with a system, and computing the expected utility of alternate real world and dialog actions. We presented an implementation of the approach in the DeepListener system. We believe that that the principles embodied in DeepListener, for guiding clarification dialog in a selective, context-sensitive manner, can be harnessed to fundamentally change the qualitative experience of interacting with spoken language systems.

REFERENCES

1. Conati, C., Gertner, A., VanLehn, K., and Druzdzel, M., 1997. Online student modeling for coached problem solving using Bayesian networks. *Proc. of the Sixth International Conference on User Modeling*, 231-242. Springer-Verlag.
2. Dagum, P., Galper, A., & Horvitz, E. 1992. Dynamic network models for forecasting. In *Proceedings of the Eighth Workshop on Uncertainty in Artificial Intelligence*, 41-48.
3. Horvitz, E. 1999. Principles of mixed-initiative user interfaces. *Proc. of CHI '99, ACM SIGCHI Conference on Human Factors in Computing Systems*, Pittsburgh, PA, May 1999, 159-166. ACM Press.
4. Horvitz, E. & Paek, T. 1999. A computational architecture for conversation. *Proc. of the Seventh International Conference on User Modeling*, 201-210. Springer Wien.
5. Horvitz, E., Breese, J., Heckerman, D., Hovel, D., Rommelse, D., 1998. The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users. *Fourteenth Conference on Uncertainty in Artificial Intelligence,* 256-265. Morgan Kaufmann.
6. Howard, R. & Matheson, J. Influence diagrams. 1981 In Howard, R. and Matheson, J. editors, *Readings on the Principles and Applications of Decision Analysis*, volume II, pages 721-762. SDG, Menlo Park, CA.
7. Kanazawa, K., & Dean, T. 1989. A model for projection and action. In *Proceedings of the Eleventh IJCAI*. AAAI/IJCAI.
8. Levin, E., Pieraccini, R., & Eckert, W. 2000 A Stochastic Model of Human-Machine Interaction for Learning Dialogue Strategies. *IEEE Transactions on Speech and Audio Processing*, Volume 8, No. 1, 11-23.
9. Paek, T. & Horvitz, E. 1999. Uncertainty, utility, and misunderstanding. *AAAI Fall Symposium on Psychological Models of Communication*, North Falmouth, MA, November 5-7, 85-92.
10. Paek, T. & Horvitz, E. 2000. Conversation as Decision Making Under Uncertainty. *Sixteenth Conference on Uncertainty in Artificial Intelligence*. 455-464. Morgan Kaufmann Publishers: San Francisco.
11. Paek, T., Horvitz, E., & Rudder, E. 2000. Continuous Listening for Unconstrainted Spoken Dialog. ICSLP 2000, Beijing.