Understanding User State and Preferences for Robust Spoken Dialog Systems and Location-Aware Assistive Technology

by

William Li

B.A.Sc. Engineering Science, University of Toronto (2009)

Submitted to the Engineering Systems Division
and the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of
Master of Science in Technology and Policy
and
Master of Science in Electrical Engineering and Computer Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

Author ....
Engineering Systems Division
Department of Electrical Engineering and Computer Science
May 18, 2012

Certified by ....
Nicholas Roy
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Certified by ....
Seth Teller
Professor of Computer Science and Engineering
Thesis Supervisor

Certified by ....
James R. Glass
Senior Research Scientist
Thesis Supervisor

Certified by ....
Joseph F. Coughlin
Director, MIT AgeLab
Thesis Reader

Accepted by ....
Joel P. Clark
Professor of Materials Systems and Engineering Systems
Acting Director, Technology & Policy Program

Accepted by ....
Leslie A. Kolodziejski
Professor of Electrical Engineering
Chair, Department Committee on Graduate Students

Understanding User State and Preferences for Robust Spoken Dialog Systems and Location-Aware Assistive Technology

by

William Li

Submitted to the Engineering Systems Division
and the
Department of Electrical Engineering and Computer Science
on May 18, 2012, in partial fulfillment of the
requirements for the degrees of
Master of Science in Technology and Policy
and
Master of Science in Electrical Engineering and Computer Science

Abstract

This research focuses on improving the performance of spoken dialog systems (SDS) in the domain of assistive technology for people with disabilities. Automatic speech recognition (ASR) has compelling potential applications as a means of enabling people with physical disabilities to enjoy greater levels of independence and participation. This thesis describes the development and evaluation of a spoken dialog system modeled as a partially observable Markov decision process (SDS-POMDP). The SDS-POMDP can understand commands related to making phone calls and providing information about weather, activities, and menus in a specialized-care residence setting. Labeled utterance data was used to train observation and utterance confidence models. With a user simulator, the SDS-POMDP reward function parameters were optimized, and the SDS-POMDP is shown to out-perform simpler threshold-based dialog strategies. These simulations were validated in experiments with human participants, with the SDS-POMDP resulting in more successful dialogs and faster dialog completion times, particularly for speakers with high word-error rates.

This thesis also explores the social and ethical implications of deploying location-based assistive technology in specialized-care settings. These technologies could have substantial potential benefit to residents and caregivers in such environments, but they may also raise issues related to user safety, independence, autonomy, or privacy. As one example, location-aware mobile devices are potentially useful to increase the safety of individuals in a specialized-care setting who may be at risk of unknowingly wandering, but they raise important questions about privacy and informed consent. This thesis provides a survey of U.S. legislation related to the participation of individuals who have questionable capacity to provide informed consent in research studies.

Overall, it seeks to precisely describe and define the key issues that are arise as a result of new, unforeseen technologies that may have both benefits and costs to the elderly and people with disabilities.

Thesis Supervisor: Nicholas Roy
Title: Associate Professor of Aeronautics and Astronautics

Thesis Supervisor: Seth Teller
Title: Professor of Computer Science and Engineering

Thesis Supervisor: James R. Glass
Title: Senior Research Scientist

Acknowledgments

I am very grateful for the assistance, guidance, mentorship, and participation of many people in the formation, writing, and review of this thesis; it truly was not a solo effort. First and foremost, I thank my advisors, Nicholas Roy, Seth Teller, and Jim Glass. Nick's patient and rigorous guidance, Seth's creativity and support, and Jim's thoughtful insights contributed immensely to the work here and in my growth as a person and researcher. A single paragraph, or even many pages of acknowledgments, would not be able to do justice to the learning and research environment that they have cultivated.

I also thank Joseph Coughlin for reading this thesis and providing guidance on the technology policy component. His involvement has helped me to consolidate my learning and coursework as a student in the Technology and Policy Program (TPP).

Other students and staff in CSAIL, especially in robotics and speech, have also been very helpful in making this thesis a reality. In particular, I thank Sachithra Hemachandra, Ross Finman, David Hayden, Sudeep Pillai, Jun-geun Park, Josh Joseph, Finale Doshi-Velez, and Javier Velez, along with many others in 32-33x, for many conversations and discussions that helped shape the ideas and results in this thesis. At numerous stages, Dorothy Curtis and Bryan Reimer were a source of answers and assistance. Meanwhile, in SLS, Hung-An Chang, Jacqueline Lee, Stephen Shum, Ekapol Chuangsuwanwich, Ian McGraw, and Scott Cyphers provided critical explanations and support that directly contributed to the development of the spoken dialog system. Finally, I have had the chance to work with several bright research students who made useful contributions to this research, including Minh Phan, Sebastian Nowak, Erika Lee, and Alex Sekula.

Navigating TPP, EECS, and dual-degree process would not have been possible without Krista Featherstone and Janet Fischer, along with Sydney Miller and Ed Ballo. I also thank Sophia Hasenfus and Bryt Bradley for their support.

Much of the work of this thesis was done at The Boston Home, where I have had the privilege to spend considerable time during my Master's studies. To me, TBH has been like a second campus; I have learned more there than I could have imagined. In particular, Don Fredette provided his help, technical advice, and assistance in every imaginable way to help make this thesis a reality. As well, I thank Marva Serotkin, Cindy Walsh, Marva Serotkin, and Alex Burnham for their openness, flexibility, and assistance at many stages of this thesis. Finally, I am truly grateful for the guidance and participation of many TBH residents who motivated and contributed to this research.

Last but not least, I thank my friends in TPP, Sidney Pacific, EECS and CSAIL, Toronto and elsewhere, and my parents and sister for their love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

People with physical disabilities may experience difficulties accessing and controlling their surroundings. Affordances [28] such as doorknobs, light switches, and remote controls, electronics ranging from computers to smartphones, and objects like writing instruments or eating utensils can be difficult or impossible to use. These limitations can adversely affect their quality of life.

Human caregivers and personal support workers can assist with these challenges. However, human assistance, along with not always being available, presents its own issues. First, having a dedicated human caregiver is expensive; nurses in specialized-care settings or home-care workers must divide their time among many residents or clients. Second, from the perspective of the person with the disability, receiving help does not address the issue of lost independence. For these reasons **assistive technology** that maximizes the independence of people with disabilities is attractive; among other benefits, appropriate assistive technology has been shown to reduce expenditures for home care costs and institutional care for the frail elderly [50]. Beyond existing devices, substantial interest exists in personal robotics as assistive devices; early ethnographic studies have suggested that assistive robots can play a role in ensuring high-quality affordable care [25].

One basic observation of human-to-human interaction in healthcare settings is that the interaction is largely speech-based. Although severe speech dysarthria or loss of speech may co-occur with physical disabilities or aging, a caregiver who adapts to the person's speaking style, asks clarification or confirmation questions, or simply is patient can mitigate many of these issues. Therefore, the focus of this thesis is enabling automatic speech recognition and spoken dialog systems as a means of controlling assistive devices. People with physical disabilities may be unable to use traditional physical input devices such as keyboards, pointing devices, or touchscreens; for these individuals, speech could be an effective human-computer interaction modality. Robust spoken dialog systems could have a wide range of assistive applications, including enabling devices that that provide a person mobility, information, communication, or environmental control.

This thesis largely focuses on the problem of spoken dialog management; however, more generally, the goal of this work is to infer a person's desires or needs (collectively, his or her "state") robustly. Most of this thesis examines inferring the user's state through *speech*, but some sections, particularly the discussion of technology policy issues and areas of further work, take a broader view of the user's state.

1.1 Application Domain: The Boston Home

The motivation for assistive spoken dialog systems stems from ongoing work at The Boston Home (TBH), a 96-bed specialized-care residence for adults with multiple sclerosis and other neurological conditions [12]. As discussed later in Section 3.1, a range of speech-enabled information, communication, and environmental-control functions could benefit residents at TBH and enhance their quality of life. The development of robust speech interfaces is a component of ongoing work to develop a voice-commandable autonomous wheelchair that can assist wheelchair users with their mobility and other needs [34]. Other systems that could potentially be placed under voice control include environmental control systems, robotic arms, or full-fledged personal robots.

A central challenge of enabling such assistive technologies, however, is that many residents exhibit high speech-recognition word-error rates when interacting with ASR systems. This may be due to fatigue, voice strain, or other pathologies that cause their

Table 1.1: Concept error rate for MIT speakers on 30 utterances

Speaker	Concept Error Rate
lab01	3.3%
lab02	9.9%
lab03	6.6%
lab04	13.3%
lab05	3.3%
lab06	3.3%
lab07	0.0%
mean	7.5%
std. dev.	4.3%

Table 1.2: Concept error rate for TBH speakers on 30 utterances

Speaker	Concept Error Rate
tbh01	13.3%
tbh02	3.3%
tbh03	33.3%
tbh04	56.7%
tbh05	26.7%
tbh06	9.4%
tbh07	6.6%
mean	21.4%
std. dev.	18.9%

speech patterns to differ from acoustic models trained on speech from able-bodied individuals. In general, these speech difficulties may cause high concept error rates, where the top hypothesis from the speech recognizer does not map to the user's actual intent (a more precise definition of a "concept" and the "concept error rate" can be found in Chapter 3. Tables 6.2 and 1.2 illustrate the differences in concept error rates between TBH residents who participated in this research and able-bodied individuals in the lab. Most notably, as Table 1.2 shows, three of seven residents at The Boston Home have very high concept error rates above 25%. As well, greater variability exists among TBH speakers than among our control group of MIT laboratory participants.

These facts suggest a need for the development of a spoken dialog system that robustly handles uncertain speech inputs. **This research hypothesizes that dialog strategies that consider the uncertainty associated with user utterances can enable higher task completion rates, particularly for speakers with high speech recognition error rates.**

1.2 Dialog Management Strategies

For speech-based assistive technologies to become widely available, many open problems in speech recognition and dialog systems remain. These challenges include developing noise-robust ASR, handling fatigued, strained, or dysarthric speech, understanding free-form, natural-language commands, and creating robust and scalable strategies for successful dialogs between humans and machines. This thesis focuses on the last of these challenges: developing a probabilistic dialog-management strategy so that assistive spoken dialog systems can more effectively serve the needs of their users.

A few intuitions about human-to-human spoken dialog guide this research. First, when a person's intent in a conversation is unclear (due to poor speech recognition or semantic ambiguity), his or her dialog partner may ask the speaker to repeat, clarify, or confirm. Critically, a sensible dialog partner chooses one of these actions only in the face of uncertainty about the intent; for example, asking questions would be unnecessary and even irritating if the intent is clear. More abstractly, dialog management can be modeled as an inference problem, in which hidden state (the user's intent) may be inferred from observations (the output of the speech recognizer), and the task of the dialog manager is to select appropriate actions (asking the user to repeat, confirming a particular intent, or fulfilling the inferred intent) in an optimal manner. These insights are formalized in work described in Section 2.5; to summarize, dialog management is modeled as a partially observable Markov decision process (POMDP) [62, 59, 70]. The focus of the technical part of this thesis is on the development of a spoken dialog system POMDP, referred to as a SDS-POMDP throughout this thesis.

1.3 Ethical Implications of Intelligent Assistive Technologies

As a dual Master's thesis with the Technology and Policy Program (TPP), this thesis also discusses the ethical implications of deploying intelligent assistive technologies in a specialized-care setting. Such technologies could employ information about the user's state, including health, location, desired activities, or other behaviors, to augment the activities of caregivers or enhance the effectiveness of other assistive devices. However, beyond the functions and capabilities of new technologies for aging, concerns related to trust, privacy, and stigma must be carefully considered for these innovations to be actually adopted and successful [19]. For example, the collection, use, and application of location data could allow staff at a skilled nursing facility to to remain aware of the whereabouts of residents, particularly those who may be at risk of wandering off the grounds of TBH due to cognitive challenges, could help improve resident safety. Location monitoring, however, raises important questions about whether these safety benefits come at the expense of the resident's independence, privacy, or dignity. How can these values be effectively balanced in a residential-care setting? What stakeholders and viewpoints need to be considered in deciding whether such technologies should be deployed? These questions are posed and discussed in Chapter 8.

1.4 Contributions

This thesis is driven by the problem of developing a robust spoken dialog system with applications to assistive technology. The main contributions are in the development and evaluation of the end-to-end SDS-POMDP dialog system (Chapters 3 and 7), concept-level utterance confidence scoring (Chapter 4), simulation and reward function optimization (Chapters 5 and 6), and ethics and public policy related to assistive technologies for the elderly and people with disabilities (Chapter 8). These contributions are outlined below.

1.4.1 SDS-POMDP Development and Evaluation

The full SDS-POMDP model is learned from a novel corpus collected from laboratory and TBH speakers. In particular, the high concept error rates of TBH speakers present difficulties to simple rule-based dialog managers. As discussed in detail in Chapter 3, the SDS-POMDP takes observations in the form of hypothesized speech recognition concepts, which is discrete, and an associated confidence score, which is continuous. How these parts of the observation are handled by the SDS-POMDP is described in this thesis. The SDS-POMDP is found in Chapter 7 to outperform rule-based dialog managers, both in terms of dialog completion rate (successful versus unsuccessful dialogs) and dialog completion time, with TBH users in a controlled experiment.

1.4.2 Confidence Scoring of Utterances

As described above, the utterance observations that the SDS-POMDP handles in order to manage the dialog consist of both a hypothesized concept and its probability of correctness. To compute this probability, a custom confidence-scoring module is trained from data in this thesis and found to exhibit strong performance on a held-out test set. Specifically, the confidence model was constructed in a supervised manner using concept-annotated utterances, Then, adaptive boosting (AdaBoost) [26] was used to train a classifier for correctly and incorrectly recognized utterances. This approach was found to be effective as a method of selecting from a large number of diverse features from the ASR output, including acoustic, language-model, n-best, syntactic, semantic, and corpus-level features; overall, the system achieved a minimum error rate of 6.9% on a held-out test set. A logistic curve was fit to the AdaBoost scores in order to convert the classifier into a confidence score between 0 and 1.

1.4.3 User Simulation and Reward Function Optimization

As discussed in detail in section 2.5, the observation function of the SDS-POMDP is used as a generative model of the user; for any given state-action pair (S, A), we have a conditional probability distribution, $P(Z|S, A)$, where Z is the set of observations. This model is used to compare the performance of the SDS-POMDP in simulation; we find that the SDS-POMDP with continuous confidence score observations outperforms both simple threshold-based methods and a SDS-POMDP without confidence scores. The user simulator is also used as a way to compute the average dialog cost to find reward function parameters that lead to a POMDP policy that performs better in the real world. Such an approach is useful when, as in this case, it is difficult to compute the optimal reward function existing methods.

1.4.4 Ethical Challenges in Assistive Technology

This section discusses the real-world ethical dilemmas that arise in deploying technologies for a vulnerable population. It describes how new technologies, such as a location-awareness system that allows staff in a specialized-care residence to know the whereabouts of residents who may be at risk of wandering, offer benefits but also raise other issues. The interests and roles of different stakeholders are analyzed in the context of a location awareness system deployed and evaluated at The Boston Home. This chapter discusses the current mechanisms that exist for individuals with questionable consent capacity to participate in research. In particular, existing guidelines for surrogates, such as a resident's family members or activated health care proxy, are not explicitly defined for decision-making on non-medical research and new technologies, but existing legislation and practices by MIT and other institutions provide some guidance on how to proceed.

1.5 Roadmap

This remainder of this thesis is structured as follows:

- Chapter 2 (Background and Related Work): This chapter examines the current state of research on spoken dialog systems, particularly POMDP-based dialog managers, along with existing work on confidence scoring of speech recognition hypotheses.

- Chapter 3 (POMDP-based Dialog Management): In this chapter, we describe how the dialog manager for this research was constructed. The key design considerations, methods of collecting data, and how the models were trained are discussed in detail.

- Chapter 4 (Utterance Confidence Scoring): This chapter focuses on how the confidence model for utterances was trained, along withhow the confidence scores were incorporated into the SDS-POMDP model.

- Chapter 5 (User Simulation Results): In order to evaluate the SDS-POMDP efficiently, a user simulator that probabilistically describes the user's behavior given certain goals and actions was developed. This chapter describes this simulator and how the SDS-POMDP compared to baseline threshold-based dialog strategies.

- Chapter 6 (Searching over Reward Function Parameters): This chapter describes how the user simulator was used to optimize the performance of the SDS-POMDP, as measured by the total time to completion.

- Chapter 7 (Human Experiments): To validate the end-to-end system, the SDS-POMDP is compared to a threshold-based dialog manager with research participants at The Boston Home. This chapter details the design and results of these experiments.

- Chapter 8 (Deploying Assistive Technologies: Ethical and Policy Considerations): This chapter describes the potential ethical dilemmas that exist with deploying intelligent assistive technologies in a specialized-care setting, particularly the potential conflict between safety and personhood that may arise with location awareness systems. A survey of existing legislation and policies on involving individuals with a questionable capacity to consent is also conducted.

- Chapter 9 (Conclusion): In this final chapter, the main contributions of this research, its limitations, and oppotunities for further work are summarized.

Chapter 2

Related Work and Technical Background

This chapter focuses on related work in speech-based assistive technology, confidence scoring in speech recognition, and spoken dialog systems. It also provides some background in partially observable Markov decision processes (POMDPs) and their application to spoken dialog management

2.1 Speech-Based Assistive Technology

Speech-enabled assistive technologies are a compelling application of automatic speech recognition; people who cannot use other kinds of input methods into computers or other devices may be able to control them using their voice. By way of example, many individuals with mobility impairments, including some residents at The Boston Home, use speech recognition software for a range of purposes. Some isolated-word recognizers have been integrated into systems that control switch-based devices, including nurse call systems and adjustable beds [10], while other residents use continuous-word speech recognition software, such as Nuance's Dragon NaturallySpeaking or Windows Speech Recognition, to access computers [7].

The success of commercial off-the-shelf speech recognition software varies among people with physical disabilities. At TBH, some residents use such software extensively, others report occasional or frequent difficulties depending on the speech task, quality of their speech, level of fatigue, or other issues, and still others have been completely unsuccessful in their attempts to use automatic speech recognition. While successful assistive technology adoption involves many considerations [42], improved ASR could help lead to wider adoption of speech-based assistive technologies. Currently available commercial systems often use train the speech recognizer through an initial training session or online speaker adaptation in order to reduce word-error rates; in the case of continuous speech recognizers, the software might show a list of hypotheses when appropriate. Notably, though, there appears to be little work in this domain involving dialog management techniques to improve system performance.

Meanwhile, speech has been proposed as an interaction modality on a number of assistive robots, including robotic wheelchairs [15, 22, 34] and humanoid assistants [52, 56]. These robotic platforms are designed to assist with mobility, personal care, information, or emotional needs. Along with being a natural way of interacting with assistive robots, speech interaction may be be particularly useful for individuals with fine-motor skill impairments that may prevent them from using existing typical input devices.

2.2 Spoken Dialog Systems

A spoken dialog system (SDS) is an agent that interacts with the end-user who uses speech. In its most general form, it includes the following components:

1. Automatic speech recognition (ASR): The user's input into an SDS is spoken language, meaning that a speech recognizer is needed to convert human speech utterances into a machine-readable form. The output from a speech recognizer could include strings of words and statistics about how those strings match the speech recognizer's models of speech.

2. Dialog management: The dialog manager receives the output from the ASR and makes *decisions* about what actions to take. In principle, the dialog manager

could vary greatly in terms of complexity; it could be range from a few rules and consider only the most recent ASR output to more sophisticated models that, for example, account for uncertainty in the ASR output and consider the history of the dialog. This thesis focuses on modeling dialog processes as a partially observable Markov decision process (POMDP), which we define in detail below. Thus, we name our spoken dialog system the SDS-POMDP.

3. Output interface: The SDS must respond to the user or send messages to other components of the system, based on the decision made by the dialog manager. This might take on the form of a graphical display or speech synthesis using text-to-speech (TTS) technologies. Some spoken dialog systems are designed to interact with other systems; for example, a robot's SDS might send messages to the robot's actuators to effect a certain action. Although a rudimentary graphical and TTS output interface was developed for the prototype system, these components of an end-to-end SDS are not the focus of this thesis.

2.2.1 Automatic Speech Recognition

Although the focus of this research is mostly on the POMDP dialog manager, understanding the nature of ASR outputs is important for at least two reasons: First, although a POMDP, in an abstract sense, can handle any kind of observation about its environment, the type of these observations have a substantial impact on how the POMDP is actually constructed. Second, it is helpful to understand how ASR works in order to grasp the terminology used in this thesis to describe certain components of the SDS-POMDP, particularly its observation function.

Briefly, modern ASR systems, including the MIT SUMMIT system used in this thesis, take input speech utterances (encoded digitally as audio files) and use probabilistic models of acoustics and language to infer strings of words [29]. ASR can be summarized by the following equation:

$$P(z|a) \propto P(a|z)P(z) \tag{2.1}$$

where z is the hypothesized string of words and a is the acoustic evidence. $P(a|z)$ corresponds to the acoustic model that maps acoustic inputs to different speech sounds, while $P(z)$ refers to the language model that encodes the prior probability of the string of words. Inferring the correct hypothesis is simply an $\arg\max_z$ operation over all possible strings of words. In its most basic form, therefore, an ASR system generates a set of hypothesized string of words with acoustic and language model probability scores. Some ASR configurations, including the one that was used in this research, return an n-best list of hypotheses (in our case, $n = 10$). As described in Chapters 3 and 4, information from the n-best list may also be useful for the dialog manager.

As shown in Tables 6.2 and 1.2, ASR is not perfectly accurate, particularly for users with speech challenges associated with their disabilities. Consequently, an effective dialog manager must be able to *uncertainty* in speech recognition hypotheses while making good decisions. Designing a dialog manager with this property is the focus of this thesis.

2.2.2 Rule-based Dialog Managers

In an SDS, the dialog manager is the subject of significant design and engineering effort. To illustrate, Paek and Pieraccini [53] provide an in-depth discussion of existing production, large-scale dialog management systems, such as those used in automated call centers. The construction of dialog managers is a major effort, with many person-hours devoted to developing specifications for the spoken dialog system's behavior, developing the system, testing extensively, and monitoring usage to make improvements. Most existing dialog systems often use rule-based approaches for handling speech recognition inputs and determining system actions. They may have large decision trees that map speech recognition hypotheses to system actions and intricately crafted error-handling techniques. Some dialog systems may simply use the top hypothesized ASR result, processing it as if it were completely confident; others might use confidence measures and set certain thresholds for confirmatory system actions. Traditionally, these thresholds might be set by hand or learned from data.

2.3 Models for Planning Under Uncertainty

This thesis models dialog management as a partially observable Markov decision process (POMDP) and discusses (in theory, simulation, and experimental work) how a spoken dialog system modeled as a POMDP (SDS-POMDP) produces robust dialog performance. An introduction to Markov decision processes (MDPs) and partially observable MDPs (POMDPs) is presented here as technical background.

MDPs and POMDPs are formulations for planning under uncertainty. In both cases, "uncertainty" means that there is stochasticity in the domain that the agent is considering. In an MDP, the agent knows what state it is in, but there is stochasticity in actions; that is, the effects of system actions are not deterministic. A POMDP is a generalization of a MDP: Along with stochasticity in system actions, the state is not known, and the system can only make observations that may be informative about the underlying state.

2.3.1 Markov Decision Processes (MDPs)

MDPs consist of the tuple $\{S,A,T,R,\gamma\}$ [47]:

- S: The set of states in the environment in which the agent may find itself.
- A: The set of actions that the agent may take.
- T: The transition function, $T(S', S, A) = P(S'|S, A)$, which models, for each state in S, the effect of each action in A in terms of transitioning to state S.
- R: The reward function, $R(S, A)$, which describes the expected immediate reward gained by the agent for taking an action in A while in state S.
- γ: The discount factor that expresses the relative value of future rewards as a function of time; in some sense, γ expresses the "time value" of the reward. Typically, $0 \leq \gamma \leq 1$; low values of γ will bias the agent away from action sequences that may have a payoff in the future. A discount factor of less than

one, with bounded rewards, will also ensure that the expected discounted reward is finite [36].

A canonical example of the Markov decision process is mobile robot path planning, with states corresponding to locations on a map, actions that may occur with some error (such as overshooting or undershooting a desired translation), and rewards (which may be positive or negative) for reaching the goal location, not reaching the goal location, or colliding into objects. A policy, $\Pi : S \to A$, defines what action the MDP should take in any given state; one way of defining the optimal policy, Π^*, is the mapping of states to actions that maximizes the expected discounted reward [36]:

$$R = [\sum_{t=1}^{\infty} \gamma^t r_t] \tag{2.2}$$

2.3.2 Partially Observable Markov Decision Processes (POMDPs)

POMDPs generalize MDPs, adding the notion of observations, Z, and an observation function, Ω [16, 62]:

- Z: The set of observations that the agent may make about the environment.
- Ω: The observation function, $\Omega(Z, S, A) = P(Z|S, A)$, which models, for each state-action pair, the probability of each observation in Z.

Extending the example above, a robot may not be able to determine exactly where it is in the environment due to imperfect sensors, meaning that its true state is not known. Like the fully observable MDP case, rewards are defined for every state-action pair. However, because the agent does not know which state it is in, it must make decisions based on its *belief* distribution over states, $b(s)$. After each action and observation, the posterior belief can be calculated from the prior belief, the transition function, and the observation function, as shown in equation 2.3. Therefore, we need to compute a policy that maps all possible beliefs to actions, $\Pi(b)$. The extension of MDPs to POMDPs adds a number of other challenges, including substantial increases in computational complexity required to compute an optimal policy [41]. In addition, most agents in POMDPs now have additional information-gathering actions to consider — actions that decrease the entropy of the belief may help the agent increase its expected discounted reward. In the robot example, a "sensing" action may incur a small cost in terms of time, but it could help the robot localize itself and perform better subsequent actions.

2.4 MDP and POMDP Dialog Management

In contrast to the rule-based, hand-crafted nature of many existing dialog management systems, researchers have focused on formulating dialog management as a model-based reinforcement-learning problem. By specifying a dialog manager as either an MDP or POMDP, standard automatic methods of computing optimal policies can be used; in other words, according to the specified reward function, the actions that the dialog manager takes should be optimal. Taking this approach, it is envisioned, could automate the construction of dialog managers, lead to more optimal dialog behaviors, and ensure system robustness even as ASR performance degrades. In particular, this last benefit is especially of interest due to the high ASR error rates of TBH residents and other speakers with disabilities.

Although robot navigation initially seems quite different from spoken dialog management, they share certain features that make it possible to cast dialog management as an MDP or POMDP. Specifically, dialog management is a stochastic domain because automatic speech recognizers may be imperfect sensors of the user's true goals in interacting with the dialog system. Given these noisy ASR results, the challenge of dialog management is to find the best system action to take, whether it be greeting the user, asking a clarification question, or assuming that the ASR result was correct and responding to the user; in other words, an optimal policy is equivalent to an effective dialog manager.

2.4.1 MDP-based Spoken Dialog Systems

Early work focused on dialog management as an MDP [45, 46, 61]. In these dialog managers, the MDP state generally reflects the state of the dialog. For example, in a dialog system in which the user is specifying two entities, such as the origin and destination of a flight, the states include all valid combinations of origins and destinations, including whether the origin and destination "slots" have been filled in the dialog. The probabilistic nature of speech recognition hypotheses is encoded in the transition function, T [45]. For example, given a particular initial state s and an system action, a, T encodes $p(s'|s, a)$ for every possible new state s', based on the performance of the speech recognizer.

2.5 POMDP-based Spoken Dialog Systems

One problem with the above formulation is that it requires the system to track the state of the dialog accurately. More recently, increasing attention has focused on modeling dialogs as partially observable Markov decision processes (POMDPs). This model, first proposed by Roy et al. [59], treats the user's intent in the dialog as the hidden state and speech recognition hypotheses as noisy observations of that intent. As it is the focus of this thesis, we work through a detailed example of how a dialog manager is encoded as a POMDP in this chapter.

The elements of the spoken dialog system POMDP (SDS-POMDP) are outlined in Table 2.1. To help illustrate these elements, consider a simple, toy dialog system that serves two functions: it can either make a phone call or report the weather forecast. The speech recognition hypotheses in this example are mapped to specific concepts with confidence scores.

As shown in Table 2.1, in an SDS-POMDP, the user is assumed to be in a state, S, which is hidden from the system. Because the state is a hidden variable, the system maintains a belief, b, which is a probability distribution that the system maintains over states (user intents). The system makes observations (results from the automatic speech recognizer) about the user's state, updates its belief about the hidden state,

Table 2.1: Random variables in a SDS-POMDP

Entity	Symbol	Description	Toy Example
States	S	The possible user intents. The state is a hidden variable; the system seeks to infer the user's intent by making observations (the user's speech utterances) and taking actions.	`get_weather_forecast`, `make_phone_call`
Actions	A	The actions that the system can take. In a dialog manager, this includes opening `GREET` actions, terminal `SUBMIT` actions, which it takes once it is sufficiently confident about the user's intent, or intermediate actions that might be used to acquire more information, such as asking the user to repeat a request, or confirmatory questions about certain intents.	`GREET`, `CONFIRM get_weather_forecast`, `CONFIRM make_phone_call`, `SUBMIT get_weather_forecast`, `SUBMIT make_phone_call`

Observations	O	The speech recognition hypotheses that are observed by the dialog manager from the speech recognizer. In the case of POMDP dialog managers, observations consist of speech recognition hypotheses encoded as concepts with associated confidence scores. Details about how strings of words are converted to concepts can be found in Chapter 3.	`( get_weather_forecast, confidence=0.98 )`,
			
			`( make_phone_call, confidence=0.41 )`
			
			`( null (did not parse) )`

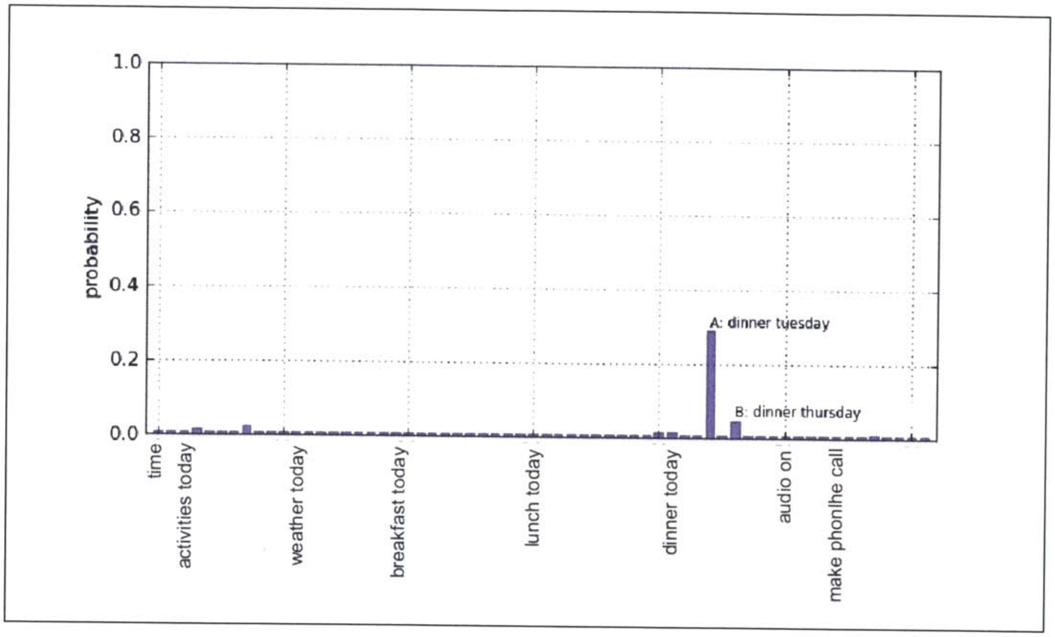

Figure 2-1: Plot of Sample SDS-POMDP belief. The horizontal axis corresponds to each POMDP state, s, while the height of each bar is the belief probability in that state. In this particular case, "dinner tuesday" (A) and "dinner thursday" (B) have non-negligible probability.

and takes an optimal action depending on its belief. Three key functions encode these relationships, as described in Table 2.2.

An example belief distribution is shown in Figure 2-1. This distribution is the posterior that is generated with a uniform prior over states and a speech recognition observation in which top hypothesis was '"dinner tuesday" and the confidence score was 0.5. As expected, the state with the highest posterior probability is "dinner tuesday"; however, this probability is not very high (less than 0.4) and the state with the second-highest probability is "dinner thursday." The intuitive explanation for these two facts is that the SDS-POMDP is not especially confident in the "dinner tuesday" hypothesis, and that it is taking into account the tendency of "dinner thursday" to be confused with "dinner tuesday." The details of the SDS-POMDP that enable it to encode these insights algorithmically are discussed in Chapter 3.

Table 2.2: SDS-POMDP probability functions

Entity	Symbol	Description	Toy Example
Transition function	$T(s', s, a) = P(s' \| s, a)$	T maps the current state from the previous state s and the machine's action a; it describes how the user's intent s' changes as a function of the user's previous intent, s and the machine's response a. Practically, when a dialog terminates (the system takes a terminal action), the dialog resets to a new initial belief.	$P($`make_phone_call`$\|$ `make_phone_call`, `GREET` $) = 1$ $P($`get_weather_forecast`$\|$ `make_phone_call`, `GREET` $) = 0$
Observation function	$\Omega(z, s, a) = P(z \| s, a)$	Ω is a model for how observations are generated from different states and actions. Corresponding observation-state pairs might have high probability. The observation model is exactly the user model – it encodes the probability distribution over possible observations for a given user intent s and machine action a.	$P($`make_phone_call`$\|$ `make_phone_call`, `GREET` $) = 0.8$ $P($`get_weather_forecast`$\|$ `make_phone_call`, `GREET` $) = 0.2$
Reward function	$R(s, a)$	R encodes the desired system behavior. It describes the rewards or costs (negative rewards) that the system incurs for actions while the user is in the given state. "Correct" terminal actions might have positive rewards, while "incorrect" terminal actions should have negative rewards. Intermediate actions, meanwhile, such as asking the user to repeat or requesting confirmation, might be associated with a small negative reward to promote shorter dialogs.	$R($ `make_phone_call`, `make_phone_call` $) = +10$ $R($`make_phone_call`, `get_weather_forecast` $) = -100$

2.5.1 Computing a Policy

Given these random variables and probability distribution functions, it becomes possible to create a POMDP dialog manager. First, the model needs to be specified, with all possible user intents mapped to states, all system responses to actions, all speech recognition results to observations, and the transition, observation, and reward functions defined. Then, the POMDP must be *solved*; that is, for every point in the belief space, a mapping from beliefs to actions, called a *policy*, is determined. The policy is denoted as $\Pi(b)$; it is a function that maps beliefs to actions. Various solution techniques for POMDPs, methods such as Point-Based Value Iteration (PBVI) that admit continuous observations (for confidence scores) have been used for POMDP dialog managers [68]; approximation techniques, including modeling the POMDP as an augmented Markov Decision Process (MDP), have also been employed to provide reasonable policies that outperform models that do not assume observation uncertainty [59]. In this work, an even simpler approximate POMDP solution technique, QMDP, is used [48]. We attempt to characterize the performance of the dialog system with the QMDP-derived solution.

In interacting with the SDS-POMDP, the user has a hidden goal state, speaks utterances (observations), and receives system responses (actions) depending on the belief state and policy of the SDS-POMDP. Updating the belief state is governed by the following equation that updates the current belief, b, to the new belief, b':

$$\begin{aligned} b'(s') &= P(s'|z,a,b) && (2.3) \\ &= \eta \cdot P(z|s,a) \sum_s P(s'|s,a) b(s) && (2.4) \end{aligned}$$

where each of the terms and variables in Equation 2.3 are as described in Tables 2.1 and 2.2 and η is a normalizing factor.

In the context of this thesis, it is worthwhile to note the potential challenges of a POMDP-based approach to dialog management. First, all of the internal probabilistic models of the POMDP—the state transition model, the observation model, and the reward function—must be learned or specified in some manner. In the absence of large amounts of training data, which may be labor-intensive to obtain, these models need to be estimated or handcrafted by the designer. Second, the proper reward function is not inherently clear for dialog POMDPs; to date, the values specified in reward functions have been arguably arbitrary (*e.g.* +10 for a correct dialog, −100 for an incorrect dialog, and −1 for a clarification question). The relative values in the reward function can have dramatic effects on the resulting policy, meaning that care must be taken in specifying them in a manner that reflects user preferences. Third, these models may not be entirely descriptive of user behavior. The observation function, for example, only models the probability distribution over the user's state and the immediate preceding system action, not the entire history of machine actions. This model limitation might affect performance if, for example, the distribution over hypotheses changes after multiple repeats of the same system action. While not all of these issues can be addressed in a single thesis, they help both to identify possible research directions and to understand the inherent limitations of the SDS-POMDP. In terms of open problems, the main thrusts of possible work might be grouped into three broad categories: extensions and refinements to the SDS-POMDP model itself, including the incorporation of continuous observations in the form of confidence scores [69], efforts to make solving SDS-POMDPs computationally tractable [70], and approaches to learn the underlying models needed for the SDS-POMDP [22].

2.6 Confidence Measures for Speech Recognition

A useful component of effective spoken dialog systems is accurate confidence scoring of speech recognition results — it would be helpful to the dialog manager had a probability of the correctness of the incoming speech recognition hypothesis [30, 31, 32, 40]. A probabilistic measure of the accuracy of the hypothesized text of a spoken utterance would make it possible for a dialog system to decide how to best respond. For instance, it might take the action assuming that the hypothesized result is correct, reject it and ask the speaker to try again, or request confirmation or clarification. A few themes of the body of research on confidence scoring are discussed in this section.

Feature Selection. One way of casting confidence scoring is to treat it as a supervised learning problem: Given a training set of speech recognition hypothesis and confidence or correct/incorrect labels, we can learn a function that predicts these scores or labels for new speech recognition hypotheses. A wide range of statistics from the speech recognizer have been shown to be useful features for confidence scoring, including signal-level features of the utterance, additional statistics from the speech recognizer's hypotheses, distributional features of the N-best lists, sentence-level characteristics, or contextual features related to the state of the dialog and the world. For example, Lemon and Konstas [44] used signal amplitude features and grammar parsability as part of similar confidence scoring work. Meanwhile, other work has used different discriminative machine-learning algorithms or have had slightly different classification tasks — San-Segundo et al. [60], for instance, used decision trees to classify utterances as being "in-domain" or "out-of-domain".

In dialog systems, dialog-level features, including the sequence of turns, have also been shown to contribute to improving confidence-scoring performance. Most notably, Lemon and Konstas [44] found that a user-system-turn n-gram ($n = 5$), which describes the probability of all possible five-turn-sub-sequences in annotated dialogs, was the feature that contributed most to information gain in their re-ranking classifier. The key to overcoming the sparsity typically associated with higher-order n-grams was the creation of large amounts of simulated data using hybrid supervised/reinforcement learning techniques [27].

Recent work has sought to incorporate language model information or other dialog context features as part of utterance confidence measures. For example, Gruenstein [30] constructed a confidence scoring module by considering additional features at the recognition and response levels of a spoken dialog system. Specifically, these statistics were augmented with two new types of features: distributional features from the n-best list, including the number of distinct possible actions and the purity of the n-best list, and response-level features that provided some indication of whether the hypothesis "made sense" in the context of the dialog; for instance, because a multimodal map interface was part of this spoken dialog system, a geographical filter feature was used to determine whether the hypothesized utterance's location was a reasonable one [31]. These additional features led to a statistically significant improvement over the classification of correct and incorrect utterances using only ASR features used in [32].

Applying Confidence Scoring to Dialog Systems. A robust confidence scoring module can be used to improve the performance of a spoken dialog system. As one example, at the aforementioned 5% false detection rate (incorrect utterances hypothesized as correct), 66% of incorrect utterances were identified using the features described in [32]. These results led to a 35% relative decrease in the concept error rate in the weather domain.

2.7 Location-Aware Assistive Technologies

Location-aware technologies have several applications to assist people with disabilities. People who are blind or have low-vision may benefit from devices that help them localize or navigate unknown environments [37, 54]; somewhat similarly, individuals with orientation difficulties may benefit from location-aware cognitive orthotics that help them with wayfinding [64]. Location information might also be used to improve ASR performance by re-weighing the language models used in the speech recognizer. These technologies may also be useful for mobile robotic devices with assistive applications, such as robotic wheelchairs.

In specialized-care settings such as The Boston Home, location-aware technologies may be able to assist with wander management — some individuals who remain mobile in power wheelchairs are at risk, due to cognitive challenges, of unknowingly wandering outside or off grounds. A system that could alert staff of a wander event could augment existing procedures. Indeed, this is not a unique problem at The Boston Home; many studies cite wandering as an increasingly important and difficult-to-manage issue for people with dementia [43]. In fact, many commercially available systems that seek to address the problem of "wander management" have become available over the last several years [20, 21].

A policy question relates to the ethics of deploying such wander-management technologies: while they may have benefits in terms of the safety of individuals, such location tagging may have an adverse effect on the dignity or personhood of the participants in the system. Over the past decade, this question has received substantial attention from ethicists focused on technology and aging, with some arguing that it is particularly complicated to strike a proper balance when it comes to technological interventions such as electronic tagging [14, 38, 57]. Any implementation of location-aware assistive technologies must be carefully considered to be accepted by residents, staff, and caregivers. There is a need for a clear policy to address these concerns.

Chapter 3

Spoken Dialog System Development

This chapter focuses on the design and development of the elements of the SDS-POMDP. While the focus of this thesis is on the data, models, and software that constitute the POMDP dialog manager, the hardware selection and interface design are also briefly discussed.

3.1 System Functions

Through conversations with residents and staff at TBH, a number of useful features were identified for a personal dialog system for residents at TBH:

- Time and date: A clock with a graphical user interface, or, for visually impaired residents, the ability for the system to speak the time and date could be a useful feature for TBH residents onboard their wheelchairs.

- Weather forecasts: Many TBH residents spend time outdoors on pleasant days. Multiple sclerosis leaves many residents susceptible to high temperatures and humidity, making weather information particularly useful.

- Activities schedules: TBH staff organize a wide range of activities, from birthday parties to art classes and religious services. The event calendar is posted in printed format throughout the building, but having access to the event schedule through spoken dialog could make it easier for residents to stay informed.

- Breakfast, lunch, and dinner menus: Many residents expressed a strong interest in knowing what's on the menu.

- Making phone calls more easily (using Skype): Using conventional telephone devices can be difficult for people with disabilities due to the small keypad buttons and the challenge of holding a handset. Some residents use adapted phones with larger buttons and alternative switches; a speech-activated telephone could eliminate the need for physical switches entirely.

- Television schedules: Most residents have televisions in their own rooms, and being able to query information about upcoming television programs could be useful.

- Environmental control: Many residents report difficulty operating light switches, nurse call buttons, or controlling devices like fans or electronics. Enabling speech-based access to these devices could improve their independence.

- Sending messages to other residents: It can be tiring or difficult for some residents to control their wheelchairs and drive to visit other residents. TBH-specific messaging software might make socializing easier.

- Personal notes and reminders: TBH residents have activities, outings, appointments, and visitors. A speech-enabled personal reminder system could help residents keep track of their own schedules.

The first four system functions (time, weather, activities, meals, and phone calls) were selected for this thesis. Table 3.1 shows the specific APIs that were used to acquire this information.

Table 3.1: Third-party application programming interfaces (APIs) for dialog system

Function	API	Notes
Time/Date	System time libraries	
Weather	Google Weather API	For Dorchester, MA (ZIP code 02124)
Activities	Google Calendar API	TBH activities were entered into an "Activities" Google Calendar
Menus	Google Calendar API	Menus were entered into a "Menus" Google calendar in a structured manner (breakfasts, lunch, and dinner at certain times on each day)
Phone Calls	Skype4Py API	Skype4Py is a Python wrapper for the Skype API, which allows the Skype interface to be called programmatically.

3.2 Data Collection

To construct the SDS-POMDP system, data was collected from a total of 20 research participants in our research lab and at The Boston Home. Participants were prompted with possible goals and asked to speak into a microphone, prefacing a natural-language question inquiring about the goal with an activation keyword like "chair" or "wheelchair". A basic power-based speech-activity detector was used to automatically segment the speech into separate utterances. An example file from the aligned dataset, with the type of annotated data, is shown in Table 3.2:

Table 3.2: Sample annotation for utterance file.

```
Filename: lab03-20101105_164.wav
Sampling Rate: 16kHz
Speaker: lab03
Date: 2010/11/05
Goal: ( lunch, Tuesday )
Orthography (ground-truth transcript): chair what's lunch for Tuesday
```

This labeled data (2039 utterances in total) was used for a number of models in the SDS-POMDP.

3.3 Automatic Speech Recognizer

The SDS-POMDP uses the MIT SUMMIT speech recognizer [29]. To train the language models of the speech recognizer, the labeled utterance was used to build a trigram language model with classes to help overcome sparsity. For example, all of the days of the week were placed in a single class called `DAY`; in training the language model, all of the days of the week were replaced by the word `DAY`, the trigram probabilities are trained with this word, and then the trigrams involving days of the week use this probability.

Overall, the recognizer uses a fairly constrained 200-word vocabulary in this recognition task. This vocabulary include tokens such as `<noise>`, `<uh>`, and `<um>`. For the duration of all experiments, the same 200-word language model was used. SUMMIT itself was run locally on the netbook or laptop computer used in each of the deployments or experiments with the dialog system, or run in batch mode on CSAIL Spoken Language System (SLS) servers.

3.3.1 Speech Recognition Outputs as Dialog System Inputs

As discussed in section 2.2.1, an ASR system like SUMMIT processes take speech utterances in the form of audio files and infer strings of words based on its acoustic and language models, along with statistics in the form of log probability scores. In addition, multiple ranked strings of words are returned in our configuration.

Our dialog system uses the notion of dialog manager observations at a "concept" level; that is, it reasons about certain concepts as opposed to reasoning directly about strings of words to make decisions. Some examples of concepts are "make phone call" and "weather today". As a result, it is necessary to map strings of words to concepts. In the current implementation, the system uses a very simple set of keyword-extraction rules:

- The top hypothesis string of words is the input into a decision tree function that chooses the concept. For example, the decision tree determines whether the word "weather" or "forecast" is in the word string. If one of these words is present, then the decision tree looks for a word like "today", "tomorrow", or "monday."

- If the top hypothesis does not map to any of the concepts in the decision tree, then the speech recognition output is assigned the "null" concept, which means that it does not map to any of the pre-defined concepts.

- If a word string maps to multiple concepts, one of the concepts is arbitrarily chosen, based on the structure of the decision tree. For example, the word string "what is the weather today tuesday" maps to "weather today" simply because

This procedure was chosen for its simplicity and ease of implementation. There are clear limitations: for example, it does not consider alternative, lower-ranked hypotheses, is brittle when there are word strings that might map to multiple concepts, and requires the user to say the necessary keywords. More sophisticated approaches could learn a probabilistic mapping of word strings to concepts from data, have methods for handling out-of-vocabulary words, or possibly consider multiple concepts. Overall, though, because residents at The Boston Home prioritized system reliability over being able to speak a large vocabulary of words, the focus was on the dialog management component of the spoken dialog system.

Meanwhile, the ASR outputs are also used to compute a confidence score, which expresses the probability of correctness of the ASR result. As discussed in chapter 4, this module is more sophisticated and uses many more features from the ASR outputs. An implicit hypothesis is that a simple keyword extractor combined with a accurate confidence score is sufficient for the dialog manager to perform effectively.

3.4 POMDP Dialog Manager

The central research and engineering effort of this thesis was the POMDP-based dialog manager. As discussed in Section 2.5, modeling a dialog manager as a POMDP is a principled, data-driven method to produce a dialog manager that exhibits desirable behavior. This section discusses how the data was used, with a special emphasis on the assumptions and approximations used to construct the different models that constitute the SDS-POMDP. By illuminating the design decisions, it becomes easier to understand the strengths and limitations of the SDS-POMDP, identify areas of further work, and explain the behavior of the system in later chapters.

3.4.1 States, S and Actions, A

The SDS-POMDP discussed in this thesis is essentially similar in structure to the toy SDS-POMDP shown in Table 2.1 but with more states. The **states** are the user goals; for example, the user may want to ask for tomorrow's weather forecast, make a phone call, or have the system go into "sleep mode". In all, there are 62 states. The actions include "submitting" and "confirming" the possible user goals, along with actions corresponding to 1) asking the user to repeat and 2) terminating the dialog. Thus, there are $2(62) + 2 = 126$ actions.

Other state spaces for SDS-POMDPs have been proposed in the literature. For example, Williams and Young [68], in a flight-booking dialog system, employs a factored state space that encodes whether various slots in the dialog have been grounded or confirmed. Indeed, many of the states share certain conceptual (*e.g.* weather for today, tomorrow, or another day of the week) or temporal (*e.g.* weather, activities, or lunch menus for Wednesday). However, this was not explored in depth in this thesis.

3.4.2 Observations, Z

The observations made by the SDS-POMDP are the audio files of the user's utterances, as processed and encoded by the speech recognizer. The form of this encoding is shown below in the form of the output from the SUMMIT XML-RPC recognizer.

Table 3.3 shows ten hypotheses from a single utterance, ordered by score. The tab-separated columns, from left to right, are the total score, the acoustic score, the language model score, the number of acoustic segments. and the number of words. The information captured by the speech recognizer includes ranked hypotheses of word strings, along with acoustic and language model scores for each hypothesis. These features are used to compute an SDS-POMDP observation in the form of a mapping to one of 65 tokens (one of the 62 concepts, "yes" or "no", or "null" if the hypothesis could not be parsed) and a confidence score.

Table 3.3: Sample ASR output for a single utterance

total log probability	acoustic model log probability	language model log probability	# phones	# words	hypothesis
-7.4845	-3.0617	-4.4227	28	5	`chair what time is it`
-12.0269	-3.8957	-8.1312	23	3	`chair what is`
-12.0955	-0.2425	-11.8530	26	3	`chair what's answer`
-12.4355	-5.0344	-7.4010	24	4	`chair what is it`
-13.0177	-1.9230	-11.0947	27	4	`chair what time is`
-13.1698	-7.3272	-5.8427	22	2	`chair voice_on`
-13.3031	-6.8496	-6.4535	25	2	`chair cancel`
-13.4944	-3.1169	-10.3776	24	4	`chair time is it`
-14.2099	-1.0144	-13.1955	26	3	`chair what answer`
-14.8912	-5.8312	-9.0599	25	3	`chair what time`

TBH residents expressed the importance of a reliable, robust system over the ability for the system to understand a wide spectrum of phrases related to the concept. For this reason, a simple keyword extractor was used to map hypotheses to concepts; for example, the hypothesis "what is the weather for tomorrow" would map to the concept `( weather, tomorrow )`. Clearly, this is a simplification of natural spoken language, but it was deemed sufficient for this application.

To compute a confidence score, the features of the n-best list are used. Intuitively, high acoustic and language model scores, grammatically correct hypotheses, or a high level of agreement among all of the hypotheses might suggest that a high confidence score should be assigned to a hypothesis; details of the confidence scoring module used in this SDS-POMDP are described in Chapter 4. In the end, the speech recognizer's outputs are encoded into a single concept with an associated confidence score. The above ASR output, for example, might be encoded as $z =$ `(concept = time_of_day, confidence = 0.56)`; features such as the acoustic score and the diversity in the n-best list might have contributed to this confidence score.

3.4.3 State Transition Function, T

The state transition function encodes changes in the underlying hidden state at each turn, given the previous state and action. More precisely, it is $P(S'|S, A)$, the conditional probability of a new state, S', given the previous state, S, and the system action, A. For the purpose of modeling dialog interactions with a POMDP, we make the assumption that, in any given dialog, the user's goal is fixed; that is, the user does not change his goal over the course of the dialog, or does not explicitly "give up" on the original intended goal.

The assumption that the user's goal is fixed in a POMDP dialog manager is a common simplification [59, 58, 68] in SDS-POMDP models. Systems with factored state spaces, as described above, use a deterministic transition function when, for example a state variable transitions from "grounded" to "confirmed" [68]; in this system, though, no such transitions can occur over the course of a single dialog.

3.4.4 Observation Function (User Model), Ω

The observation function is a probabilistic map from states and actions to observations, $P(z|s, a)$. In words, it describes, given a particular user state and the previous system action, the distribution of observations (pairs of concepts and confidence scores). The observation function is learned from data; that is, the system uses actual speech recognition outputs with known states and actions in order to learn this probabilistic function. It is, as described in Section 2.5, the user model: it encodes the speech hypotheses that are generated when the user has a particular goal and the system has just performed a certain action.

The observations from the speech recognizer consists of a discrete part (the concept) and an continuous part (the confidence score). We call these two parts z_d and z_c, respectively, and factor the observation function into two parts as per equation 3.1 using the multiplication rule of probability:

$$P(z_d, z_c|s, a) = P(z_d|s, a)P(z_c|s, a, z_d) \tag{3.1}$$

Conditioning z_d on the user goal, s. If the dialog system could perfectly reproduce the speaker's utterance 100% of the time, then the observation function would be deterministic; for example, if the words "lunch" and "today" were observed only when the user wanted to know the concept `(lunch, today)`, then $P(z|s, a)$ would be unity for that goal and zero otherwise. However, the speech recognizer is an imperfect information source. For example, some of the utterances in which the user seeks to know the concept `(lunch, today)` might be incorrectly recognized as other concepts; meanwhile, other utterances with different true goals might be recognized incorrectly as `(lunch, today)`. The labeled training data can be used to generate the probability of a particular observation given a particular state by counting the number of times that observation, z_d^*, occurs as a fraction of all observations, as shown in equation 3.2.

$$P(z_d^*|s, a) = \frac{c(z_d^*, s, a)}{\sum_{z_d} c(z_d, s, a)} \tag{3.2}$$

Conditioning on the system action, a. The observation function also depends on the preceding system action, a — the user's utterance is clearly influenced by the previous system prompt. The utterances collected from research participants implicitly assume that the previous system action was `GREET`; the data collected did not capture the user's response to confirmation questions or incorrect terminal actions. As a result, a few assumptions need to be made.

First, users may be likely to answer confirmation questions with a "yes" or "no" response. Specifically, if the correct user goal is being confirmed, there should be a high probability that the user will respond with "yes"; if an incorrect user goal is being confirmed, a high probability should be associated with a "no" response.

Importantly, "yes" and "no" observations need to be treated in the same way as other observations in equation 3.2.

From early data collection, another insight was that, when prompted with a confirmation question, the user might speak an utterance corresponding to the original goal. For example, if the user's goal is to know the weather and the system asks, "Did you want to know the lunch menu?", the user might respond, "No, what is the weather?" As a result, we assume that, when prompted with a confirmation question, the user will either speak an utterance corresponding with the desired goal or the correct (yes or no, depending on concept being confirmed) confirmation utterance. Consequently, conditional probability distribution is the normalized sum of two distributions, as defined in equation 3.2.

Lastly, when the system takes an incorrect "terminal" action (submitting an incorrect user goal), we assume that the user will provide an utterance either corresponding to "no" or the correct goal. In doing so, the SDS-POMDP should extinguish the probability mass associated with the incorrect goal. To illustrate, if the user's actual goal is `X`, a different goal is `Y`, and the preceding system action is `CONFIRM Y`, then a very low probability is assigned to $P(z = \text{NO}|\text{X, CONFIRM Y})$.

Incorporating confidence scores, z_c. In addition to being one of 65 possible discrete observations, the observation includes a continuous part, the confidence score. As expressed in equation 3.1, learning the model of confidence scores (a continuous observation space) for every state, action, and observation triple seems potentially daunting in terms of the data required; clearly, some approximations are needed.

A model similar to the one used by Williams [66] is used for this SDS-POMDP. Specifically, we use the following approximation:

$$P(z_c|s, a, z_d) = \begin{cases} P(z_c|\text{correct observation}) & \text{if } z_d \mapsto s \\ P(z_c|\text{incorrect observation}) & \text{otherwise} \end{cases} \tag{3.3}$$

Equation 3.3 encodes two different conditional probability density functions: 1) the distribution of confidence scores when utterance hypothesis is correct ($P(z_c|\text{correct observation})$), and 2) the distribution of confidence scores when there is an error ($P(z_c|\text{incorrect observation})$). These two distributions reveal that the confidence score contains important information about whether the observed concept is correct or incorrect. Figure 3-1 shows histograms of confidence scores for top hypotheses that are correct and incorrect. During the belief update step of the SDS-POMDP, we draw from the "correct observation" distribution for the state corresponding to the observation concept and from the "incorrect observation" distribution for all other states.

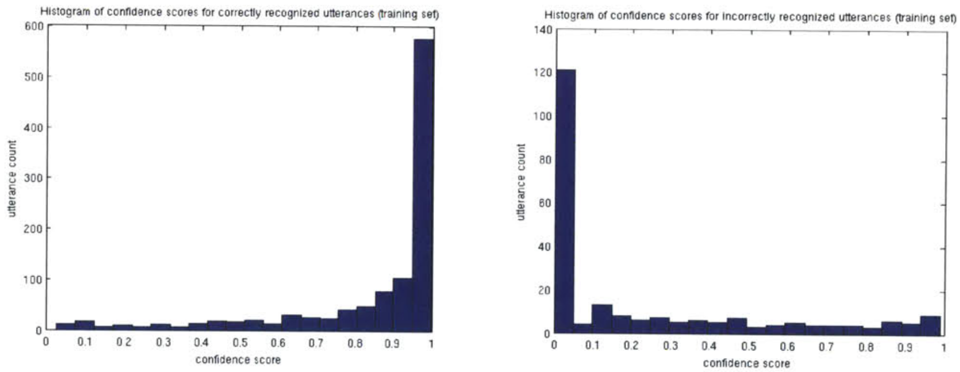

Figure 3-1: *Left:* Distribution of confidence scores for top hypotheses in training set that were correct. *Right:* Distribution of confidence scores for top hypotheses in training set that were incorrect.

This approximation assumes that the distribution of confidence scores for correct and incorrect observations are the same for every concept; however, it helps overcome data sparsity issues while still capturing the key insight that the confidence score reflects whether the utterance has been correctly or incorrectly recognized.

3.4.5 Reward Function

A challenge in developing SDS-POMDPs is determining an appropriate reward function; indeed, this is still an open challenge [67]. In general, the characteristics of a reasonable reward function this task domain are:

- The highest reward should be associated with the correct terminal action, since this is the desired behavior of the SDS-POMDP.

- Incorrect terminal actions should be assigned large negative rewards, since this is precisely the kind of behavior that the SDS-POMDP should not exhibit. The relative values of rewards for correct and incorrect terminal actions contributes substantially to how much risk the SDS-POMDP in submitting a terminal action, as Chapter 6 will show.

- A small negative reward should be assigned to asking the user to repeat his or her query. The magnitude of this reward also influences when the SDS-POMDP will submit a terminal action instead of asking the user for more information.

- Small negative rewards should also be assigned for confirmation questions. The magnitude of asking a correct confirmation question should be smaller than simply asking the user to repeat (since it may be reassuring if the SDS-POMDP asks to confirm the user's actual goal); meanwhile, the magnitude should be larger for an incorrect confirmation question.

3.4.6 Policy, Π

The policy computed by solving the SDS-POMDP seeks to maximize expected total reward. The challenge, however, is that the expected total reward may not be a perfect representation of the user's satisfaction. We use the QMDP approximation to compute a policy for the SDS-POMDP. Chapter 6, which discusses the search over reward parameters to optimize SDS-POMDP performance, provides more details.

3.5 Interface Design

Along with the speech recognizer and the POMDP dialog manager, the end-to-end spoken dialog system includes several other modules that are required for it to operate.

1. Given that the intended users of the SDS-POMDP have difficulty pushing buttons or activating switches to access electronic devices, a push-to-talk system would be undesirable. A simple GStreamer-based speech activity detector

(SAD) is used to automatically detect the start and end of speech input [51]. Briefly, the endpoint detector relies on a few parameters, including energy an threshold, minimum-duration before detecting a speech segment , and holdover time after the system drops below the energy threshold. Hand-tuning these parameters produced reasonable performance, particularly by using a close-talking microphone in the experiments.

2. The different user goals manifest themselves in different ways in the actual spoken dialog system. Instructing the system to wake up or go to sleep changes the activate state of the system; if the system is sleeping, for example, then it must be woken up before it will provide the user with other services. The interface also needs other components that translate the system actions and returned data from APIs into text outputs that are meaningful to human users that can be displayed visually or spoken by a voice synthesizer.

3. A basic graphical user interface is used for the deployed and experimental SDS-POMDP, as shown in Figure 3.5. The simple GUI shows prompts and system information in text form. The "Awake" box changes color and reads "Listening" while the user is speaking an utterance.

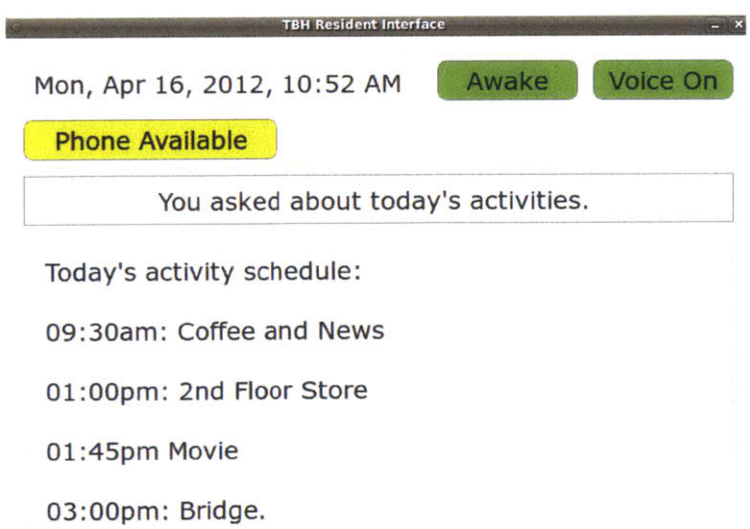

Figure 3-2: Graphical user interface (GUI) of SDS-POMDP

3.6 Hardware Selection

We use a netbook with Ubuntu 10.04 to deploy the spoken dialog system. Figure 3.6 shows one system setup at The Boston Home. In this particular case, the system is available to the user near his bed; he can use the system while in bed or by driving up to it in his wheelchair.

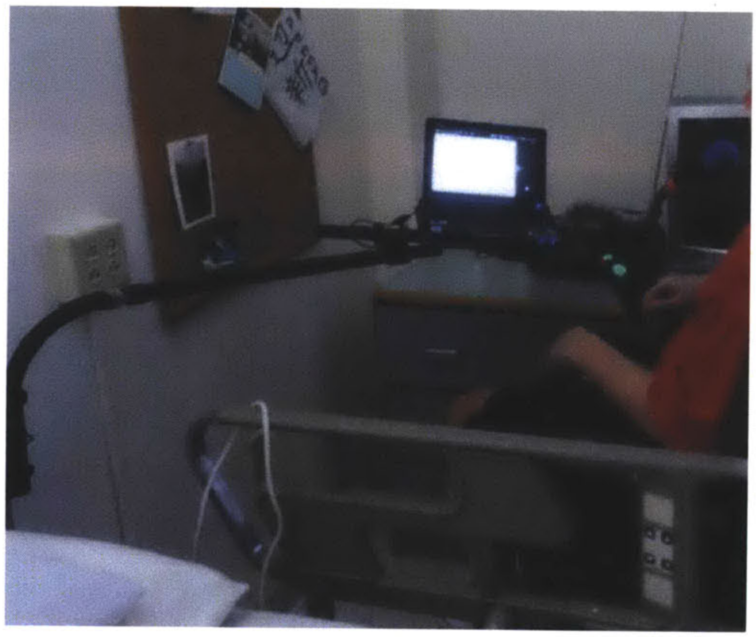

Figure 3-3: Example setup of speech interface in resident room at The Boston Home.

Chapter 4

Utterance Confidence Modeling

As discussed in Chapter 3, observations in the SDS-POMDP have a discrete part (the concept to which the top ASR hypothesis maps) and a continuous part (the confidence score). The confidence score encodes the certainty with which the speech recognizer is correct; consequently, different scores for the same concept have substantially different effects on the SDS-POMDP's belief distribution. Accurate confidence measures, therefore, are critical for good system performance. This chapter focuses on the development and evaluation of the utterance confidence scoring model used in the SDS-POMDP.

4.1 Problem Definition

For the SDS-POMDP, we seek a *probability of correctness* of the concept hypothesis, z_d. This can be expressed as $P(z_d = z_g|w)$, where z_d is the concept of the top ASR hypothesis, z_g is the concept of the ground-truth utterance, and w is the output of a speech recognizer. The output w is in the form of ranked hypotheses in an n-best list, each with acoustic and language model scores. A sample output from the SUMMIT speech recognizer is shown in Section 3.4.2.

A few notes about the design of the confidence-scoring module are warranted. The experiments and system development discussed in this chapter assign a confidence score to the top hypothesis in the n-best list. Furthermore, this confidence score seeks to reflect the probability that this hypothesis concept accurately maps to the ground truth concept, not that each word is correct or that the utterance as a whole has been correctly recognized; we are assigning a probability of correctness at the *concept* level. It is worthwhile to note that other formulations of the problem are possible; for example, one could seek to assign confidence scores to every hypothesized in the n-best list, or even every observation in the SDS-POMDP's observation space based on the data. More generic word-based or utterance-based confidence scoring might also have been employed, and the results could have been used to inform a concept confidence score. However, given the required task (assigning a confidence score for a given hypothesis concept) and the data available for this problem, the approach taken was deemed most viable.

For this task, 2701 utterances from 17 different individuals in the lab and at The Boston Home were used. Each of these utterances were labeled with their ground-truth orthography (transcript) and processed by the SUMMIT speech recognizer. Each of the hypotheses, with associated acoustic and language model scores, are used as individual data points in the classification task, leading to 25869 (up to 10 hypotheses per utterance) data points that are divided into a training/test set by a 90/10 split.

It is worthwhile to comment on the use of 10 hypotheses per utterance and treating them as separate data points. This is an unusual setup for at least two reasons: First, the way that the problem has been defined is to classify the top hypothesis. Second, because the hypotheses are drawn from the same underlying utterance, they cannot be considered independent observations. We decided to use all of the hypotheses from each utterance was done in order to create a balanced dataset with a roughly equal number of positive and negative training examples; using only the top hypotheses would have resulted in about 80% of the training examples being correct, and sub-sampling from the set of correct examples to create a balanced set of positive and negative examples would have resulted in an even smaller training set. While collecting more data from laboratory participants or other sources might be inexpensive, collecting more in-domain data from TBH participants would have been relatively costly and time-consuming. Importantly, the ten hypotheses from any given data point were either all assigned to the training set or all assigned to the test set in learning the classifier; therefore, no training examples are part of the test set.

4.2 Feature Engineering

Given the high dimensionality of any output w, a more tractable approach is to extract features from w and learn confidence scores based on these feature values. Potential features were identified manually by examining the outputs of correctly and incorrectly recognized utterances. These features were identified from preliminary data exploration and existing work in the literature on confidence scoring [32, 30, 60]. For example, a high-level of diversity in the n-best list, the presence of many stop words in an utterance, or the presence of words that were often confused often seemed to suggest that the hypothesis was incorrect. Features were computed from the ranked n-best lists of each of the utterances and corpus-level information from the training set. The features that were eventually selected by the AdaBoost classifier are described in more detail below.

Concept-level features: The hypothesized concept itself could provide important hints as to whether the hypothesis is correct or incorrect. These features include:

- `asr_classifier` is an binary feature that simply indicates whether the hypothesis maps to the "null" concept; since these hypotheses are, by definition, incorrect, all of the data points with a value of 0 for this feature should be classified as incorrect.

ASR outputs: The log probabilities of the acoustic, language model, and total scores provide some signal related to the correctness of a hypothesis. Useful features include:

- `one_two_difference_total`: The difference between the total log probability score of the top hypothesis and the total log probability score of the second-ranked hypothesis in the n-best list.

- `normalized_lm_score`: The language model log probability score of the hypothesis divided by the number of words in the hypothesis. Normalizing the score in this manner removes the correlation between the sentence length and the language model log probability, since this language model score is simply the sum of the log probabilities of the trigram scores in the language model.

- `one_two_difference_lm`: Similar to `one_two_difference_total`, the value of this feature is the difference between the language model log probability scores of the top- and second-ranked hypotheses.

Sentence- and word-level features: The "grammaticality" of the sentence, the number of repeated or stop words, or the count of different part-of-speech tags in a sentence could be useful features. Some data exploration using the Stanford parser on the dataset to automatically determine whether sentences were grammatical were not successful, perhaps because many hypotheses with the correct concept were not strictly grammatically correct, or many hypotheses with the incorrect concept were grammatically valid. However, taking counts of different parts of speech, such as adjective phrases, noun phrases, and verb phrases, seemed to be more promising. These features include:

- `stop_words_fraction`: The fraction of words in a hypothesis that are short function words ("stop words"), including "the", "is", "at", "and", "which" and "on" is used as a feature of correctness. Hypotheses that have a high fraction of stop words might be more likely to be incorrect; for example, it seems likely that the hypothesis "it the time of the" (with a `stop_words_fraction` of 0.8) may be incorrect.

- `multiple_days_count`: This feature counts the number of "day" words (e.g. "today", "tomorrow", "monday", "tuesday", ...) in a hypothesis. A higher number of day words, particularly two or more, might suggest that this is a speech recognition error. For example, the hypothesis "what is lunch monday wednesday" would have a value of 2 for this feature.

n-best list features: A number of features can be extracted from the grouping of utterances (up to ten ranked hypotheses in our configuration). The difference in scores between the top hypothesis and the other hypotheses or the purity or diversity of concepts or words in the n-best list at a concept or word level are among the many possible statistics that can be computed.

- `nbest_rank`: The rank, from 1 to 10, of the hypothesis in question within its n-best list is used as a feature. Intuitively, we might expect lower-ranked hypotheses to tend to be incorrect more frequently than higher-ranked hypotheses.

- `category_and_action_entropy`: The diversity of the n-best list is expressed in terms of its entropy, defined as $-\sum_i p(x_i)\log_{10}(p(x_i))$, where x_i is a concept and $p(x_i)$ represents the "probability" of that concept in the n-best list. This feature takes on a value of 0 if all 10 hypotheses have the same concept and 1 if every hypothesis maps to a different concept.

- `log_hypothesis_high_purity_fraction`: This feature is computed from the fraction of the n-best list that is comprised of the most popular concept. For example, if 6 out of 10 hypotheses map to the same concept, then the logarithm of 0.6 is used (the log probability is used for numerical reasons for AdaBoost).

- `current_hypothesis_nbest_purity`: The fraction of hypotheses that map to the same concept as that of the hypothesis being considered is used for this feature.

- `log_total_score_fraction`: The total log probability score for the given hypothesis is expressed as a fraction of the sum of all log probability scores in the n-best list and used as a feature.

Corpus-level features: A number of summary features on the training set as a whole are also used as features. For instance, the most common words that were incorrectly inserted, missed, merged, or split could be determined from the recognition results on the training set, and the presence of these words was used as a feature.

- `merged_word_pairs` and `split_word_pairs`: SUMMIT computes summary statistics on the words that were merged and split most often. The number of these words were counted for each of these features.

- `speaker`: An indicator function corresponding to whether the speaker was from TBH, the target population, was used. This information should be readily available in the test set because each dialog system is configured for a specific setting and person.

4.3 Feature Selection with AdaBoost

To assign confidence scores to the data, a two-step process was used. First, using AdaBoost [26], a classifier for correctly and incorrectly hypothesized utterances was trained. Although the ultimate goal of this supervised learning task is regression (*i.e.*, to learn a function that maps features to a meaningful confidence score), AdaBoost offers an attractive framework for selecting relevant features. Specifically, AdaBoost is an iterative algorithm that combines a collection of weak classifiers to produce a single, strong classifier. At each iteration, a weak classifier is chosen and assigned a weight. Critically, the loss function that AdaBoost seeks to minimize at each iteration places more weight on datapoints incorrectly classified by the existing collection of classifiers.

The trained AdaBoost classifier assigns a positive or negative prediction to each datapoint, based on the sign of the sum of weighted weak classifiers. However, for regression, the score itself can be used. A logistic curve was fit to the AdaBoost scores to the labeled data. The evaluated logistic function is interpreted by the SDS-POMDP as the confidence score.

It is worth noting that more features than those explained in the previous section were evaluated by AdaBoost. An attractive property of this algorithm, though, is that AdaBoost only selects those features that have an impact in reducing the weighted classification error.

4.4 Classifier Performance

Figure 4-1 shows the performance of AdaBoost on the training and test set at each iteration. The performance on the test set reaches a minimum error rate of 6.9%. Table 4.1 lists the features selected by AdaBoost at each iteration. The AdaBoost classifier greedily chooses the feature and threshold at each iteration that minimizes the weighted classification error. Some important features include the concept classifier itself ("null" observations are classified as incorrect), the "purity" of the n-best list (the fraction of hypotheses that map to the same concept), and the difference in the total ASR score between the top and second hypothesis.

Figure 4-2 illustrates the fit of the logistic curve to the training data. The function representing this curve is used by the confidence scoring module in the SDS-POMDP.

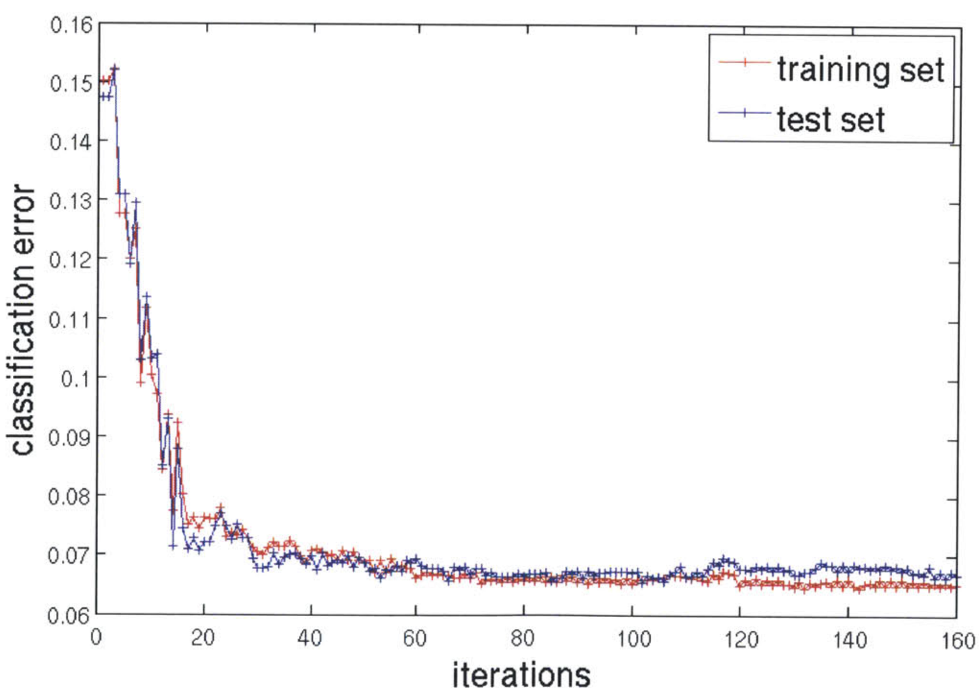

Figure 4-1: Performance of AdaBoost by iteration on training and test sets

Table 4.1: Features selected by AdaBoost

		Training Set		Test Set	
Iteration	Selected Feature	FPR	FNR	FPR	FNR
0	asr_classifier	0.00%	15.02%	0.00%	14.75%
1	current_hypothesis_nbest_purity	0.00%	15.02%	0.00%	14.75%
2	log_hypothesis_high_purity_fraction	5.43%	9.78%	5.56%	9.65%
3	one_two_difference_total	4.73%	8.03%	5.12%	7.97%
4	asr_classifier	4.73%	8.03%	5.12%	7.97%
5	nbest_rank	3.93%	8.08%	4.01%	7.91%
6	current_hypothesis_nbest_purity	4.55%	7.95%	4.89%	8.06%
7	normalized_lm_score	3.03%	6.87%	3.21%	7.09%
8	one_two_difference_lm	4.69%	6.49%	5.03%	6.33%
9	asr_classifier	2.70%	7.34%	2.77%	7.55%
10	category_and_action_entropy	2.91%	6.81%	3.17%	7.22%
11	current_hypothesis_nbest_purity	4.41%	4.03%	4.78%	3.72%
12	one_two_difference_total	3.86%	5.50%	4.54%	4.76%
13	log_hypothesis_high_purity_fraction	3.40%	4.35%	3.96%	3.19%
14	asr_classifier	3.21%	6.02%	3.76%	5.03%
15	one_two_difference_total	3.37%	4.65%	3.96%	3.48%
16	current_hypothesis_nbest_purity	3.06%	4.46%	3.30%	3.81%
17	merged_word_pairs	3.40%	4.23%	3.99%	3.30%
18	asr_classifier	2.64%	4.81%	2.99%	4.10%
19	category_and_action_entropy	3.38%	4.26%	3.94%	3.28%
20	current_hypothesis_nbest_purity	2.88%	4.72%	3.10%	4.12%
21	split_word_pairs	2.35%	5.25%	2.79%	4.69%
22	speaker	2.96%	4.83%	3.23%	4.47%
23	current_hypothesis_nbest_purity	2.43%	4.89%	3.01%	4.47%
24	one_two_difference_lm	2.86%	4.53%	3.34%	3.92%

Continued on next page

		Training Set		Test Set	
Iteration	Selected Feature	FPR	FNR	FPR	FNR
25	asr_classifier	2.33%	5.00%	2.83%	4.67%
26	category_and_action_entropy	2.93%	4.49%	3.37%	3.92%
27	split_word_pairs	2.40%	4.88%	3.06%	4.23%
28	current_hypothesis_nbest_purity	2.58%	4.55%	3.28%	3.65%
29	log_hypothesis_high_purity_fraction	2.49%	4.56%	3.14%	3.63%
30	normalized_lm_score	2.48%	4.53%	2.86%	3.92%
31	one_two_difference_total	2.46%	4.67%	3.01%	3.79%
32	one_two_difference_lm	2.44%	4.78%	2.88%	4.16%
33	one_two_difference_acoustic	2.35%	4.80%	2.97%	3.88%
34	normalized_acoustic_score	2.75%	4.39%	3.19%	3.76%
35	asr_classifier	2.26%	4.98%	2.83%	4.16%
36	one_two_difference_total	2.78%	4.37%	3.23%	3.81%
37	current_hypothesis_nbest_purity	2.30%	4.67%	2.99%	3.94%
38	log_hypothesis_high_purity_fraction	2.44%	4.42%	2.99%	3.85%
39	one_two_difference_lm	2.31%	4.77%	2.92%	4.05%
40	merged_word_pairs	2.33%	4.77%	2.83%	3.92%
41	current_hypothesis_nbest_purity	2.49%	4.57%	3.37%	3.68%
42	stop_words_fraction	2.40%	4.59%	3.14%	3.68%
43	multiple_days_count	2.49%	4.52%	3.21%	3.65%
44	one_two_difference_total	2.50%	4.41%	3.34%	3.57%
45	asr_classifier	2.36%	4.72%	3.01%	3.85%
46	category_and_action_entropy	2.62%	4.30%	3.43%	3.54%
47	asr_classifier	2.35%	4.70%	3.06%	3.74%
48	category_and_action_entropy	2.58%	4.32%	3.41%	3.54%
49	normalized_acoustic_score	2.45%	4.41%	3.28%	3.59%
50	one_two_difference_acoustic	2.39%	4.52%	2.88%	3.85%
51	log_total_score_fraction	2.55%	4.15%	3.32%	3.45%

Continued on next page

		Training Set		Test Set	
Iteration	Selected Feature	FPR	FNR	FPR	FNR
52	asr_classifier	2.36%	4.56%	2.70%	3.92%
53	log_total_score_fraction	2.48%	4.23%	3.26%	3.50%
54	current_hypothesis_nbest_purity	2.36%	4.58%	2.77%	3.96%
55	normalized_lm_score	2.21%	4.58%	2.97%	3.85%
56	split_word_pairs	2.51%	4.31%	3.08%	3.65%
57	current_hypothesis_nbest_purity	2.21%	4.56%	2.90%	4.01%
58	one_two_difference_lm	2.44%	4.33%	3.08%	3.79%
59	normalized_total_score	2.28%	4.34%	2.99%	3.94%
60	hypothesis_average_keyword_purity	2.42%	4.23%	2.99%	3.83%
61	asr_classifier	2.19%	4.52%	2.72%	4.05%
62	normalized_acoustic_score	2.45%	4.21%	2.99%	3.79%
63	asr_classifier	2.20%	4.48%	2.72%	4.05%
64	one_two_difference_total	2.45%	4.22%	2.95%	3.79%
65	split_word_pairs	2.27%	4.39%	2.90%	3.68%
66	current_hypothesis_nbest_purity	2.45%	4.16%	3.12%	3.68%
67	normalized_lm_score	2.35%	4.27%	3.03%	3.72%
68	merged_word_pairs	2.46%	4.22%	3.06%	3.74%
69	number_of_words	2.39%	4.23%	3.03%	3.65%
70	one_two_difference_total	2.36%	4.35%	3.01%	3.76%
71	log_total_score_fraction	2.47%	4.06%	3.21%	3.57%
72	one_two_difference_total	2.35%	4.25%	2.97%	3.70%
73	one_two_difference_lm	2.47%	4.09%	3.19%	3.52%
74	stop_words_count	2.34%	4.24%	2.95%	3.70%
75	split_word_pairs	2.46%	4.08%	3.12%	3.52%
76	asr_classifier	2.29%	4.32%	2.81%	3.83%

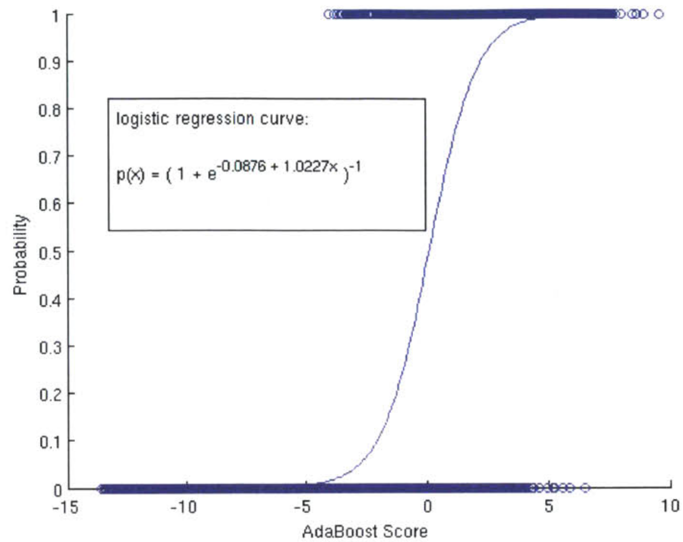

Figure 4-2: Fitting of logistic curve to AdaBoost scores

4.5 Limitations

Many of the features that contribute to the confidence score would likely be useful in other ASR applications. However, it is also important to note that some of the features used to improve results for this specific task might not be directly applicable for other domains. For example, the labels of correctness for this classification task are based on whether the hypothesis, using a basic keyword extractor, maps to the correct concept; other confidence-scoring modules might require more stringent definitions of correctness at the word or utterance level. In addition, the use of corpus-level features restricts the transferability of the confidence scoring module to other domains with different language models.

As Figure 4-1 shows, 6.6% of the test set utterances remain incorrectly classified. Characterizing the errors that remain could be useful to identify other informative features. Currently, for example, an utterance with a highly consistent-but-incorrect n-best list is vulnerable to being classified incorrectly, due to the importance of low concept entropy in the dataset. Meanwhile, the use of dialog-level features, such as the turn number and the results of the previous utterance, might also be useful features; this would require data from actual dialogs.

Finally, a number of extensions to the behavior and capability of the confidence-scoring module could be useful for the dialog task. First, in the current problem definition, a confidence score is assigned to a single hypothesis concept. Assigning confidence scores to multiple concepts, however, could be useful in updating the belief of the SDS-POMDP. Second, this framework assigns a single confidence score to the entire hypothesis. Word-level confidence scoring could also be useful in updating the belief of the SDS-POMDP. For example, in an utterance where the hypothesis is "what is the weather on Wednesday?", if a high confidence score were assigned to the word "weather" in a hypothesis but a lower confidence score were assigned to the word "Wednesday", the observation model might distribute probability mass to "weather" states corresponding to different days.

Chapter 5

User Simulation

Evaluating the effectiveness of a spoken dialog system involves addressing several challenges. Ideally, human users should use and evaluate the end-to-end system. However, because spoken dialogs can involve multiple turns, conducting such experiments can be time-consuming and expensive. As well, it is unclear whether data from a set of dialog experiments with one set of parameters is applicable to different sets of parameter configurations, especially because a dialog consists of a user's utterances, the system's responses, and the user's reaction to those responses.

This chapter describes a user simulator that was used to evaluate the effectiveness of the SDS-POMDP compared to other dialog strategies. As discussed in Chapter 6, this user simulator is also used to identify optimal reward function parameters for the SDS-POMDP.

5.1 Design of User Simulator

The observation model of the SDS-POMDP, $P(Z|S, A)$, can be interpreted as a generative model of the user's behavior: It encodes the probability of any observation, given the user's goal and the most recent system action. Therefore, we can sample a hypothesis concept and a confidence score for a given state and system action and provide this observation to the dialog manager. The procedure is as follows:

1. The user simulator samples a hidden state, s^*, from a distribution over states

$P(S)$. This mimics a real user who wishes to interact with the system to achieve his or her goal (*e.g.* learn the time, make a phone call, or ask for Thursday's weather).

2. The user simulator samples an observation from the distribution $P(Z|S = s^*, A)$ If this is the first simulated utterance in the dialog, $A =$ GREET, which corresponds to the start of the dialog. The interpretation of this step is that the user speaks an utterance based on his or her goal and the fact that the system has just performed its initial GREET action. The distribution over observations is the observation model learned from training data, meaning that the simulator may either choose an observation that maps correctly to the user's state, which simulates a correctly recognized utterance, or an observation that maps to another meaning, which mimics an incorrectly recognizer utterance.

3. Along with a hypothesis, the simulated observation requires a confidence score. This confidence score is sampled from the appropriate probability distribution shown in Figure 3-1, as dictated by Equation 3.1: A selected observation hypothesis that "corresponds" to the goal state is more likely to have a high confidence score, while a mismatching hypothesis concept is more likely to have a low confidence score.

4. Together, the simulated observation hypothesis and confidence score comprise the user input into the dialog manager. In the case where the dialog manager is the SDS-POMDP, this observation is used to update the belief, just as the SDS-POMDP would behave with an actual utterance. The new belief is used by the policy of the SDS-POMDP, $\Pi(b)$ to select an action, a'. Notably, however, we can use *any* dialog manager with this model, making it possible to compare the performance of different dialog strategies or parameter settings. In the case of a threshold-based dialog manager, for example, the policy would be determined solely by the observation and confidence score of the previous observation.

5. Steps 2-4 are repeated until the dialog manager performs the correct system

action (*i.e.* fulfills the user's goal). At each turn, the user simulator selects an observation based on the distribution $P(Z|S = s*, A = a')$.

With this approach it becomes possible to simulate many dialogs rapidly and to compute performance statistics, including the number of turns, the number of correct or incorrect dialogs, and the time required per dialog. It is also possible to compare the SDS-POMDP against other dialog managers (as discussed below) and to use the user simulator to optimize the SDS-POMDP's behavior (as discussed in Chapter 6).

5.2 Candidate Dialog Managers

Using the simulator, three dialog strategies are compared against the full SDS-POMDP model developed in this thesis:

- The `argmax` dialog manager simply accepts the first (non-null) concept uttered by the user.

- The `threshold` dialog manager considers the confidence score in deciding to accept or reject the hypothesized concept in the utterance. There are two possible outcomes from any observation: it either does not meet the threshold and the "GREET" system action is repeated, or it meets or exceeds the threshold and a terminal action is submitted.

- The `POMDP-no-confidence-scores` dialog manager computes belief updates in the SDS-POMDP without the use of confidence scores. The system may ask the user to repeat, confirm a concept, or submit a terminal action.

These dialog managers are compared against the `POMDP-with-confidence-scores` experimental condition, which uses the full SDS-POMDP with confidence scoring of utterances, as detailed in Chapters 3 and 4. A total of 50000 dialogs are simulated for each dialog manager.

5.3 Results and Discussion

Table 5.1 compares the performance of the different dialog managers described above. Dialogs are marked as "incorrect" if a terminal action is taken that does not correspond to the user's simulated goal.

Table 5.1: Performance of dialog managers over 50,000 simulated dialogs

Dialog Strategy	Correct Dialogs	Incorrect Dialogs	Dialog Error Rate	Mean Turns per Dialog	Standard Error on Number of Turns
argmax(State) (no threshold)	45920	4080	8.16%	1.00	0.00
Confidence threshold (thr=0.50)	46589	3411	6.82%	2.60	0.07
Confidence threshold (thr=0.85)	46976	3024	6.05%	5.44	0.14
Confidence threshold (thr=0.95)	46179	3821	7.64%	10.39	0.17
POMDP-no-confidence-scores	48022	1978	3.96%	1.23	0.00
POMDP-with-confidence-scores	49421	579	1.16%	1.95	0.01

5.3.1 Threshold-Based Dialog Managers

The threshold-based dialog managers shown in Table 5.1 exhibit a trade-off between the mean number of turns and the number of incorrect dialogs. The main trend is that, as the the threshold increases, the percentage of correctly completed dialog increases, but so does the required number of turns to complete each dialog. At one extreme (the `argmax` dialog manager), having no confidence threshold minimizes the number of turns at the expense of generating more incorrect dialogs; at a threshold of 0.95, meanwhile, a high mean number of turns (10.39 ± 0.17) was needed.

Notably, the trade-off between the number of turns and the dialog error rate does not simply change monotonically over the entire range of thresholds; for instance, thresholds of 0.50 and 0.85 both yielded better performance than the threshold dialog manager with a confidence threshold of 0.95 — they both had a higher number of correct dialogs and fewer mean turns. Interestingly, this result can also be explained analytically: it turns out that the behavior of the user simulator can be explained by a simple Markov chain, as shown in Figure 5-1. In the experiment, after the simulator selects a goal $s*$, it chooses an observation based on the probability distribution $P(z|s*, a = \text{GREET})$. Then, depending on the sampled observation, a confidence score is sampled either from the "correct" confidence-score distribution or the "incorrect" confidence-score distribution. Because the threshold is fixed and the confidence-score distributions are pre-defined, there is a fixed probability for each distribution that a confidence score above the threshold will be sampled. If the selected confidence score is selected, the dialog terminates (in one of the shaded states); if not, then the Markov chain returns to the initial state. The expected probabilities of ending up in each of two shaded terminal states can be easily calculated with basic infinite-series formulas. It turns out that these expected values can change non-monotonically because the confidence-score distributions that we sample differ depending on whether the observation we are choosing an observation that maps to the correct state or the incorrect state.

5.3.2 POMDP-based Dialog Managers

As shown above, the `POMDP-with-confidence-scores` dialog manager outperforms the other dialog strategies. By choosing actions based on the SDS-POMDP's belief, the dialog manager makes better decisions about when to repeat the original "GREET" question, ask a confirmation question, or submit a final terminal action. Interestingly, the `POMDP-no-confidence-scores` dialog manager, which simulates user utterances but does not sample confidence scores, performs substantially worse;

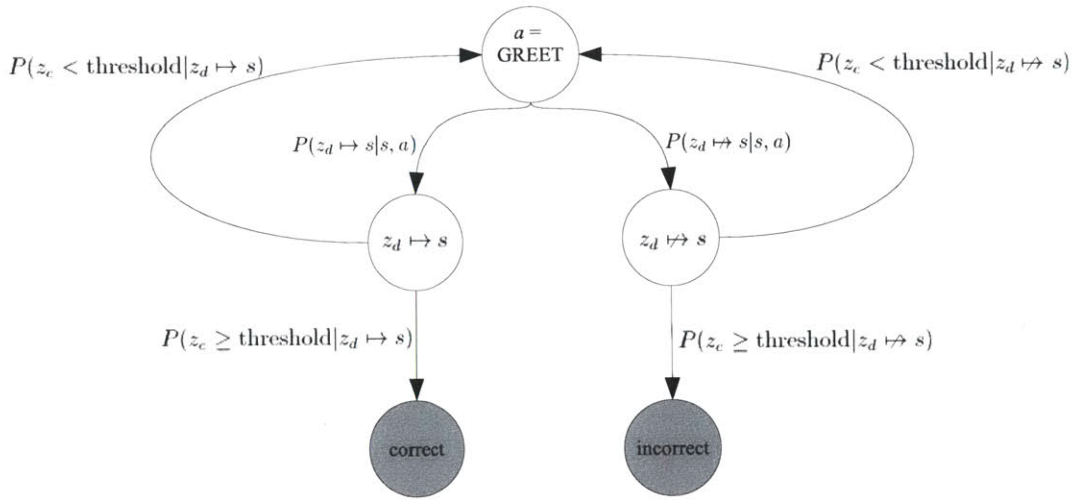

Figure 5-1: Markov chain model of user simulator with threshold-based dialog managers

in fact, many of the threshold-based dialog managers perform better. This is because mismatched simulated observations are not assigned low confidence scores; thus, such observations can result in a belief in which high probabilities are assigned to states that do not correspond with the true user goal.

5.3.3 Limitations

It is worthwhile to note that the threshold-based dialog managers studied do not include the concept of confirmation questions in this set of comparison experiments. A second, lower "confirmation" threshold could be assigned in which a confirmation question is asked if the simulated confidence score exceeded this threshold but did not reach the higher "submit" threshold. However, this second threshold would not be the only parameter that would need to be fixed for a certain experimental condition. For example, we would need to decide how the dialog manager would act in response to another observation for the same hypothesis concept that does not exceed the "submit" threshold. Should the dialog manager ask the confirmation question again, submit a "terminal action" because the observation has been repeated more than once, or somehow combine the two confidence scores (perhaps in a Bayesian manner)? If the next observation corresponded to a different hypothesis, but also was in the "confirmation" range of confidence scores, which confirmation question should the system ask next? Although handcrafted rules could be generated for all of these different scenarios, these questions in themselves motivate the formulation of the SDS-POMDP, which maintains a belief over different states and determines "thresholds" for confirming and submitting terminal actions based on learned models of hypotheses and confidence scores to states.

An important question that arises from the user simulator is whether it truly is an accurate representation of the user, and how closely the simulation results will reflect actual user behavior. First, it is important to note that the user simulator chooses observations based on the model $P(Z|S, A)$, where S and A are the user's intent and the immediately preceding system action, respectively. Evidently, a Markov assumption is made; the simulated observation does not depend on system actions beyond the immediate preceding action or other aspects of the dialog history. In reality, human dialog actors might change the words that they use in response to "unsuccessful" utterances (*e.g.* when there is a recognition error); such dynamics have not been captured in $P(Z|S, A)$. Secondly, meanwhile, it is almost certain that the observation model of individual users will differ from $P(Z|S, A)$ used in the simulations; indeed, as noted in the introduction of this thesis, residents at The Boston Home exhibit wide variability in word error rates. Ultimately, therefore, the results must be compared to dialog data with human participants.

Chapter 6

Reward Parameter Search

As discussed in the previous chapter, the SDS-POMDP with confidence scores appears to be superior to simpler threshold-based dialog managers. However, the results were demonstrated with just one handcrafted SDS-POMDP policy; it is likely that other policies could yield even better results. This chapter focuses on optimizing the behavior of the SDS-POMDP with the aid of the user simulator.

6.1 Solving the POMDP

As a toy example, consider a three-state SDS-POMDP; that is, the user can ask the dialog system for three different concepts. A three-state POMDP is minimally required so that confirmation questions are preferred in some beliefs and generic requests for information (greeting the user or asking the user to repeat) are preferred in others. For this section of the thesis, we assign numerical values that are reasonable approximations of the structure of the actual SDS-POMDP and make it easier to convey the key ideas associated with solving the POMDP through the Q-MDP approximation [48]. We can now form a finite-world POMDP, with a finite number of states, actions, and observations.

6.1.1 Enumerating POMDP States and Functions

States, Actions, and Observations. The state space S consists of three states, denoted by s_1, s_2, and s_3. The action space A consists of seven actions, including "submit" actions corresponding to each state (a_{1s}, a_{2s}, a_{3s}), "confirmation" questions corresponding to each state (a_{1c}, a_{2c}, and a_{3c}), and a generic request for information (greeting the user, or asking him or her to repeat), a_r. Finally, there are observations that map to each of the three states, (z_1, z_2, and z_3), "yes" and "no" confirmation questions (z_{yes} and z_{no}), and a "null" observation if the speech hypothesis does not map to any of these observations.

Transition Function. For the transition function, we assume that the user's goal does not change over the duration of the dialog. Therefore, we can write:

$$P(s|s', a) = \begin{cases} 1 & \text{if } s' = s \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

In other words, the transition function is independent of the system action.

Observation Function. The observation function, $P(Z|S, A)$, must be defined for all observations for each state-action pair. First, we consider the distribution over observations for the initial system action, a_r, and state s_1. This set of values fully defines the conditional probability distribution for (s_1, a_r); the probabilities sum to 1.

$$P(z_1|s_1, a_r) = 0.7 \tag{6.2}$$

$$P(z_2|s_1, a_r) = P(z_3|s_1, a_r) = 0.05 \tag{6.3}$$

$$P(z_{\text{yes}}|s_1, a_r) = P(z_{\text{no}}|s_1, a_r) = 0.025 \tag{6.4}$$

$$P(z_{\text{null}}|s_1, a_r) = 0.15 \tag{6.5}$$

The interpretation of the above values is that the speaker has a 0.7 probability of an utterance that maps to the correct state, with the remainder of the probability mass assigned to the other possible observations. For completeness, the observation functions for the other two hidden states are listed:

$$P(z_1|s_2, a_r) = P(z_3|s_2, a_r) = 0.05 \tag{6.6}$$

$$(z_2|s_2, a_r) = 0.7 \tag{6.7}$$

$$P(z_{\text{yes}}|s_1, a_r) = P(z_{\text{no}}|s_1, a_r) = 0.025 \tag{6.8}$$

$$P(z_{\text{null}}|s_1, a_r) = 0.15 \tag{6.9}$$

$$P(z_1|s_3, a_r) = P(z_2|s_3, a_r) = 0.05 \tag{6.10}$$

$$P(z_3|s_3, a_r) = 0.7 \tag{6.11}$$

$$P(z_{\text{yes}}|s_1, a_r) = P(z_{\text{no}}|s_1, a_r) = 0.025 \tag{6.12}$$

$$P(z_{\text{null}}|s_1, a_r) = 0.15 \tag{6.13}$$

The confirmation questions are a kind of "sensing" system action; they provide information about the system state. For example, if the underlying system state is s_1 and the action is a_{1c}, the probability mass function is assigned as follows:

$$P(z_1|s_1, a_{1c}) = 0.4 \tag{6.14}$$

$$P(z_2|s_1, a_{1c}) = P(z_3|s_1, a_{1c}) = 0.04 \tag{6.15}$$

$$P(z_{\text{yes}}|s_1, a_{1c}) = 0.4 \tag{6.16}$$

$$P(z_{\text{no}}|s_1, a_{1c}) = 0.02 \tag{6.17}$$

$$P(z_{\text{null}}|s_1, a_{1c}) = 0.1 \tag{6.18}$$

$$\tag{6.19}$$

An important subtlety exists in the above probability distribution for the confirmation question a_{1c}: we assign equally high probability to the observation z_1 (which maps to s_1) and the observation z_{yes}. This means that when the SDS-POMDP asks a confirmation question, the user tends to confirm the question by answering "yes" or by providing an observation that maps to the goal state. The probability distribution for the other goal states follows a similar structure, in that corresponding conditional probability distributions exist for states s_2 and s_3.

Another important probability distribution is when the incorrect confirmation question is asked; that is, the SDS-POMDP asks a confirmation question that does not correspond to the user's goal:

$$P(z_1|s_1, a_{2c}) = 0.4 \tag{6.20}$$

$$P(z_2|s_1, a_{2c}) = 0.04 \tag{6.21}$$

$$P(z_3|s_1, a_{2c}) = 0.04 \tag{6.22}$$

$$P(z_{\text{yes}}|s_1, a_{1c}) = 0.02 \tag{6.23}$$

$$P(z_{\text{no}}|s_1, a_{1c}) = 0.4 \tag{6.24}$$

$$P(z_{\text{null}}|s_1, a_{1c}) = 0.1 \tag{6.25}$$

Similar to the probability distribution $P(\cdot|s_1, a_{1c})$, there is a relatively high probability assigned to observation z_1 when the true goal is s_1 because the user can provide either a "no" utterance or an utterance that maps to the true goal.

Finally, there is a class of distributions in the observation function when the action is a terminal action, *i.e.* a_{1s}, a_{2s}, a_{3s}. These take on essentially the same form as the distributions conditioned on a_r.

The above equations for $P(Z|S, A)$ omit the continuous component of the observation model, namely the confidence scores as discussed in Chapter 3. Having a continuous observation function increases the complexity of solving the POMDP, making approximate methods more attractive.

Reward function. The reward function is defined on all state-action pairs. The reward function encodes a positive reward for submitting the correct terminal action and a large negative reward for submitting the incorrect terminal action. The information-gathering actions are also given smaller negative rewards. The "correct" confirmation question has a reward that is less negative than the initial system action, a_r, while the "incorrect" confirmation question has a higher-magnitude negative reward. As a initialization point for our search over parameters, we use, for state s_1:

$$R(s_1, a_{1s}) = +10 \tag{6.26}$$

$$R(s_1, a_{2s}) = R(s_1, a_{3s}) = -100 \tag{6.27}$$

$$R(s_1, a_{1c}) = -5 \tag{6.28}$$

$$R(s_1, a_{2c}) = R(s_1, a_{3c}) = -15 \tag{6.29}$$

$$R(s_1, a_r) = -10 \tag{6.30}$$

The reward function is similar for states s_2 and s_3, with high positive rewards for the correct terminal action and a lower-magnitude negative rewards for the corresponding confirmation question in each of these states.

6.2 Approximately Solving the POMDP

With the full SDS-POMDP specified, a solution method needs to be chosen. As discussed above, the continuous nature of the observations means that methods that make use of the fact that the observation space is finite and discrete cannot be used. To incorporate the confidence score function, Williams and Young [68] use a method of POMDP policy optimization that admits continuous observations in a dialog system with close to 2000 states [35]. This implies that a similar method can be used for the 62-state SDS-POMDP discussed in this thesis. A useful extension of this thesis might be a comparison of these solutions to the QMDP-derived solution discussed in this chapter.

As an alternative, it is interesting to examine whether simpler approximately POMDP solution techniques of the SDS-POMDP can be used to generate reasonable policies. In particular, the QMDP method that uses the value functions of the underlying Markov decision process (MDP). QMDP makes the assumption that the state is fully observable after one control step; as a result, it becomes possible to use the value functions, $\hat{V}(s)$, of the underlying MDP to compute a policy [48]. Specifically, the QMDP algorithm computes a function Q for each state-action pair,

$$Q(s_i, a) = R(s_i, a) + \sum_{j=1}^{N} \hat{V}(s_i) P(s_j | s_i, a) \tag{6.31}$$

for each system action, a. Then, for a belief state $b = (p_1, p_2, ...p_N)$, where p_i corresponds to the probability mass in state i, the policy is simply

$$\Pi(b) = \arg\max_a \sum_{i=1}^{N} p_i Q(s_i, a) \tag{6.32}$$

6.2.1 Structure of the QMDP-Generated Policy

It is interesting to examine how the parameters of the reward function affect the QMDP-generated policy. First, the underlying MDP's value function is critical for the QMDP algorithm. The value function $\hat{V}(s)$ is initialized to $r_{\min}$ and then computed iteratively until convergence; for completeness, this procedure is shown below as Algorithm 1.

Algorithm 1 MDP Value Iteration

Input: $R(S, A), T(S|S, A)$

```
for all s do
    V̂(s) = r_min
end for
repeat
    for all s do
        V̂(s) = γ max_a(r(s, a) + Σ_{j=1}^{N} V̂(s_j) p(s_j|s_i, a))
    end for
until convergence
return V̂
```

In fact, given that the transition function $P(S'|S, A)$ is deterministic in our SDS-POMDP, the steady-state, converged value of $\hat{V}$ for all states is straightforward to calculate with the given reward function and $\gamma = 0.95$:

$$\hat{V}(s) = R_{\min} = -100 \tag{6.33}$$

$$\hat{V}(s_i) = \gamma \max_a [R(s_i, a) + \sum_{j=1}^{N} \hat{V} p(s_j | s_i, a)] \tag{6.34}$$

$$= \gamma [R(s_i, a_{\text{submit i}}) + \hat{V}(s_i)] \tag{6.35}$$

$$\hat{V}(s_i) - \gamma \hat{V}(s_i) = 10\gamma \tag{6.36}$$

$$\hat{V}(s_i) = 10\gamma/(1-\gamma) \tag{6.37}$$

$$= 190 \tag{6.38}$$

The value function can now be used in the QMDP algorithm for computing a policy for the SDS-POMDP, as given by Equation 6.32. One method of examining the behavior of the SDS-POMDP is to plot the expected value function for different belief distributions. In particular, a class of distributions in an N-state SDS-POMDP is one in which one state has most of the probability mass, while the remaining probability mass is distributed among other states, as follows:

$$b(s) = \begin{cases} p^* & \text{if } s = s^* \\ (1-p^*)/(N-1) & \text{if } s \neq s^* \end{cases} \tag{6.39}$$

Figure 6-1 shows such a plot for our three-state example, where $s^* = s_1$. The plots start at $s_1 = 0.33$; by symmetry, the plots would be similar for s_2 and s_3. The curves illustrate the value of the QMDP-derived expected payoff, $\sum_{i=1}^{N} p_i Q(s_i, a)$ for each action, a, meaning that the action corresponding to the line with the highest value is the value of the policy function at that particular belief distribution. The behavior induced by solving the SDS-POMDP using QMDP is reasonable:

- Between $b(s_1) = 0.33$ and a $b(s_1) = 0.5$, the action a_r produces the maximum expected payoff. This means that the SDS-POMDP initially makes a generic request for information (assuming a uniform initial distribution over states) and continues to ask this question so long as the belief in s_1 remains less than 0.5.

- Between $b(s_1) = 0.5$ and $b(s_1) = 0.81$, the action a_{1c} dominates. In this region, the SDS-POMDP asks to confirm whether the user's goal is s_1. This seems sensible; the system asks the "correct" confirmation question when it is "somewhat" sure of the user's intent.

- Above $b(s_1) = 0.81$, the maximum curve corresponds to a_{1s}; in other words, the SDS-POMDP "commits" to goal s_1. This policy also seems reasonable — when the SDS-POMDP's belief is sufficiently confident in one state over all others, it takes the "submit" terminal action for that state.

- It is worthwhile to comment on the curves that are not at the maximum in this part of the belief region. They correspond to the "submit" actions for states s_2 and s_3, a_{2s} and a_{3s}, and the confirmation questions corresponding to these two states, a_{2c} and a_{3c}. The policy decides that, when there is low probability in states 2 and 3, the SDS-POMDP should not submit or confirm either of these two goals.

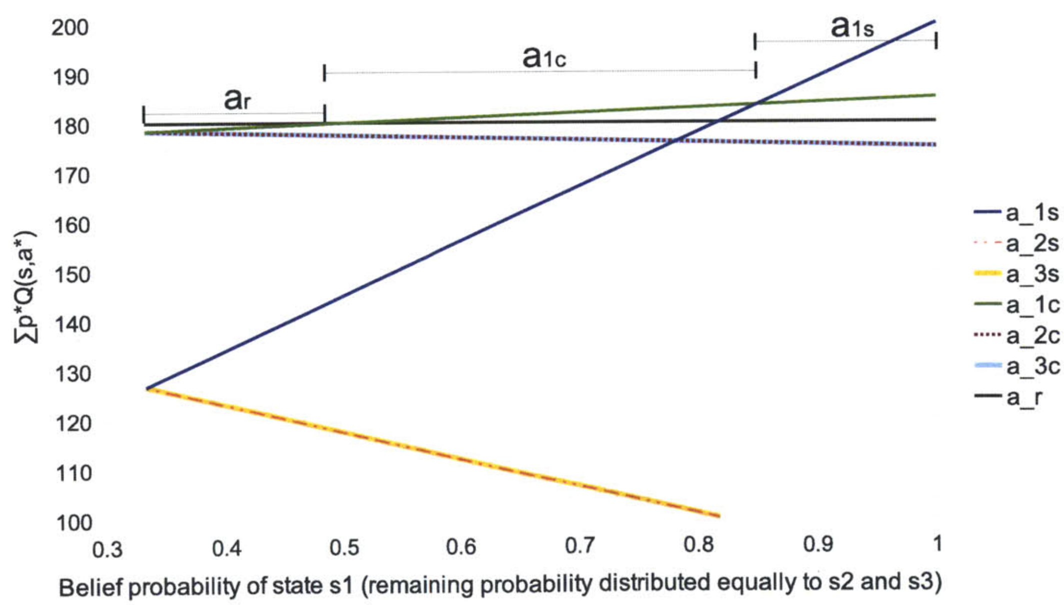

Figure 6-1: QMDP-derived expected payoff of SDS-POMDP actions

Importantly, depending on the reward parameters, the values of $\hat{V}$ and Q change, which affects the cross-over points represented in Figure 6-1. The reward parameters can also have different effects on other behaviors of the system, such as when there are two leading hypotheses. The impacts of these policies on the performance of the SDS-POMDP is the subject of the remainder of this chapter.

6.3 Defining Good Dialog Behavior

A challenge in spoken dialog systems is characterizing the nature of successful dialogs. A dialog's success in achieving the user's goal, total duration, number of turns, or "naturalness" may all be features that determine whether a dialog is "good" or not. Frameworks in the literature, such as PARADISE, consider factors such as task success, efficiency metrics, and qualitative measures, along with surveys of satisfaction based on user evaluations [65]. Even in the user simulator experiments in the previous chapter, it is not necessarily clear what the trade-off between successfully completed dialogs and the mean number of turns might be.

For this set of experiments, a reasonably simple metric is adopted: the wall-clock time required to complete a dialog successfully. Importantly, the dialog is complete only when the user achieves his or her true goal; in other words, when the SDS-POMDP performs an incorrect terminal action, the simulated dialog continues. Intuitively, the best policy should execute a terminal action as soon as SDS-POMDP is sufficiently confident that about the user's goal, but not so soon as to result in many incorrect terminal actions, which would waste time.

Because simulations can be completed much more quickly than actual dialogs, we assign estimated time costs (in seconds) for different categories of actions. These estimated times reflect the amount of time that would be used for the actual action to be taken and for the user to realize that the incorrect action has occurred. Some actions are more costly than others; for example, if the system mistakenly makes a phone call, it could be time-consuming to wait for a connection and end the accidental call. In contrast, showing the activities or weather forecast might require some time to call the relevant API, but should otherwise be less costly. Table 6.1 summarizes the estimated time for each kind of actions. With these estimated time costs, the objective

Table 6.1: Estimated time costs of different user actions

Action Category	Estimated Time (s)
time	10
activities	20
weather	15
menus	15
phone	50
confirmatory question	5
greet user	5

function was to minimize the time required for each dialog. The time required for the final terminal action (the correct one) is excluded to avoid skewing the total time required for actions that inherently take longer periods of time. This can be interpreted as the user feeling "satisfied" once the SDS-POMDP has committed to fulfilling his or her goal.

6.4 Optimization Approach

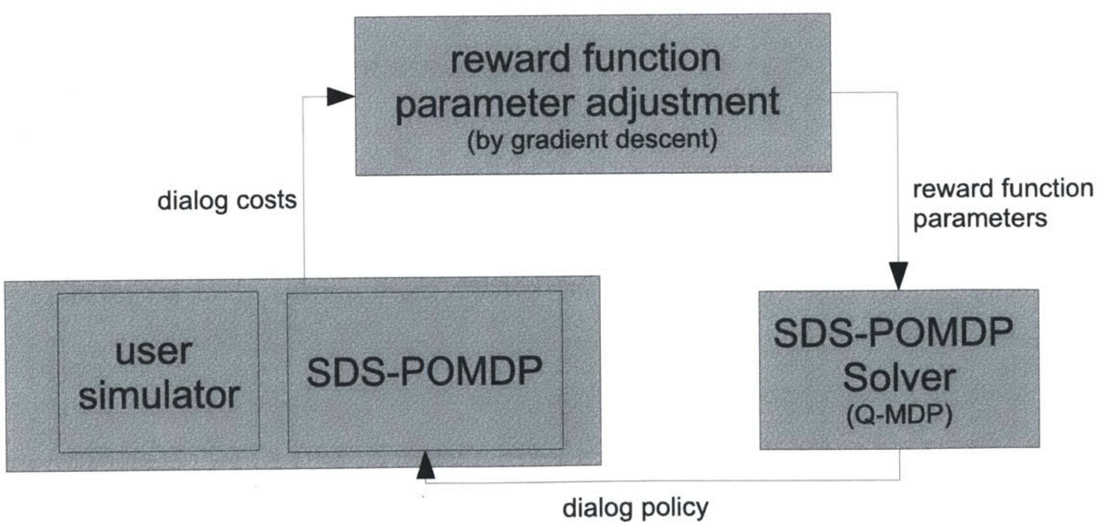

Figure 6-2: Framework for optimizing dialog policy

The general approach that we take is shown in Figure 6.4. The initial reward parameters can be used to compute a policy for the SDS-POMDP (using the Q-MDP approximation); this policy is then used by the SDS-POMDP with 10,000 simulated dialogs and the dialog cost metric. As discussed below, a brute-force search is conducted in order to find regions in the parameter space with good performance; then, gradient descent is used to refine the parameters for further performance improvement.

The reward function $R(S, A)$ is a discrete function defined on every state-action pair (s, a) of the POMDP. Different values can be defined for each unique (s, a), meaning that the search could involve $S\dot{A}$ parameters. However, the number of parameters can be reduced as follows:

$$R(S,A) = \begin{cases} r_{\text{tc}} & \text{if } S = s_x \text{ and } A = \text{submit } s_y, x = y \\ r_{\text{ti}} & \text{if } S = s_x \text{ and } A = \text{submit } s_y, x \neq y \\ r_{\text{cc}} & \text{if } S = s_x \text{ and } A = \text{confirm } s_y, x \neq y \\ r_{\text{ic}} & \text{if } S = s_x \text{ and } A = \text{confirm } s_y, x \neq y \\ r_{\text{terminate}} & \text{if } A = \text{terminate dialog} \\ r_{\text{repeat}} & \text{if } A = \text{greet} \end{cases} \tag{6.40}$$

The focus of this chapter is finding a set of values for the above parameters that yield good dialog performance. Instead of searching from $-\infty$ to $+\infty$ for all six parameters, the nature of the POMDP and the dialog problem constrains the search space. First, since the solved policy seeks to maximize the total reward, scaled values of the parameters yield the same policy, meaning that we can reduce the total search space. Second, as discussed below, the actual meaning of the reward in the dialog provides other constraints:

- r_{tc} is the reward associated with the SDS-POMDP submitting the correct terminal action. The brute-force search constrains this value to be non-negative.

- r_{ti} is the reward for incorrect terminal actions, *e.g.* the SDS-POMDP makes a phone call when the user wanted to know the weather. This parameter is set to be negative, since it is an undesirable outcome.

- r_{repeat} is the reward associated with a generic request for the user to provide information. For all states, this action is assigned a reward that is greater than the value of r_{ti} but less than the value of r_{tc}; otherwise, the SDS-POMDP's policy would automatically prefer to guess terminal actions (for the lower bound) or continue to ask the generic repetition question (for the upper bound). A stricter upper bound is that this parameter is set to be negative to bias against acquiring more positive reward simply by repeatedly asking the same question.

- r_{cc} is the reward received by the SDS-POMDP when the "correct" confirmation is asked; that is, the confirmation question conforms to the underlying state. This reward is set to be non-negative and greater than r_{repeat} because a confirmation question that reflects the user's true intent is desirable dialog behavior.

- r_{ic} is the reward received by the SDS-POMDP for an "incorrect" confirmation question; that is, a confirmation question that does not map to the underlying goal. This parameter is set to be negative and less than r_{repeat}, since confirming the incorrect question could be less desirable dialog behavior than simply asking the user for generic clarification question.

Practically, these constraints will also help to generate policies that will ensure that dialogs will reach the correct final state within a reasonable period of time.

6.5 Brute Force Search and Gradient Descent

A coarse-to-fine approach is taken to search over the parameter space. First, we use brute-force search to find good initialization points. At each iteration, the policy is recomputed (using MATLAB) and 10,000 dialogs were simulated (using the Python-based dialog simulator described in Chapter 5). One representation of the results of this search are shown in Figure 6.5, which illustrates the mean and standard deviations of dialog rewards for particular parameter values and the minimum observed dialog cost at that value. Following the brute-force search, gradient descent was used to find a dialog-cost minimum in the reward-parameter space. This turned out o be a noisy process, likely due to the stochastic nature of the simulation costs, the possible non-convex nature of the cost space, and the relatively small differences in computed costs; different iterations often settled at slightly different points not far from the initial point. To overcome these challenges, the gradient descent was repeated. One parameter setting that yielded good results (a mean cost of below 9.6 seconds) was as follows:

Table 6.2: Reward function parameter values for lowest-cost simulation

Parameter	Description	Value
r_{tc}	correct terminal action	+21
r_{ti}	incorrect terminal action	−71
r_{repeat}	repetition question	−9
r_{cc}	correct confirmation question	−1
r_{ic}	incorrect confirmation question	−23
$r_{\text{terminate}}$	terminate dialog	−15

For visualization purposes, a magnified plot that shows the cross-over points is shown in Figure 6-4, assuming three states instead of 62 states in the actual POMDP. Of particular note is that the region in which the confirmatory question, a_{1c}, represents the optimal action is much smaller than in the toy example shown in Figure 6-1. A closer inspection of the actual simulated dialogs reveals that this has to do with the time-based cost function. Specifically, the cost of a confirmatory question is often higher than simply asking the user to repeat: If an incorrect confirmation question is asked, the user may give a "no" observation, which, while informative, usually does not immediately lead to the correct terminal action being taken. As a result, the dialog often consumes the time associated with another turn. In contrast, a system that rarely, or never, asks a confirmation questions could simply ask the user to repeat, which can lead to the correct terminal action. Indeed, some of the policies that exhibited good performance never involved the confirmation question being asked.

6.6 Limitations

The method used to search over reward parameters yields some policies with better performance than others. However, it is not possible to make any claims of optimality from the QMDP-derived policy. This should not be considered surprising; QMDP does not in any way take into account the observation function. A comparison of the QMDP-derived policies to other methods, including the aforementioned POMDP solver that admits continuous observations [35], would be useful to determine how this non-optimality affects performance. Such methods would also make it possible to relax the parameterization of the reward function. Among other benefits, we could allow every state-action pair in the reward to take on different values instead of fixing the reward function to a small number of parameters needed to make brute-force search and gradient descent tractable.

One possible rationale for using the parameter-search method described in this chapter is if the dialog reward function were non-Markov in some way. For example, it may be that the "reward" associated with a certain system action cannot be sufficiently described by $R(s, a)$. For example, the reward for a confirmation question could become more negative (more costly) as a function of the dialog length, or if the cost changes when the confirmation question has been asked previously in the same dialog. Incidentally, the costs outlined in Table 6.1 are sufficiently described by $R(s, a)$, but it was found that simply setting the reward function to these costs and using the computed policy did not yield better performance. It would be worthwhile to explore the reasons for this observation and whether they relate to the generated approximate policy or other issues.

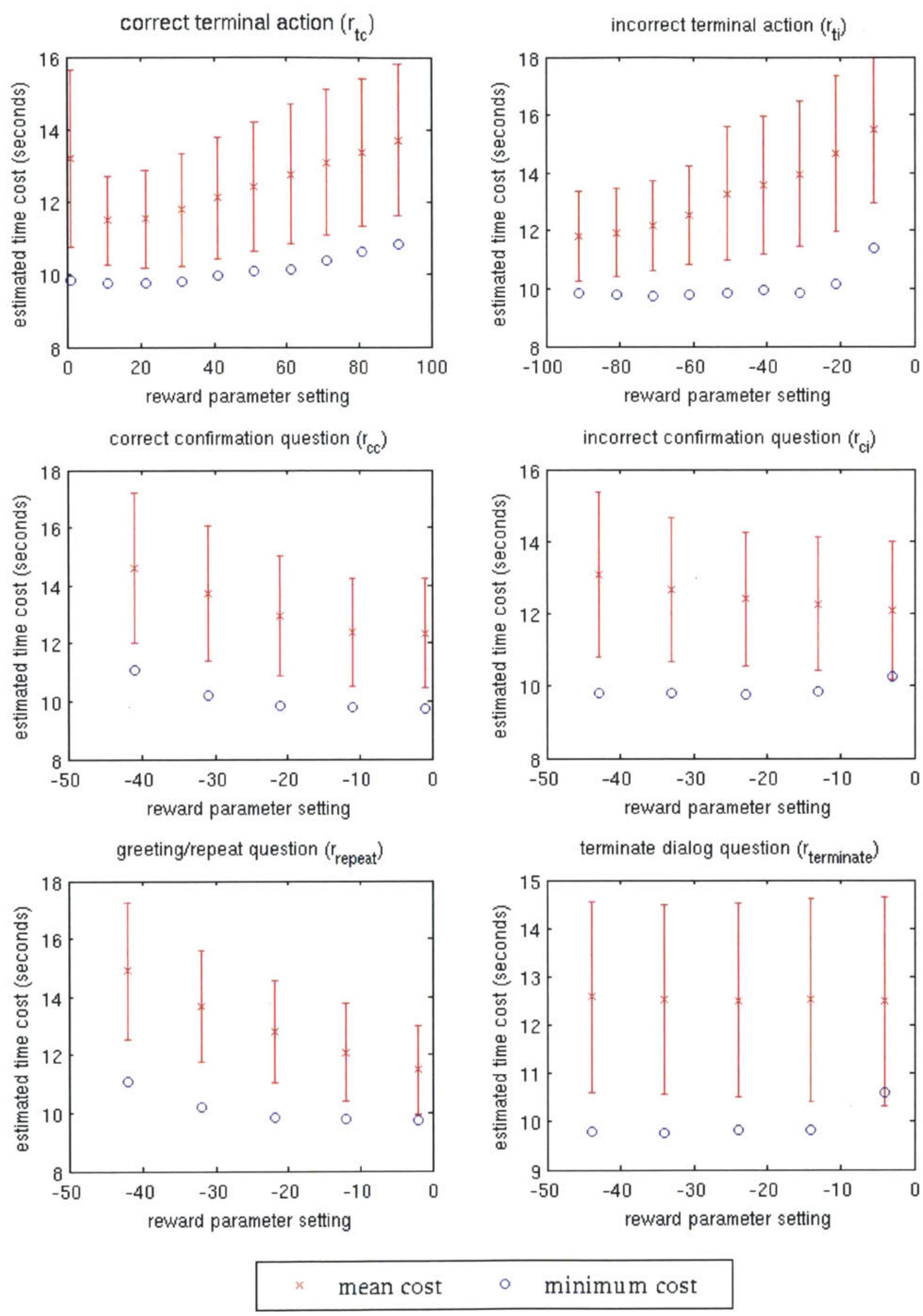

Figure 6-3: Average dialog duration with brute force search. Intervals indicate the variance of the mean value as the other parameters are adjusted.

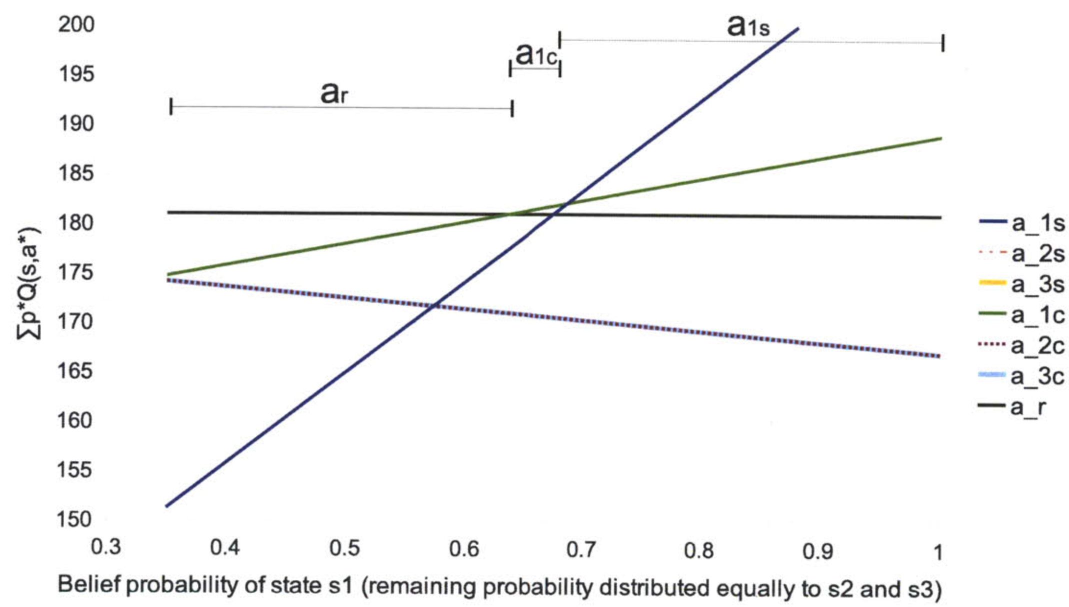

Figure 6-4: QMDP-derived expected payoff of SDS-POMDP actions with best parameters from gradient descent

Chapter 7

User Experiments

The previous chapters describe the design and development of the SDS-POMDP, the user simulator employed to validate its effectiveness in simulation, and the search over the reward parameters to find good simulated behavior. Ultimately, the purpose of these efforts was to create a dialog manager that would lead to more successful dialogs with human participants, especially the target population of residents at The Boston Home. This chapter focuses on the dialog experiments with human participants to validate the simulation results.

7.1 Experiment Design

The human subject experiments compared a threshold-based dialog manager to the SDS-POMDP dialog manager with confidence scoring. Based on a few exploratory tests with users, a confidence-score threshold of 0.7 to trigger the terminal action was found to be a reasonable value for accepting a user utterance. Notably, 0.7 is approximately the value of beyond which the SDS-POMDP "submits" a particular state, given that the remaining probability is distributed equally among the other states.

The actual experiments were similar in design to the simulations in Chapter 5. Specifically, at the start of each dialog, users were given one of the 62 possible goals encoded in the dialog system to pursue and asked to achieve that goal. In total, twenty dialog goals were selected, and each participant participated in forty dialogs: twenty with the threshold-based dialog manager, and twenty using the SDS-POMDP with confidence scoring. The ordering of each of these forty dialogs was randomized, and it was not necessarily the case that the same goal for the threshold and POMDP dialog managers were presented consecutively.

In contrast to the user simulator, the human research participants may be influenced by the structure of the sentence that prompts them with their goal; for example, if the system suggests, "Ask the weather on Wednesday", the user may be more likely to utter, "What is the weather on Wednesday?" as opposed to "What is Wednesday's weather?" To reduce this bias while also evoking the necessary vocabulary for the keyword extractor, the key terms were reordered randomly in a grammatically correct manner.

7.2 Results

Table 7.1 shows the dialog completion rates for lab and TBH participants. As the table shows, all lab participants were able to complete the dialogs successfully with both dialog management strategies. Meanwhile, a few of the TBH residents had difficulty achieving the goals for certain dialogs, and the SDS-POMDP increased the completion rate for most TBH participants.

Figure 7.2 illustrates the performance of the threshold-based and POMDP-based dialog managers for the fourteen users listed in Table 7.1. In the case of unsuccessful dialogs, we assume that the total time elapsed was 60 seconds to compute the values in the figure. Participants "lab03", "tbh01", "tbh06", and "tbh07" had mean dialog durations that were significantly lower (at $p = 0.10$) for the SDS-POMDP than for the threshold-based dialog manager. None of the other differences are significant.

Table 7.1: Number of completed dialogs by lab and TBH users

User	Threshold Dialogs Completed (/20)	POMDP Dialogs Completed (/20)
lab01	20	20
lab02	20	20
lab03	20	20
lab04	20	20
lab05	20	20
lab06	20	20
lab07	20	20
tbh01	18	13
tbh02	17	16
tbh03	20	20
tbh04	19	18
tbh05	13	5
tbh06	18	10
tbh07	17	10

7.3 Discussion

The results demonstrate the benefits of the SDS-POMDP for speakers with high-error rates — the POMDP approach generally appears to perform better than the threshold-based dialog system, and this improvement increases as the mean dialog duration (for either dialog system) increases. This trend underscores the benefit of the POMDP in handling noisy speech recognition inputs: the SDS-POMDP performs just as well as simpler, threshold-based methods for speakers with low concept error rates (*i.e.* laboratory participants), but the SDS-POMDP becomes superior for speakers with high ASR errors.

The benefits of the SDS-POMDP in the human experiments were seen in several different ways. First, correct hypotheses with low confidence scores still contributed to useful belief updates; that is, they increased the probability of the correct observation in the belief distribution. After more turns with the same observation, the SDS-POMDP asked a confirmation question or submitted the correct action. In contrast, the threshold-based dialog system did not consider this dialog history because below-threshold observations were essentially discarded.

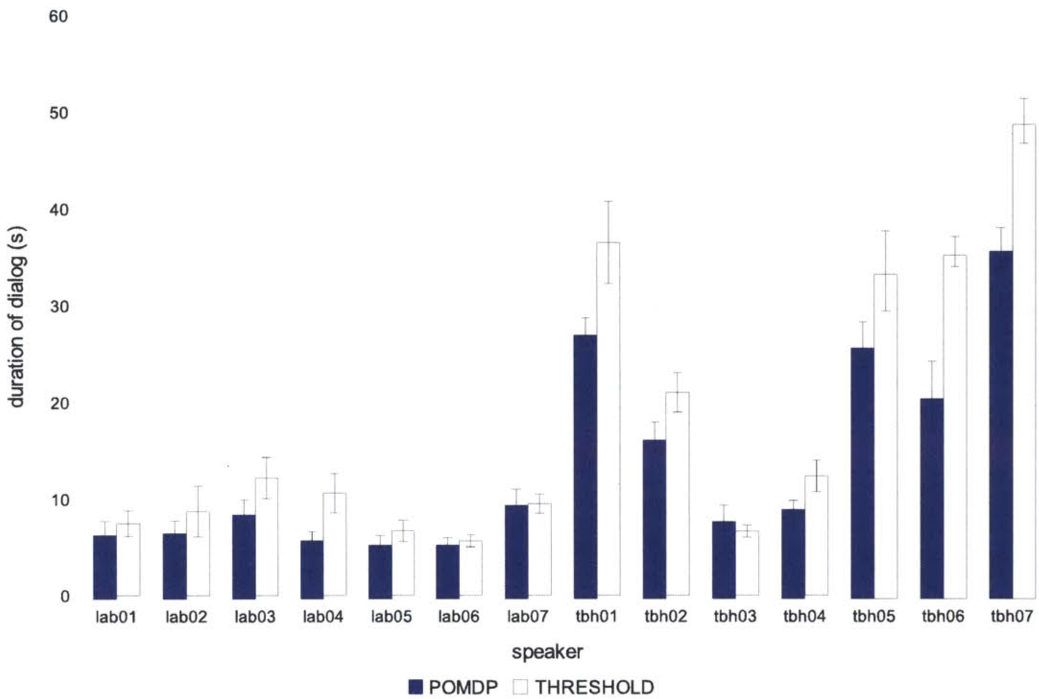

Figure 7-1: Dialog durations for POMDP- and threshold-based dialog systems.

Second, the use of confirmation questions and the consideration of alternative hypotheses by the SDS-POMDP was beneficial in a number of cases. Although the collected data seemed to suggest that the "yes" and "no" observations were somewhat noisy (likely contributing to the small confidence region in which confirmation questions would be asked), most of the participants seemed to be able to produce correct "yes" and "no" utterances without difficulty. The ability to confirm low-confidence utterances did not exist with the threshold-based system.

The results also reveal the limitations of the SDS-POMDP, at least in the current formulation of the model. In at least one case, the speaker misspoke and provided an observation for an incorrect state; providing a high-confidence observation for the correct state in the next turn would result in the immediate successful termination of the dialog for the threshold-based system, but often required an extra turn to overcome the low prior probability for the correct state. Meanwhile, if a speaker was consistently unable to provide an observation that resulted in a higher belief probability for the correct state, then the SDS-POMDP did not succeed, leading to some of the unsuccessful dialogs for TBH participants. These failures demonstrate the problem of off-model observations.

The comparison of the threshold- and POMDP-based dialog systems might have been more objective if the threshold-based system had been able to ask confirmation questions. Instead of having just one confidence-score decision boundary at which the dialog system accepts or rejects a hypothesis, a second, lower threshold could be used to determine when the dialog system asks a confirmation question. Ultimately, however, such handcrafted rules would be challenging to set optimally; in the limit, we could handcraft the policy similar to the one determined by the SDS-POMDP.

Chapter 8

Deploying Assistive Technologies: Ethical and Policy Considerations

The successful deployment and evaluation of any assistive technology must take into account the context in which it is used. Critically, these devices and systems are intended for a vulnerable population: the elderly and people with disabilities. Arguably, therefore, considerations such as the unintended consequences of new technologies are especially important.

This chapter focuses on an experimental location awareness system deployed at The Boston Home that informs staff of the current room-level locations of participating residents. Between June and September 2011, a study on the efficacy of the system to improve staff productivity and resident safety was conducted [23]. In the process of preparing and implementing the study, we encountered two related issues: First, knowing the whereabouts of residents has benefits for their caregivers, families, the specialized-care institution, and the residents themselves, but it may also raise issues related to individual privacy, dignity, and personhood. Second, obtaining the consent of certain individuals who have experienced cognitive decline raises a new set of concerns about personhood and choice.

Section 8.1 discusses the issues associated with wander management technologies. This review is included in this section of the thesis, as opposed in Chapter 2, to help frame the discussion on technology and policy issues. Then, the specific contributions of this chapter are as follows:

1. **Location Awareness Technologies:** In Section 8.1, we discuss the key stakeholders, benefits, and tradeoffs associated with the location awareness system, drawing on our deployment of an actual system at The Boston Home as a case study. While it may not be possible to generalize the experience at TBH to all other residential-care settings, it is worthwhile to compare this case study to discussions in the literature on location awareness systems for similar target populations.

2. **Consent and Planning for Getting Older:** Section 8.3 discusses the issue of obtaining informed consent from residents who may have difficulty understanding the risks and benefits of a research study due to cognitive challenges. Typically, for medical decisions, an activated health care proxy can make decisions on a person's behalf; however, it is unclear whether using assistive technologies is a decision that such proxies are currently authorized to make. This section discusses how individuals, caregivers, and society currently plan for decisions as people age and face new challenges, particularly given the existence of new technologies, such as location monitoring, that may not be anticipated.

The content of this chapter centers on the ethical questions related to knowing a resident's location, but it might be generalizable to other knowledge about the individual's state and preferences. Due to demographic shifts and advances in computing, interest in technology that monitors a resident's location, health status, or other needs is increasing [18]. More directly, the issues raised in this chapter may also be applicable to a future deployment of an assistive dialog interface, such as the one described in the rest of this thesis.

8.1 Ethics of Location Monitoring Technologies

One effect of society's aging population is that there are growing numbers of people with Alzheimer's disease, dementia, and other forms of cognitive decline [33]. Such conditions can cause people to become lost or disoriented, leading to the phenomenon of "wandering" in which individuals may put themselves in danger by getting lost and finding themselves in unfamiliar surroundings [43, 49]. Residential-care facilities, including The Boston Home, are ethically and legally responsible for the safety of their residents. This concern has led to the development of wide range of wander management technologies that allow staff to know the whereabouts of residents in these settings, some of which are commercially available.

An ethical issue can be framed in terms of a tradeoff between two or more competing values. These values may include a desire to do good or prevent harm. In the case of wander management technologies, it seems clear that the desired "good" is is to increase resident safety and well-being by knowing their location. A possible tradeoff is that such technologies can come at the expense of a resident's autonomy or independence, but this notion requires a more nuanced analysis In many specialized-care settings, however, one could argue that such systems may increase an individual's autonomy: A wander management system could ensure that doors may not need to be locked, that a caregiver may not need to be constantly watching the person, or that a person can continue to use a power wheelchair. Therefore, while a wander management system could have some adverse impact on a person's freedoms and autonomy, these considerations may need to be balanced with the positive impacts on quality of life, particularly compared to alternative approaches [39].

The potential harms of a wander management system center on how a participant in the system is perceived and treated. First, a wander management systems could affect personhood of people who are "tagged" or "tracked" with an electronic device [38]. Specifically, the idea of "tagging" people potentially implies that individuals might be seen more as objects as opposed to human beings. Second, residents who are part of a wander management system must have some kind of marker or tag on themselves or on their mobility device, which could cause them to be stigmatized. Third, as new technologies that have the potential to monitor people's activities in greater spatial and temporal detail are developed, the scale and scope of threats to people's privacy also increase. These threats to the freedoms and civil liberties of individuals must be balanced against the potential benefits of such systems.

The above discussion lists hypothetical risks; the actual *implementation* of a location awareness system by technologists in a real-world setting seems central to determining whether it is appropriate for the technology to be used. This is the focus of the next section, which discusses our study at The Boston Home.

8.2 Resident Location Awareness at The Boston Home (TBH)

The responsibilities of TBH staff include knowing the general whereabouts of residents, particularly individuals experiencing cognitive decline who may be at risk of "wandering" off the grounds of the facility. In addition, knowing the location of residents is useful to reduce the time required for staff to find them for appointments, visitors, or other activities. Anecdotally, this issue is one of growing concern at TBH; interestingly, one contributing factor is that, with new innovations in alternative drive controls for power wheelchairs, many residents continue to remain mobile even as they become older.

Both of these issues were identified by staff and management at The Boston Home, leading to the development of the "MIT People Finder" software system. Over the course of two years, the Organic Indoor Localization (OIL) system, a WiFi-based indoor localization system, was deployed and tested at The Boston Home, with twenty highly mobile residents having N810 tablets attached to their wheelchairs [63]. Figure 8-1 shows a Nokia N810 tablet (referred to in this chapter as the "client") mounted on a resident's wheelchair and the location-finding software interface for staff at The Boston Home. Briefly, the system uses the signal strengths of wireless access points, as measured by the client, as features to determine room-level estimates [55, 63]. The room-grain location of the client is estimated on the client, transmitted to a server, and finally displayed on a software interface at designated staff computer terminals. Based on the configuration of the system, the location information displayed on the software interface typically showed the real-time location of the resident [23].

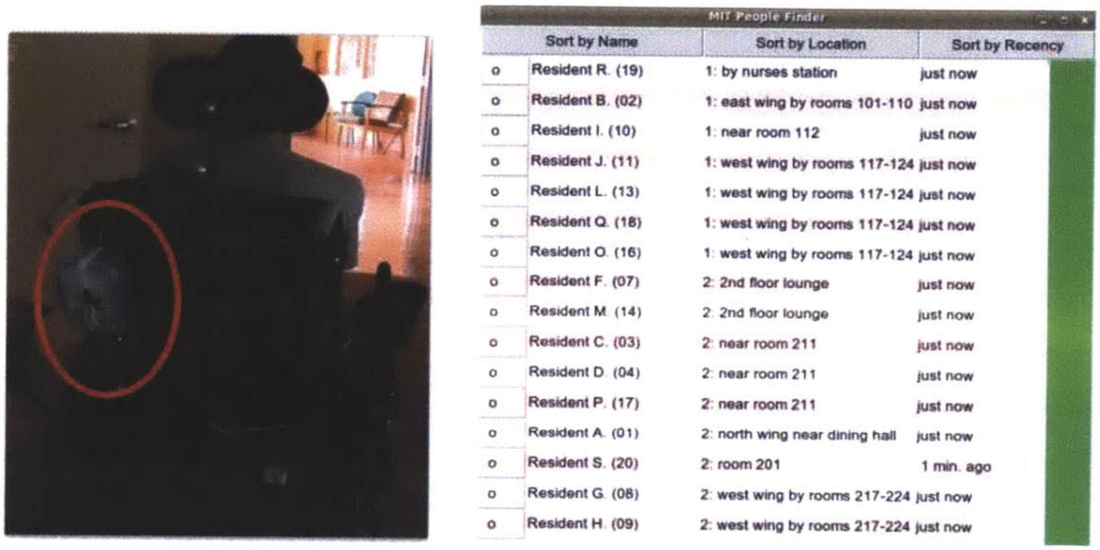

Figure 8-1: *Left:* PDA tablet mounted on resident wheelchair at The Boston Home. *Right:* Software interface used for resident-finding study (names redacted).

The pilot study of the system consisted of two parts. In Study One, nineteen highly mobile residents and nine staff (clinicians from physical therapy and rehabilitation, the adaptive technology coordinator, and the unit secretaries on each floor) participated in a study that examined the effect of the system on the time required to find residents. Specifically, the participating staff members were given clipboards and stopwatches and asked to record the time required to successfully search for a participating resident when he or she was needed. The study consisted of a four-week control period without the location awareness system, followed by a four-week period with the "MIT People Finder" software interface. Next, in Study Two, nine residents were selected because they were highly mobile and were identified by staff as being at risk of wandering without informing staff. Over a two-month period, a modified version of the software interface, which displayed pop-up alerts when one of the participating residents had gone outside or left the TBH premises, was deployed [23].

8.2.1 Study Findings on System Efficacy

The results of the study suggest that the goals of reducing finding times and improving resident safety were achieved. In Study One, we found a significant (p=0.0008) decrease in the mean time required to find residents with the system, from 311.1 seconds (24 find events) to 110.9 seconds (18 find events). In Study Two, meanwhile, over the course of two months, the software interface recorded 194 correct alerts that residents participating in the study had gone outside or left the grounds of TBH (along with 21 false positives, when the system incorrectly suggested that a participant had gone outside). Through semi-structured interviews, staff responded positively to the system, describing many examples when the software was useful. Full details can be found in [23].

8.2.2 Design Decisions to Mitigate Ethical Risks

Evidently, the system was useful in terms of improving resident safety and staff productivity at The Boston Home. In addition to contributing to the technical efficacy of the system, the design choices and engineering efforts that went into the resident-finding system also contributed to mitigating the risks associated with wander management system. Using the issues hypothesized by Hughes et al. [39] as a starting point, each of the possible risks are addressed in turn, based on our observations of resident and staff attitudes toward the system during the study and semi-structured interviews with staff upon completion of the both phases:

Personhood and Dignity: A possible risk with the location awareness system might have been that participants could be seen as "objects" as opposed to people. However, our semi-structured interviews and informal conversations with residents and staff did not give us concerns related to this issue. Importantly, the number of staff authorized to view the interface was limited; only the clinical staff, adaptive technology coordinator, and unit secretaries on the two resident floors could view the interface on their computer screens and were trained to use and communicate this information judiciously. During the interviews, some staff participants acknowledged that it was interesting to see how the system performed as participants moved throughout the building, but the comments generally centered on whether the system was useful for finding residents.

Privacy: In the context of the study, the potential intrusion into the privacy of the participants was guarded in two main ways. As mentioned above, only a few staff members were authorized to view the interface, and the computer terminals showing the information were not positioned in areas that could be easily seen by casual bystanders. The computer screens of the unit secretaries, who sit at a desk in a busy area of The Boston Home, were potentially viewable by other nurses and staff; however, in general, restricting the number of people with access to the software interface discouraged behaviors such as "watching" the interface to see how people move over time. Second, the software interface, explicitly by design, did not have a "playback" or "log" mode for the end user; only the instantaneous location information could be seen by the user.

It is worthwhile to note that our system did not have a distinct "temporary opt out" feature on each client that could allow the participating residents to become invisible on the interface for a period of time. If a participant wished to become invisible, the possible ways might be to inform a staff member or to push the "opt out" button of the study for a period of time; arguably, however, taking these actions would be more difficult to do for a resident. From a practical standpoint, one concern with such a feature was that the participant would forget to re-activate his or her client, thereby defeating the purpose of the system. In the end, none of the participants opted out of the study, but a "temporary opt-out" option is worth at least some consideration in future systems.

As part of examining the effect of the system on privacy, understanding the nature of privacy in a 24-hour skilled nursing facility like The Boston Home is worthwhile. Almost by definition, the care requirements of residents in a skilled-nursing facility, in contrast to other eldercare options such as assisted living or living at home, are greater [71]; at The Boston Home, residents often need assistance with sensitive activities ranging from taking medications to having meals and transferring in and out of bed.

Inevitably, knowing potentially sensitive information about residents is a by-product of these care requirements. In terms of location information specifically, staff members generally know about the whereabouts of residents at all times; residents are advised to "sign out" with the staff at the front desk if they are leaving the premises, and staff are particularly vigilant about residents who may have a tendency to wander. Overall, these observations suggest that the location awareness system, in providing location information about a certain set of residents, did not substantially change the culture or definition of privacy at The Boston Home.

Stigma: As Figure 8-1 shows, the Nokia N810 clients were clearly visible on the wheelchairs of participating residents. This placement was chosen mostly for practical reasons: the client is fairly large (the size of a large cellular phone), the back of the wheelchair provided the best place to access power and mount the plastic container, and the position made it fairly straightforward to fix problems quickly. Arguably, the visible placement of the clients helped to ensure that residents themselves remained aware that they were part of the research study; since the system requires no active input on their part, it could be easy to forget about the tablet. Given that the clients have been on the wheelchairs of participating residents for over two years (as part of a longer-term characterization of the location awareness system) and that no resident has ever opted out of the system, it may be reasonable to state that the participating residents believe that the placement of the clients is not a concern or a source of stigma.[1] Overall, while it might be desirable for future systems to have more discreet clients, the setup for our experimental system did not seem to cause major concerns for participating residents.

In addition to the thoughts and attitudes of participating residents themselves, the views of other caregivers, staff, and family members are critical for assessing the possible stigma attached with participating in the system. In particular, the second part of our study focused on nine residents whom staff had identified as being at risk

[1]Informal conversations with residents about their attitudes toward the hardware on their wheelchairs range from being indifferent ("It doesn't affect me or my wheelchair in any way") to being positive ("I'm happy to help") or whimsical ("We call you the 'Massachusetts Institute of Tupperware'").

of wandering, and for whom having such a visible electronic tag could be stigmatizing. However, an interesting artifact of our study was that participants in the first phase retained their tablets during the duration of the second phase, even though their location information was not being displayed on the interface. This fact is noteworthy because these residents were specifically chosen because they were highly mobile, had an interest in engaging in a research project, and were generally independent. As a result, residents identified by staff in the second phase of the study were not the only individuals with clients on their wheelchairs. One insight that can be drawn from this experience is that the reducing stigma might be achievable by thinking about how the system is perceived publicly (the hardware visible on wheelchairs) versus how it is actually used (through software only accessible to a few staff members).

Independence and autonomy: Similar to the above discussion on resident privacy, our experiences with the location awareness system did not suggest that it had an adverse impact on the independence and autonomy of participating residents. In both phases of the study, residents were never prevented from going outside or leaving the premises of TBH; the system simply displayed this information and left it up to designated staff member to decide what to do. The system was also reactive; it did not make predictions on when an individual might be going outside or to a specific location. In the absence of the location awareness system, caregivers were still responsible for being aware of the whereabouts of residents and ensuring their safety.

Consent: Perhaps most importantly, all of the residents who participated in the study at The Boston Home provided their informed consent to participate in a research study that received approval from the MIT Committee on the Use of Humans as Experimental Subjects (COUHES); we chose, for this pilot study, not to involve residents who might have required a proxy. As part of the study, they had the right to opt out at any time by pressing a button on the client or by informing a member of the TBH staff or the research team. By agreeing to participate in the research study, they asserted that they were aware of the potential risks and benefits of the location awareness system. It is worth noting that some residents who were approached in the initial round of recruitment expressed disinterest or declined to participate in the study. In addition, of the twenty residents enrolled in the original system research study, one declined to participate in the resident-finding study. In itself, this is an interesting finding, since many commercial wander management systems or strategies are deployed without the explicit consent of residents who are part of the system.

8.2.3 Stakeholder Analysis on Ethical Issues

A useful approach to summarize the tradeoffs might be to understand the relative importance of different competing values to the various stakeholders in a skilled nursing facility. Figure 8-2 speculates on the relative importance on which these stakeholders (residents themselves, management, nursing staff, and family members) might place on the potentially conflicting values of resident safety and personhood. The positions on this axis are based on analyzing the interests and responsibilities of each group, discussions at The Boston Home, including two presentations to the TBH Ethics Committee in 2011, and a review of the literature (specifically, limited survey results (n=143) of these different groups on this topic [39]. To be clear, a strong emphasis on one value does not imply a a complete neglect of the other. Moreover, the relative positions of different groups, not the exact values, are the most salient feature of this plot.

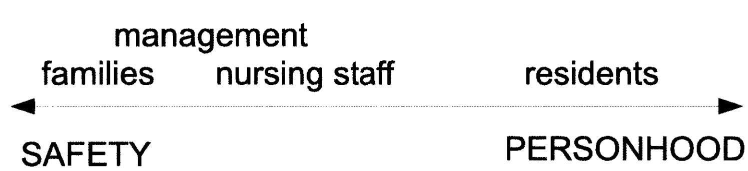

Figure 8-2: Hypothesized relative weights on the importance of safety and personhood for different stakeholders. This diagram does not imply that these stakeholders are in conflict about the decision on whether a location awareness system should be used; rather, it may be a guide for understanding the responsibilities and priorities of different stakeholders.

An explanation of the viewpoints of different groups is discussed below:

Residents: In skilled nursing facilities, residents themselves have an interest in their own well-being and safety, but also their dignity and privacy. A location awareness system has the potential to provide participating residents with an added sense of safety, but it may come at some expense to their independence and personhood. Interestingly, the evidence on which value is more important is conflicted: among a small set of survey respondents with mild memory problems, Hughes et al. [39] found strong acceptance of electronic tagging for safety reasons. However, at The Boston Home, it is perhaps notable that the attitudes toward participating in the system ranged from residents who were enthusiastic to participate to those who declined to be involved in the study. Overall, it seems reasonable to state that, among all of the stakeholder groups, residents have the strongest interest in their own sense of personhood and dignity.

Management: While management in a skilled nursing facility has an overall interest in the well-being of residents, which encompasses both safety and personhood, its legal responsibility for the safety of residents may be a dominant consideration. A system that provides staff with tools to save time and manage wandering behavior might also produce cost savings. Granted, focusing only on the legal and financial considerations is a simplified view of administrators and risk managers in a residential care setting — an important nuance, for example, is that the location awareness system may actually *preserve* a resident's mobility (and, consequently, their independence and personhood) by allowing him or her to continue to use a power wheelchair. However, these other pressures suggest that safety is paramount.

Family Members: Conversations at The Boston Home and discussions in the literature suggest that family members may actually be among the most concerned about the safety of their loved ones; in the survey results, family caregivers (n=32) placed much stronger emphasis on safety than on civil liberties [39]. Much of the interest on location monitoring in the literature can also be attributed to family members of elderly people living independently at home [38]. For these reasons, Figure 8-2 suggests that family members may be among the most concerned about resident safety, possibly at the expense of other values.

Clinical Staff: Nurses and nursing assistants potentially have a strong interest in knowing the whereabouts of residents; their job responsibilities include being aware of residents' needs and a whereabouts, which would make a location awareness system. Interestingly, although all stakeholder groups were strongly supportive of electronic tagging (compared to other alternatives) in the survey by Hughes et al. [39], the biggest source of concern about the dignity of individual's came from clinical staff, which consisted of geriatricians, old age psychiatrists, nurses, and occupational therapists. While this tendency was not explored in detail, one hypothesis could be that front-line staff may be more cognizant of the importance of person-centered care than other groups [39].

These qualitative observations suggest that most stakeholders have a primary interest in ensuring the safety of residents who participate in a location awareness system. Meanwhile, although the personhood of residents is an important consideration for all stakeholders, it is arguably a superior priority only for the resident stakeholder group. This observation may make the next section, which discusses the issue of obtaining consent from individuals who may have cognitive challenges.

8.3 Informed Consent for the Elderly and People with Disabilities

The previous section outlined some of the tradeoffs in terms of resident safety and independence, both of which are important values in specialized-care settings. Overall, it seems that the benefits of the system for improving resident safety and staff productivity were clear; however, lingering questions remain about the risks that the system presents to the personhood of people who are part of the location awareness system. In particular, the analysis of the views and considerations of different stakeholders suggests that residents themselves are the group most concerned about the value of dignity and personhood, while other stakeholders may be more concerned about their safety.

In the study that we conducted at The Boston Home, the balance between these competing values was perhaps achieved because all of the residents who were part of the system gave their written consent to participate. The protocol was approved by the MIT Committee on the Use of Humans as Experimental Subjects, which is required by federal law to approve all research involving humans at MIT [3]; in addition, the research protocol was reviewed by the TBH Ethics Committee, a body consisting of community members, staff, and management at The Boston Home who review important ethical issues.

A significant ethical and policy question, however, arises when individuals who might benefit from a location awareness system cannot fully appreciate the nuanced tradeoffs between competing values. As one indicator, at The Boston Home, some residents have an activated health care proxy, meaning that that they have given authority to a person whom they trust to make important health care decisions on their behalf [13]. Understanding the nature of informed consent and the roles of an individual's legal surrogates, if they exist, could answer the specific question of how to obtain consent for the location awareness system study and other similar research projects. More broadly, this discussion could help inform how people might plan and make decisions with competing value tradeoffs as they age and potentially face disability or cognitive decline.

8.3.1 Consent Capacity and Legally Authorized Representatives

Informed consent is a pillar of human subjects research in the United States; according to the Code of Federal Regulations (CFR), research can only proceed after "legally effective informed consent of the subject or the subject's legally authorized representative" has been obtained [1]. This regulation can be parsed into two parts that merit further discussion: the definition of "legally effective informed consent" and the laws that govern who may serve as the "legally authorized representative."

Some of the discussion in this section is drawn in a 2009 policy directive from the National Institute of Health on research with vulnerable populations with people who may have questionable capacity to give informed consent [11].

To aid the reader, it may be helpful to give an overview of the possible legal sources from which we can define the proper procedure for obtaining consent. In this case, the rules and policy directives promulgated by the U.S. Department of Health and Human Services (HHS) on the protection of human subjects in research is most relevant. HHS has been given rulemaking authority in this domain by Congress. In addition to these administrative laws, other statutes or regulations at the state level may also be relevant. In particular, as discussed below, how the Commonwealth of Massachusetts defines health care proxies and durable power of attorney agreements is important.

Consent Capacity: The ability of an individual to give consent for research studies is not determined by whether he or she has an activated health care proxy or other decision-making surrogate; rather, it is defined by whether he or she can comprehend the "information required to make an informed, voluntary decision to participate in the research study" [11]. Dementia or other forms of cognitive decline can compromise the consent capacity of individuals [24]. A number of studies have attempted to develop assessment tools to determine whether a person has sufficient capacity to provide consent [17]; to date, though, no agreed-upon standard has emerged, and the varying complexity of different study interventions, along with the complex nature of different ethical tradeoffs make it difficult to develop a truly universal standard [11]. At The Boston Home, it was clear that some potential research participants in the location awareness system studies did not fully comprehend the nuanced benefits and risks associated with participation, thereby ultimately necessitating a the role of a representative if we had decided to include these individuals in the initial study.

Selecting a Representative for Obtaining Consent: If it is determined that an individual does not have the capacity to provide informed consent, the Code of Federal Regulations (CFR) provides for "legally authorized representative (LAR)" to fulfill this role. This LAR is defined as "an individual or judicial or other body authorized under applicable law to consent on behalf of a prospective subject to the subject's participation in the procedure(s) involved in the research" [2]. The key difficulty, however, is that no law seems to clearly give authority to an individual for non-medical research studies; Children's Hospital Boston, for instance, states in its Clinical Investigation Policy and Procedure Manual that "Massachusetts state law has no specific definition of legally authorized representative just for research decision making" [5]. In the absence of clear guidelines, the suitability of three possible LARs are described below: health care proxies, durable power of attorney agents, and family members.

Health Care Proxy: In its policy directive, the NIH states that a LAR might be someone who has been granted a "durable power of attorney for health care". Under Massachusetts law, this person is called the health care proxy, and authority to make health care decisions on is granted to him or her if a physician determines that the patient lacks capacity [9].

Notably, the wording of the Massachusetts health care proxy law states that the agent granted the health care proxy has the "authority to make health care decisions on the principal's behalf". It further defines "health care" as "any treatment, service or procedure to diagnose or treat the physical or mental condition of a patient" [9]. Combined with the NIH policy directive, it could be fairly concluded that the health care proxy should have the authority to make decisions on research participation if the study has a medical nature; however, it is not entirely clear that studies of assistive technology fit this criteria. For instance, the location awareness system seeks to improve resident safety, but it seems debatable whether this can defined as a "health care" intervention.

Durable Power of Attorney: Massachusetts state law, under the Massachusetts Uniform Probate Code, also makes provision for an individual (the "principal") to grant durable power of attorney to an "agent". Critically, the term "durable" refers to the power of attorney remaining valid even after the principal has become incapacitated [8]. Support for the notion that durable power of attorney grants decision-making authority for research studies, however, appears limited unless the principal has clearly included authority to participate in research. As one example, the Clinical Investigation Policy and Procedure Manual at Children's Hospital Boston states that a appropriate surrogate decision-maker may be a "person with power of attorney who has clear authority to consent to participation in research, or clear authority to make health care decisions for a class of diagnostic and therapeutic decisions that are inclusive of the proposed research ..." [5].

Family members: Given that it is unclear whether the health care proxy or the agent with durable power of attorney can make consent decisions on non-medical research studies, some support exists for a trusted family member to be designated as the legally authorized representative, potentially on an ad-hoc basis. Three examples of policies among Boston-area institutions include:

- MIT COUHES suggests that the subjects with "limited comprehension" should have an "authorized third party ...to represent the subject's interests"; the main requirement is that the third part should "understand the incompetent subject's situation and act in that person's best interest." [4].

- The Dana-Farber Cancer Institute's Office for Human Research Studies states that, in order of priority, a spouse, adult child, parent, sibling, other relative, or close friend could serve as a legally authorized representative for research studies, and that designating such representatives has precedence in Massachusetts case law on medical decision-making [6].

- Children's Hospital Boston states that the adult spouses, adult children, or parents may be authorized to make decisions on research studies. Similarly, its justification for such authority lies in the notion that research has essential similarities to clinical procedures, stating that "it is generally legitimate to consider a surrogate who is involved in the patient/subjects care and is knowledgeable about him or her, and his or her preferences and needs, as having the ability to consent to research participation in which the subject will receive a direct health benefit relevant to the care he or she receives at Children's Hospital." [5].

This survey of federal and state laws and regulations, along with policies of organizations that conduct research with people with limited consent capacity, provides some modest guidance for how to conduct non-clinical research studies in a setting like The Boston Home. First, whether a person has an activated health care proxy or other surrogate has no direct bearing on whether he or she can properly give informed consent; instead, the relevant criteria is whether he or she can understand the risks and benefits of a research study and make an informed choice. Second, if the person cannot make an informed decision, unless he or she has granted durable power of attorney with specific authority on research studies to an agent, there is no clear answer on whether a health care proxy, or a trusted family member is the most correct choicc for informed consent. Importantly, though, as illustrated by the the policy directives of the NIH and three Boston-area research institutes, in spite of the lack of perfectly clear guidelines, it is still possible to ethically and legally include people with limited consent capacity.

8.4 Conclusions and Future Research Directions

Moving forward at The Boston Home, it seems worthwhile to define TBH-specific policies for involving residents with limited capacity to consent in research studies. Given the guidelines set out by MIT COUHES (having an "authorized third party" to represent the potential research subject's interests [4]), future MIT-TBH research might involve identifying the most suitable third party for each potential research participant to discuss the study and obtain informed consent. The process might involve researchers, social workers and clinicians at The Boston Home, and oversight by the TBH Ethics Committee.

8.4.1 Planning for Future Living and Care

An important undercurrent of this chapter is how individuals make decisions about their future living and care as they get older, especially if they face the prospect of disabilities that affect their reasoning capacity. This theme unites the discussion about the ethics of location awareness systems and the legal frameworks that govern informed consent. Although decisions related to aging and disability extend beyond questions that are of a strictly medical nature, the relevant ethical and legal guidelines are ill-defined at best. With the growing number of technologies that could benefit the elderly and people with disabilities, these difficult decisions are likely become more common. Moreover, individuals and their families will not be able to predict all technological innovations in advance, further increasing the complexity and uncertainty of decision-making.

Overall, the discussion in this chapter reveals a number of possible further avenues of research related to planning and decision-making for the elderly and people with disabilities. For instance, the usefulness of advanced planning and directives from individuals before they become cognitively impaired, especially given that technological change can be difficult to predict, is worth exploring. Meanwhile, lessons might also be drawn from how the elderly and people with disabilities make other kinds of difficult decisions, including whether to live independently or enter a residential-care setting, continue to drive a car, or make other life-changing choices. Such work could help inform what policies and practices could best help the elderly and people with disabilities make make good choices about the use of technology and other life domains.

Chapter 9

Conclusions and Future Work

This thesis describes the development of speech-based assistive technologies for people with disabilities, with a focus on the design and evaluation of a POMDP-based spoken dialog system. A corpus of utterances was collected from research participants from the laboratory and from The Boston Home to train speech-recognition and dialog-system models, a SDS-POMDP was formulated and learned from this data, and experimental results demonstrated its benefits over simpler, threshold-based models. A few possible refinements and extensions arise from the experiments discussed in previous chapters.

9.1 SDS-POMDP Policy Optimization

Although good dialog experiment results with actual huamn users were seen with the QMDP-derived policy for the SDS-POMDP, further work is needed to move toward finding optimal dialog policies, as outlined in Chapter 6. First, an analysis of policies derived using QMDP and point-based backup methods would be informative and useful to determine how far from optimality the QMDP-derived solution actually is.

The simulation-based method used to generate dialogs could be useful if the reward function cannot be sufficiently expressed by $R(s, a)$. In addition to the examples suggested in Chapter 6, there are other features of good dialogs beyond its total duration, then finding the best reward parameter settings using the user simulator.

An interesting idea worth exploring might be to learn features of "good" dialogs, possibly from speech- or text-based ratings collected from actual dialog participants. Many of these features could be non-Markov, meaning that the reward function would not be expressible simply as $R(s, a)$.

9.2 Extensions

Validation of the SDS-POMDP in extended, real-world experiments is an important area of future work. The experiments described in Chapter 7 prompted the user with goals, making it straightforward to compare the POMDP- and threshold-based dialog managers. Ultimately, though, the usefulness of this approach for assistive technology applications depends on whether users derive benefit from the system over extended periods of time. From a modeling standpoint, such experiments could reveal how the SDS-POMDP performs as a user's speech recognition accuracy changes (over multiple time scales) and how it is used in real-world settings.

Future work could also focus on each of the components of the SDS-POMDP framework outlined in Chapter 3. In the area of SDS-POMDPs, there are many possible directions related to learning better models, solving large POMDPs efficiently, or adapting the model; some possible directions that seem useful for the assistive-technology domain are discussed here. First, even though there are related goals in the state space (either by the type of information or by the date), a simple state space that treats every distinct goal as a separate, unrelated state was implemented in the SDS-POMDP is described in this thesis. A factored, slot-based, or hierarchical state space might be more scalable; such models would also change the nature of the observation and transition functions. An additional compelling challenge is how the state space and the associated observation functions might be defined for other kinds of concepts; for instance, if the goal of the dialog system were to understand a desired path that the user wanted to take in a robotic wheelchair, there could be hundreds or thousands of possible paths, and it is unclear whether all of these possibilities would need to be pre-defined in the state space.

Second, the observation function, which describes the distribution of observations for every state-action pair, is also ripe for refinement. Most notably, a simple keyword extractor is used to map speech-recognition hypotheses to concepts, but this clearly constrains the user to commands that use the pre-defined keywords. An exploratory dataset of sentences collected for many of the categories on Amazon Mechanical Turk reveals many reasonable ways to request the different functions in the system. For example, some of the ways to ask for the weather, without using the word "weather" or "forecast", include "Is it sunny?", "Do I need to wear a coat when I go out?", "What's the temperature like?", "Is it nice out there?", and "Is it going to be windy outside?" Understanding these different ways of asking about the same concept likely requires more sophisticated parsing and language-understanding tools.

In addition to this language-understanding challenge, there could be better ways of assigning confidence scores to different hypotheses. Currently, the system simply assigns a confidence score to the top hypothesis, using the high- and low-confidence models to update the belief probabilities of the other states. As a result, the only mechanism by which the belief probability of other states changes is through the function $P(Z_d|S, A)$, which itself is trained on counts of the top hypothesis for every different state. Alternative models could potentially assign different confidence scores to lower-ranked hypotheses, making each observation more informative.

Lastly, more sophisticated clarification questions (*e.g.* disambiguating among closely related concepts) could be a useful addition to the system. Currently, the only kind of clarification question within the action space of the SDS-POMDP is confirming individual concepts; it could be useful, for instance, to ask "Did you mean X or Y?" if there are two competing hypotheses or to ask about a particular part of the concept if related concepts are similarly probable. A related challenge, meanwhile, is that actual dialog data would be most ideal to train the appropriate observation functions.

The broader theme of this work is intelligent assistive technology that leverages noisy sensor inputs and knowledge about the environment to provide appropriate assistance to human users. In this work, we have focused on speech input and handled the uncertainty of speech recognition hypotheses; a more sophisticated system, however, could use additional sensors in addition to speech to infer the preferences and goals of users. For instance, one might infer the user's needs based on his or her location, the time of day, the presence of other people in the environment, or other kinds of multimodal user interactions. Ultimately, being able to reason over these different sources of knowledge could lead to assistive technologies that better serve their intended users.

Bibliography

[1] Code of Federal Regulations Title 45 Part 46 Section 111, Protection of Human Subjects — Criteria for IRB approval of research. http://www.hhs.gov/ohrp/humansubjects/guidance/45cfr46.html.

[2] Code of Federal Regulations Title 45 Part 46 Section 102, Protection of Human Subjects — Definitions. http://www.hhs.gov/ohrp/humansubjects/guidance/45cfr46.html.

[3] MIT Committee on the Use of Humans as Experimental Subjects. http://web.mit.edu/committees/couhes/.

[4] MIT Committee on the Use of Humans as Experimental Subjects: Subjects with Limited Comprehension. http://web.mit.edu/committees/couhes/population_limcomp.shtml.

[5] Children's Hospital Boston Clinical Investigation Policy and Procedure Manual: Informed Consent/Permission/Assent Process. http://www.childrenshospital.org/cfapps/research/data_admin/Site2206/mainpageS2206P3.html.

[6] Dana-Farber Cancer Institute Office for Human Research Studies: Legally Authorized Representatives. http://www.dfhcc.harvard.edu/fileadmin/DFHCC_Admin/Clinical_Trials/OPRS/Investigator_Resources/Information_Sheets/IS_-_Legally_Authorized_Representatives.pdf.

[7] Nuance Dragon NaturallySpeaking. http://www.nuance.com/dragon/index.htm.

[8] Massachusetts Session Laws, Chapter 521 Section 5-501, Durable Power of Attorney - Definition. http://www.malegislature.gov/Laws/SessionLaws/Acts/2008/Chapter521.

[9] Massachusetts General Laws, Chapter 201D Section 6, Health Care Proxies — Incapacity of patient; regained capacity. http://www.malegislature.gov/Laws/GeneralLaws/PartII/TitleII/Chapter201d/Section6.

[10] Quartet Technology, Inc. - Products. http://www.qtiusa.com/products.aspx.

[11] National Institutes of Health: Research Involving Individuals with Questionable Capacity to Consent: Points to Consider. `http://grants.nih.gov/grants/policy/questionablecapacity.htm`.

[12] The Boston Home. `http://www.thebostonhome.org`.

[13] G.J. Annas. The Health Care Proxy and the Living Will. *New England Journal of Medicine*, 324(17):1210–1213, 1991.

[14] A.J. Astell. Technology and personhood in dementia care. *Quality and Ageing - Policy, Practice, and Research*, 7(1):15–25, 2006.

[15] A. Atrash, R. Kaplow, J. Villemure, R. West, H. Yamani, and J. Pineau. Development and Validation of a Robust Speech Interface for Improved Human-Robot Interaction. *International Journal of Social Robotics*, 1:345–356, 2009.

[16] R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, USA, 1 edition, 1957.

[17] V.D. Buckles, K.K. Powlishta, J.L. Palmer, M. Coats, T. Hosto, A. Buckley, and J.C. Morris. Understanding of informed consent by demented individuals. *Neurology*, 61(12):1662–1666, 2003.

[18] J.F. Coughlin and J. Lau. Cathedral Builders Wanted: Constructing a New Vision of Technology for Old Age. *Public Policy and Aging*, 1(1):4–8, 2003.

[19] J.F. Coughlin, L.A. D'Ambrosio, B. Reimer, and M.R. Pratt. Older Adult Perceptions of Smart Home Technologies: Implications for Research, Policy and Market Innovations in Healthcare. In *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1810 –1815, August 2007.

[20] K.M. Daniel, C.L. Cason, and S. Ferrell. Emerging Technologies to Enhance the Safety of Older People in Their Homes. *Geriatric Nursing*, 30(6):384 – 389, 2009.

[21] K.M. Daniel, C.L. Cason, and S. Ferrell. Assistive technologies for use in the home to prolong independence. In *Proceedings of the 2nd International Conference on Pervasive Technologies Related to Assistive Environments*, PETRA '09, pages 26:1–26:4, New York, NY, USA, 2009. ACM.

[22] F. Doshi. Efficient model learning for dialog management. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 65–72. AAAI Press, 2007.

[23] F. Doshi-Velez, W. Li, Y. Battat, J. Park, B. Charrow, D. Curtis, S. Hemachandra, B. Reimer, J. Velez, C. Walsh, D. Fredette, N. Roy, and S. Teller. WiFi-based Localization for Improving Safety and Operational Efficiency in Residential Care Settings. *Journal of the American Medical Directors Association (JAMDA)*, pages –, 2012. accepted.

[24] R. Dukoff and T. Sunderland. Durable Power of Attorney and Informed Consent with Alzheimer's Disease Patients: A Clinical Study. *American Journal of Psychiatry*, 154(8):1070–1075, 1997.

[25] J. Forlizzi, C. DiSalvo, and F. Gemperle. Assistive Robotics and an Ecology of Elders Living Independently in Their Homes. *Human-Computer Interaction*, 19 (1):25–59, 2004.

[26] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the Second European Conference on Computational Learning Theory (EuroCOLT)*, pages 23–37, London, UK, 1995. Springer-Verlag.

[27] K. Georgila, J. Henderson, and O. Lemon. Learning User Simulations for Information State Update Dialogue Systems. In *Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH)*, 2005.

[28] J.J. Gibson. *The theory of affordances*, pages 67–82. Lawrence Erlbaum, 1977.

[29] J.R. Glass. A probabilistic framework for segment-based speech recognition. *Computer Speech and Language*, 17(2-3):137–152, 2003.

[30] A. Gruenstein. Response-based confidence annotation for spoken dialogue systems. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, SIGdial '08, pages 11–20, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[31] A. Gruenstein. *Toward widely-available and usable multimodal conversational interfaces.* PhD thesis, Massachusetts Institute of Technology, 2009.

[32] Timothy J. Hazen, Theresa Burianek, Joseph Polifroni, and Stephanie Seneff. Recognition confidence scoring and its use in speech understanding systems. *Computer Speech and Language*, pages 49–67, 2002.

[33] L.E. Hebert, L.A. Beckett, P.A. Scherr, and D.A. Evans. Annual Incidence of Alzheimer Disease in the United States Projected to the Years 2000 Through 2050. *Alzheimer Disease and Associated Disorders*, 15:169–173, 2001.

[34] S. Hemachandra, T. Kollar, N. Roy, and S. Teller. Following and interpreting narrated guided tours. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2574 –2579, May 2011.

[35] J. Hoey and P. Poupart. Solving POMDPs with Continuous or Large Discrete Observation Spaces. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1332–1338, 2005.

[36] R.A. Howard. *Dynamic Programming and Markov Processes.* The MIT Press, 1960, Cambridge, Massachusetts.

[37] A. Hub, J. Diepstraten, and T. Ertl. Design and development of an indoor navigation and object identification system for the blind. In *Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, Assets '04, pages 147–152, New York, NY, USA, 2004. ACM.

[38] J.C. Hughes and S.J. Louw. Electronic tagging of people with dementia who wander. *BMJ*, 325(7369):847–848, 2002.

[39] J.C. Hughes, J. Newby, S.J. Louw, G. Campbell, and J.L. Hutton. Ethical Issues and Tagging in Dementia: a Survey. *Journal of Ethics in Mental Health*, 3:1–6, 2009.

[40] H. Jiang. Confidence measures for speech recognition: A survey. *Speech Communication*, 45(4):455–470, April 2005.

[41] L.P. Kaelbling, M.L. Littman, and A.R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[42] Anja Kintsch and Rogerio Depaula. A framework for the adoption of assistive technology. In *Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, pages 1–10, 2002.

[43] C.K.Y. Lai and D.G. Arthur. Wandering behaviour in people with dementia. *Journal of Advanced Nursing*, 44(2):173–182, 2003.

[44] O. Lemon and I. Konstas. User simulations for context-sensitive speech recognition in spoken dialogue systems. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 505–513, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[45] E. Levin, R. Pieraccini, and W. Eckert. Learning dialogue strategies within the Markov decision process framework. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 72–79, dec 1997.

[46] E. Levin, R. Pieraccini, and W. Eckert. Using Markov decision process for learning dialogue strategies. In *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, volume 1, pages 201 –204 vol.1, may 1998.

[47] M.L. Littman. *Algorithms for Sequential Decision Making.* Dissertation, Brown University, 1996.

[48] M.L. Littman, A.R. Cassandra, and L.P. Kaelbling. Learning policies for partially observable environments: Scaling up. *Proceedings of the Twelfth International Conference on Machine Learning*, pages 362–370, 1995.

[49] Rebecca G. Logsdon, Linda Teri, Susan M. McCurry, Laura E. Gibbons, Walter A. Kukull, and Eric B. Larson. Wandering: A Significant Problem among Community Residing Individuals with Alzheimer's Disease. *Journals of Gerontology: Series B*, 53(5):294–299, 1998.

[50] W.C. Mann, K.J. Ottenbacher, L. Fraas, M. Tomita, and C.V. Granger. Effectiveness of assistive technology and environmental interventions in maintaining independence and reducing home care costs for the frail elderly. A randomized controlled trial. *Arch Fam Med*, 8(3):210–7, 1999.

[51] M. Mason. Speech Activity Detection Module. `https://groups.csail.mit.edu/rvsn/wiki/index.php?title=Speech_Acquisition_and_Recognition_using_GStreamer_and_SUMMIT`.

[52] G. Nejat, B. Allison, N. Gomez, and A. Rosenfeld. The Design of an Interactive Socially Assistive Robot for Patient Care. *ASME Conference Proceedings*, 2007 (43033):821–830, 2007.

[53] T. Paek and R. Pieraccini. Automating spoken dialogue management design using machine learning: An industry perspective. *Speech Communication*, 50 (8-9):716 – 729, 2008. Evaluating new methods and models for advanced speech-based interactive systems.

[54] K. Pahlavan, Xinrong Li, and J.P. Makela. Indoor geolocation science and technology. *Communications Magazine, IEEE*, 40(2):112–118, February 2002.

[55] J. Park, B. Charrow, D. Curtis, Y. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie. Growing an organic indoor location system. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 271–284, New York, NY, USA, 2010. ACM.

[56] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun. Towards robotic assistants in nursing homes: Challenges and results. *Robotics and Autonomous Systems*, 42(3-4):271 – 281, 2003.

[57] L. Robinson, D. Hutchings, H. O. Dickinson, L. Corner, F. Beyer, T. Finch, J.C. Hughes, A. Vanoli, C. Ballard, and J. Bond. Effectiveness and acceptability of non-pharmacological interventions to reduce wandering in dementia: a systematic review. *International Journal of Geriatric Psychiatry*, 22(1):9–22, 2007.

[58] N. Roy, J. Pineau, and S. Thrun. Spoken dialogue management using probabilistic reasoning. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 93–100, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[59] Nicholas Roy, Joelle Pineau, and Sebastian Thrun. Spoken Dialog Management for Robots. In *Proceedings of the 38th Annual Meeting of the Association for Computation Linguistics*, 2000.

[60] R. San-Segundo, B. Pellom, W. Ward, and J.M. Pardo. Confidence measures for dialogue management in the CU Communicator system. In *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, volume 2, pages 1237–1240, 2000.

[61] S. Singh, D. Litman, M. Kearns, and M. Walker. Optimizing Dialogue Management with Reinforcement Learning: Experiments with the NJFun System. *Journal of Artificial Intelligence Research*, 16:105–133, 2002.

[62] E.J. Sondik. *The Optimal Control of Partially Observable Markov Decision Processes.* PhD thesis, Stanford, California, 1971.

[63] S. Teller, Y. Battat, B. Charrow, D. Curtis, R. Ryan, J. Ledlie, and J. Hicks. Organic Indoor Location Discovery. Technical Report MSU-CSE-00-2, MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, 2008.

[64] J. Torres-Solis, M. Guan, E. Biddiss, and T. Chau. Navigation in smart environments using mediated reality tools. *Journal of Technology and Health Care*, 17:237–251, August 2009.

[65] M.A. Walker, D.J. Litman, C.A. Kamm, and A. Abella. PARADISE: a framework for evaluating spoken dialogue agents. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 271–280, 1997.

[66] J.D. Williams. Partially observable markov decision processes with continuous observations for dialogue management. In *Computer Speech and Language*, pages 393–422, 2005.

[67] J.D. Williams. A Case Study of Applying Decision Theory in the Real World: POMDPs and Spoken Dialog Systems. In *Decision Theory Models for Applications in Artificial Intelligence: Concepts and Solutions.* IGI Global, 2010.

[68] J.D. Williams and S. Young. Partially observable Markov decision processes for spoken dialog systems. *Computer Speech and Language*, 21(2):393 – 422, 2007.

[69] S. Young, J. Schatzmann, K. Weilhammer, and Hui Ye. In *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, volume 4, pages IV–149 -IV–152, april 2007.

[70] S. Young, M. Gasic, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, and K. Yu. The Hidden Information State model: A practical framework for POMDP-based spoken dialogue management. *Computer Speech and Language*, 24(2):150–174, 2010.

[71] Sheryl Zimmerman, Philip D. Sloane, Christianna S. Williams, Peter S. Reed, John S. Preisser, J. Kevin Eckert, Malaz Boustani, and Debra Dobbs. Dementia

Care and Quality of Life in Assisted Living and Nursing Homes. *The Gerontologist*, 45(suppl 1):133–146, 2005.