Situated Intelligent Interactive Systems

Zhou Yu

CMU-LTI-01-018

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Alan W Black
Alexander I. Rudnicky
Louis-Phillip Morency
Dan Bohus
David Suendermann-Oeft

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies*

Abstract

The recent wide usage of *Interactive Systems (or Dialog Systems)*, such as Apple's Siri has attracted a lot of attention. Besides personal assistants, dialog systems can also be applied in various domains, such as education, health care and entertainment. However, none of the current systems coordinate well with their users, leading to less than ideal performance. The ultimate goal for dialog research is to transform these current systems into real intelligent systems that can communicate with users effectively and naturally. There are three major challenges to this ultimate goal: first, how to make systems cooperate with users in a natural manner; second, how to provide adaptive and personalized experience to each user; and last, how to make multiple task systems transition from one task to another fluidly to achieve overall conversation effectiveness and naturalness. To address these challenges, I proposed a theoretical framework, ***Situated Intelligence (SI)***.

In the SI framework, we argue that three capabilities are needed to achieve natural and high quality conversations: (1) **situation awareness**, which can be approximated by multimodal information; (2) a rich repertoire of **conversation strategies** to regulate systems' situational contexts; and (3) a **global planning policy** that optimally chooses among different conversation strategies at run-time to achieve an overall natural conversation flow. We applied the SI framework to three different types of conversations: *non-task-oriented conversations* (e.g. social chat), *task-oriented conversations* (e.g. job interview training) and *conversations that interleave both task and no-task content* (e.g. movie promotion) and made a number of contributions in both algorithm development and end-to-end system building.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Dialog systems with a fixed goal have already achieved many functions, for example, providing bus schedule information [Raux et al., 2003], setting up an alarm clock [Apple, 2015] and making a restaurant reservation [He and Young, 2003]. We are at the point to consider how to make the dialog systems not only be able to complete certain simple tasks, but also be able to provide engaging and effective interaction experiences. I aim to create dialog systems that users enjoy interacting with repeatedly.

Human-human communication is highly coordinated, with participants expressing goals, developing common grounds and monitoring attention while maintaining an implicit social relationship. However, traditional automated agents were designed to focus on the task only and disregard users' mental states, which could lead to dull and less engaging interactions with a human. Even worse, people might get frustrated with the linearity of the conversation and lose motivation to complete the task. This is an even greater problem for interactions that are machine-initiated and the human has no intention to interact in the first place. For example, consider a robot that wanders in a public space and tries to ask people for help. Intuitively, one would expect the robot to know how to engage the human in a conversation and keep it going by using conversation strategies commonly used by humans. However, we are still far away from this ideal interaction. My thesis work brings speech and natural language processing and human-computer interaction together to work towards the ideal *situated intelligent agents* that can communicate with users effectively and naturally across different tasks.

The skills of situated intelligence have been argued to be indispensable and perhaps most important for success in human life [Goleman, 2007]. While traditional systems are socially ignorant [Pentland, 2005]. Because they do not account for the fact that human-human communication is always situated. FOr example, discussions are not just information exchanges but part of a bigger social interplay. Not all systems need the social intelligence and none will need all of the related skills humans have. Systems that have context independent tasks, such as finding bus schedules, will work well without the social intelligence. However, if we want to integrate such systems in future smart environments and aim to improve the quality of human life in complex settings, these systems are not sufficient. Computer systems with capabilities to sense agreement and attention that are able to adapt or respond to these signals in an appropriate way will likely be perceived as more natural, efficacious and trustworthy [Vinciarelli et al., 2009]. For example, in education, students' social signals inform the teacher their mental states and successful human teachers acknowledge these signals and work with them to improve students' learning. Thus I argue that conversational systems should do the same by employing tools that can accurately sense and interpret social signals and social contexts of users, learn successful context-dependent social behaviors, and use a proper socially adept presentation language to drive the actions of the system.

1.2 Contribution

The goal of the thesis is to develop a situated intelligence (SI) framework for interactive systems to enable system to be effective and natural. I also adapted the general SI framework to both non-task-oriented and task-oriented dialog systems. In the end, I built and deployed end-to-end systems to test our framework. The main expected contributions of my thesis are the SI framework and the end-to-end systems that adopted the framework.

The **SI** framework first empowers interactive systems with ***situation awareness***. Next, with ***conversation strategy*** the system coordinates its actions with users. In the end, ***global planning policy*** is used to achieve overall effectiveness and naturalness across multiple conversation turns and beyond. I will describe the framework in details in Chapter 2.

In practice, this thesis also introduced two dialog system software frameworks, TickTock and HALEF. TickTock is a stateless dialog framework, that supports non-task-oriented systems. It is modular and multi-threaded. It supports both typing and audiovisual inputs. TickTock is used as the system framework in the CMU Dialog System Labs course in both 2015 and 2016, as well as in the Amazon Alexa Prize competition. I implemented a social chatbot system under the TickTock framework applying the SI framework. The system is also one of the six chatbots for the shared tasks in the RE-WOCHAT workshop at two international conferences: LREC 2016 [D'Haro et al., b] and IVA 2016 [D'Haro et al., a]. HALEF is a task-oriented dialog framework, with the support of audiovisual input and output. It is modular and multi-threaded as well. I implemented a job interview training system to help unemployed people to gain more interview experience. The situated intelligent job interview system is in the process of classroom deployment in countries with a high demand in English learning, such as China and Brazil.

1.3 Outline

I will talk about the formulation of the SI framework and then the addaptations to different types of conversations. Chapter 2 gives an overview of the SI framework. Chapter 3 gives an overview of the details on applying the SI framework to address different challenges in three types of conversations. Then, I cover the non-task-oriented conversation systems (social chatbots) in Chapter 4-9, the task-oriented conversation systems (e.g. a job interview training system) in Chapter 10-11, systems that interleave task and non-task content (a movie promotion system) in Chapter 12 and I conclude in Chapter 13.

Chapter 2

Situated Intelligence Framework

The **Situated Intelligent (SI)** first empowers interactive systems with ***situation awareness***. Next, with ***conversation strategy*** the system coordinates its actions with users. In the end, ***global planning policy*** is used to achieve overall effectiveness and naturalness across multiple conversation turns and beyond. Before we go into details about each component of the SI, I will first go over some related previous work.

2.1 Related Work

Some dialog systems are designed to take user information beyond speech as input signal in planning dialog actions. Bohus and Horvitz (2009) used multimodal behavior to predict engagement. They modeled engagement as a binary problem to control the start and the end of the conversation [Bohus and Horvitz, 2009]. This work first introduced multimodal user behaviors in real-time dialog system action planning with respect to user engagement tracking. However the action is restricted to open or close conversations. Later [Skantze et al., 2015] introduced a multimodal dialog system that uses user gaze information in deciding turn-taking opportunities. In another example, situated incremental speech synthesis that accommodates user's cognitive load was shown to improve user experience but not user's task performances in a in-car navigation task [Kousidis et al., 2014a]. However, the system only has simple rules and pre-designed strategies that can change system's action only within a conversational turn. The actions will not affect the flow of the conversation. These projects mentioned focused on a single task in the conversation, the dialog actions that deal with the situation contexts are executed based on pre-built rules. These actions only changes the conversation within a single conversation turn, thus not able to affect the flow of the conversation. In this thesis, we proposed the SI framework that managers situation context with global policy that optimizes the entire conversation globally across multiple conversation turns and beyond. The SI framework is also general enough to accommodate all types of conversations, such as non-task-oriented, task-oriented conversations.

In another example, [Forbes-Riley and Litman, 2012] designed a Wizard-of-Oz spoken dialog system that adapts to student's disengagement and uncertainty in an education task. They found that the adaptive system increased the learning motivation, and breaks the negative correlation of the task success and disengagement and reduced the likelihood of continued disengagement. However, the system only uses features extracted from speech transcription to detect student disengagement, thus resulting in less ideal engagement detection performance. Thus, the low performance of the engagement predictor made building fully-automated systems infeasible. In this thesis we based the SI framework to build end-to-end fully autonomous systems for different types of conversations.

2.2 Situation Awareness

Situation awareness refers to the system's awareness of its context, which includes various aspects, such as individual partner's behaviors, interpersonal relationships among a group of partners, and even their social and cultural contexts. An example situation context is a partner's engagement in conversations. Literature suggests that situation awareness is a crucial component in assessing human communication competencies besides conversational strategies [Gumperz, 1982]. Therefore, I believe enabling interactive systems with situation awareness is the first step towards effective and natural conversations. I used *multimodal machine learning* methods to learn the representations of users' behaviors automatically from different modalities (e.g. eye gaze from vision modality, speech loudness from audio modality). I then use these quantified behaviors to infer the system's situation context. I have studied different types of situated contexts, such as mental states (e.g. engagement [Yu et al., 2016b] and attention [Yu et al., 2016g]), user psychological conditions (e.g. depression and PTSD [Yu et al., 2013b]), user ability (e.g. language proficiency [Yu et al., 2015c]), interpersonal relationships between users (e.g. friendship [Yu et al., 2013a]); and the users' cultural backgrounds [Yu et al., 2016a]. In particular, I have contributed to the computational social science field by modeling engagement across different conversation types and user groups. One interesting finding is that Chinese users prefer a more responsive dialog system more than Americans, as in Chinese culture, people value immediate feedback which has the function of maintaining interpersonal harmony [Hofstede and Hofstede, 2001]. However, having situation awareness is not sufficient, as the system needs to take interventions to be effective and natural.

2.3 Conversation Strategy

I also encode conversation theories in conversation strategies to foster user coordination, understanding and adaptation. In particular, I designed three types of strategies. *Active participation strategies* engage users by actively contributing to the conversation, such as asking more information on the current topic [Wendler, 2014]. *Grounding strategies* assist open-domain natural language understanding. Grounding strategies were automatically synthesized via leveraging knowledge-base (e.g. Google Knowledge Graph) information and natural language processing algorithms, such as named-entity detection and statistical language generation. For instance, if an ambiguous named-entity "Clinton" is mentioned, the system would provide users with details of two popular "Clintons" from the knowledge base to collaboratively resolve the ambiguity with users. *Personalized strategies* adapt to users by leveraging automatically extracted information from individual user's conversation history. An example personalized strategy is to suggest talking more about movies knowing the user was previously engaged in this topic.

2.4 Global Planning Policy

Global planning policy refers to a policy that considers previous interaction history in planning a sequence of actions to achieve optimum performance globally. In spoken dialog research, the most popular theory is to model conversations as MDPs [Misu et al., 2012] and POMDPs [Williams and Young, 2007], so the system considers previous history in dialog state tracking (dialog states are pre-defined, e.g. request user name). I adopted the same paradigm and used reinforcement learning methods (e.g. Q-learning) to achieve overall conversation effectiveness and naturalness. The policy considers both real-time situation contexts and conversation history, such as strategies used before (state variables) in selecting conversation strategies (actions) to achieve overall interaction effectiveness and naturalness (reward function) [Yu et al., 2016g]. This is also the first work in applying reinforcement learning methods to model open-domain dialogs without a pre-built agenda.

The SI framework is general and can be applied to all types of conversations. We will discuss the application in each type of conversation in the fallowing chapters.

Chapter 3

Application Overview

The Situated Intelligence (SI) framework leverages the situation awareness of the system in performing actions to achieve long-term effectiveness and naturalness. I applied the SI framework to *non-task-oriented interactive systems* (*social chatbots*) (§4-9), *task-oriented interactive systems*, such as *job interview training systems* (§10-11) and *systems that interleave social and task content* (§12). In this chapter, I will give an overview of why and how I applied the SI framework to all these different types of conversational systems.

3.1 Non-Task-Oriented Interactive Systems

Non-task-oriented conversation (or social chatting) refers to everyday chatting without a specific task or goal. Recently, chatbots have attracted much attention in both research and industry. Jiang et al. found that more than 30% of the user utterances addressing Microsoft Cortana are social chats [Jiang et al., 2015], which reflects the general desire of such functionality. Yet, social chatting systems are not simply cool toys, could serve in various areas for social good. Take health care as an example, social chatbots can benefit elderly populations by providing social companionship. Social chatting could also smooth complex interactions and build rapport with partners. For example, in a clinical interview that assesses mental illness, the clinician will start the conversation with social chatting to build rapport before he moves on to diagnostic questions. Literature suggests rapport is essential for self-disclosure. Therefore, only with good rapport would the user disclose useful information for diagnostic purposes. Social chatting also serves for the purpose of collecting information that would benefit the task of the conversation. For example, the real estate agents would use social chatting to collect customers' background information in order to make a more targeted recommendation.

3.1.1 Challenge

There are three major challenges to effective and natural social conversation systems. 1) Open-ended content: unlike the traditional task systems that have a pre-defined agenda, a social conversation system allows users to talk about anything, thus making natural language understanding more difficult. 2) Personalization: social chat is personal; users expect a unique and personalized experience, especially after interacting with the system repeatedly. 3) Smooth transition: unlike traditional single-task systems, social conversation systems need to transition from one conversation topic to another smoothly .

3.1.2 Related Work

Lat century, during the last heat wave of AI, there was a lot of effort in developing chatbots that can pass the Turing test. In 1966, Eliza, a hand-crafted rule-based chatbot system was developed [Weizenbaum, 1966]. It used rhetorical strategies to elicit more information from users and was developed mainly for mental illness therapies. Since 1990s, the Loebner Prize, an annual competition to develop systems that can pass the Turing test, was very popular. A lot of chatbots were developed in this context, for example Mitsuku, the 2016 winner developed by Steve Worswick (http://www.mitsuku.com/). Recently, the good progress in machine learning and natural language processing made chatbots a popular topic in AI research as well. A lot of effort has been put into automatic dialog generations for social chatbots. For example, IRIS system leverages natural language technologies, such as information retrieval [Banchs and Li, 2012]. In industry, Microsoft XiaoBing is very popular in Chinese social media. Research on sequence-to-sequence models also contributed to response generation methods to simulate social chatting conversations as well.[Sordoni et al., 2015, Li et al., 2016a].

3.1.3 Approach

To address challenges mentioned above, I applied the proposed SI framework to social conversation systems. I focus mainly on one type of situation intelligence, engagement. ***Engagement*** refers to the user's interest to continue the conversation [Peters et al., 2005]. It is one of the most critical situation contexts, as it ensures that the communication channel is open. To track user engagement in real time, I built a statistical model that leverages automatically quantified human behaviors, such as speech loudness and smiles via supervised machine learning methods. In Chapter 6 and Chapter 7, I will describe the details about engagement modeling.

I also designed three types of conversation strategies, such as the **active participation strategies** to coordinate with user engagement. Active participation strategy refers to a set of conversational strategies, such as, elicit more information on the current topic or change the current topic. They are designed based on a communication theory that asserts taking an active stand in conversation flow management improves partner's engagement [Wendler, 2014]. Additionally, **grounding strategies** and **personalized strategies** were used to leverage knowledge base information. Grounding strategy is designed to assist the language understanding in open-domain conversations. Grounding strategy resolves ambiguous entities and unknown words via a combination of different natural language processing algorithms, such as named-entity detection and statistical language generation. An example grounding strategy for entity disambiguation is when users mentioned "Clinton". "Clinton" can refer to "Bill Clinton" or "Hillary Clinton", thus the system asks "Which Clinton are you talking about? Bill Clinton, the 42rd president of the United States or Hillary Clinton, the secretary of the state?" Personalized strategy refers to actions that embed user's personal knowledge and is designed to achieve a personalized user experience. An example personalized strategy is to suggest "Captain America 2" given that a user's favorite movie type is superheroes [Yu et al., 2016f]. I will describe the detailed implementations of conversation strategies in Chapter 8.

Global planning policy is finally introduced to consider system situational contexts to achieve long-term conversation effectiveness and naturalness. The learned policy sequentially selects among different strategies and generated system responses produced by a sequence-to-sequence neural model [Kiros et al., 2015] to coordinate. The policy considers conversation history to achieve optimal overall conversation coherence, as well as conversation depth, and content varieties [Yu et al., 2016g]. In Chapter 9, I will discuss the policy trained with reinforcement learning algorithms in detail.

3.1.4 System

I implemented an end-to-end social chatbot software framework: TickTock, based on algorithms described. TickTock is modular, multi-threaded and extensible to different tasks. It provides web-browser access for users and supports different communication media, such as typing, audio and audiovisual interactions. I will describe the system framework in detail in Chapter 4. In addition, in Chapter 5, I leveraged crowd power to expanding conversation database and conducting system evaluations.

3.1.5 Impact

TickTock is used as the system framework in the CMU Dialog System Labs course in both 2015 and 2016, as well as the base system for the Amazon Alexa Prize competition. The system is also one of the six chatbots in the shared tasks in the RE-WOCHAT workshop at two international conferences: LREC 2016 [D'Haro et al., b] and IVA 2016 [D'Haro et al., a]. It can be accessed via http://www.cmuticktock.org/. More than a thousand users have interacted with it so far.

3.2 Task-Oriented Systems

Task-oriented conversations refer to dialogs with a specific task to complete. The system works with users to complete a task through natural language interaction. For example, a job interview training system acts as an interviewer to train users for job interviews. Conversation ability is considered one crucial ability in assessing second-language proficiency [He and Young, 1998]. In addition, preparing unemployed people for job interviews has an impact on social good. Therefore, through an interdisciplinary collaborative effort with David Suendermann-Oeft and other researchers in Educational Testing Service (ETS), I applied the SI framework to this task [Yu et al., 2016c].

3.2.1 Challenge

Unlike non-task-oriented systems, task systems have pre-defined tasks. However, it still requires adaptation towards different users. Another challenge in designing task-oriented system is that it requires interdisciplinary knowledge. Take the job interview training task as an example, it calls for knowledge in user communication training and behavioral assessment. Therefore, I applied the SI framework to the task under the guidance of experts from both education and people analytics.

3.2.2 Related Work

Task-oriented systems have been the main focus of the spoken dialog community since 1990. In the early 2000, DARPA Communicator program contributed to the boost of automated spoken dialog systems. The frame-based systems developed from the program have been the dominant architecture till now for task-oriented systems. These frame-based dialog systems have a pre-defined agenda of the task. Usually these systems only can handle a limited number of tasks, most of them only has one task. The representative systems came from the Communicator program is the CMU Let's Go system that provides bus information [Raux et al., 2003]. Task oriented dialog systems have various applications to provide both real-life services and research support in all fields of science. All the mobile personal assistants, such as Apple's Siri, Google's Google Now, Microsoft's Cortana and Amazon's Echo are task-oriented systems that aim to complete tasks with users. Task-oriented systems have also used in various other research fields, for example, to monitor elderly patients [Bickmore and Picard, 2005] and even serve as nurses to discharge patients [Bickmore et al., 2009] in health care domain. They have also used in education, such as providing maths tutoring [Forbes-Riley and Litman, 2012], language training [Raux and Eskenazi, 2004].

In this thesis, I applied the SI framework for frame-based task-oriented systems to coordinate the system and user behaviors. In particular, I looked into one type of training systems: job interview training. I will talk about the details of our work on non-task-oriented systems in Chapter 10 and 11.

3.2.3 Approach

To address challenges mentioned, I adapted the computational engagement model from social chatting to this task via transfer learning methods [Daumé III, 2009]. To regulate user's disengagement, I designed **positive feedback strategy** based on job interview literature [Macan, 2009]. The content of the generated strategy is also infused with dialog content automatically. One example strategy is: *I heard that Freecane is a really good school*, after user said that he went to Freecane. Through interdisciplinary collaboration, the conversation flow is designed to be relatively deterministic in order to cover the assessment scenarios valued in human analytics. So reinforcement learning policies are not necessary and a simply deterministic policy was used to select automatically generated positive strategy to regulate user's disengagement. I will describe the details of the conversation strategy and policy in Chapter 11.

3.2.4 System

I implemented a situated intelligent job interview system under HALEF, a task-oriented dialog framework that I co-designed and implemented with researchers from ETS [Yu et al., 2016d]. I will talk about HALEF in Chapter 10 in detail. The system can be accessed via web-browser and it can stream user's audio and video recording to cloud machines in real time. This design reduced the cost and effort in data collection and system evaluation for interactive systems.

3.2.5 Impact

The situated intelligent job interview system is in the process of classroom deployment in countries with a high demand in English learning, such as China and Brazil. Teachers would assign homework that requires students to use the system outside classroom for training.

3.3 Systems That Interleave Task and Non-Task Content

I observe that most human-human conversations are not purely talking about a certain task or random social content. I interleave social and task conversations all the time in conversations [Schegloff, 1968]. These interactions happen most frequently when the two conversational partners' intentions are not calibrated. Therefore, I designed a type of system that interleaves task and non-task content to handle tasks that users' intentions are not explicit, such as recommendation tasks. The system uses social content to keep users in the conversation so that the user would have a higher chance to complete the task. These systems have various usages in different areas, such as survey, education and health care.

3.3.1 Challenge

None of the previous work has tackled systems that interleave task and non-task content so far. The main challenge of these systems lies in how to perform smooth transition between social content and task content.

3.3.2 Approach

I also extend the SI framework to systems that interleaves social and task content. I used a policy trained by reinforcement learning algorithms to choose among task and non-task response candidates to reach the smooth transition.

3.3.3 System

I implemented an example system for movie promotion. It promotes a specific movie based on user's interests. Through experiments, I found that a system interleaves social content with task content outperformed a pure task system on both the task success rate and self-reported user engagement rating. I also used the example system to collect audience information that interests the film industries and demonstrated the system's ability to collect information that correlates with statistics provided by the film industries. I will describe the details in Chapter 12.

Chapter 4

TickTock, A Non-Task-Oriented Dialog System Framework

To study non-task-oriented conversations, I developed a non-task-oriented dialog system (chatbot) framework that is capable of conducting free-form conversations. In contrast to the traditional task-driven systems, which are designed to acquire information, provide feedback, or negotiate constraints with the human. A chatbot removes any built-in task for the user and its success depends on keeping the human in the conversation. Thus, task completion is no longer an applicable metric, I choose to optimize on generating appropriate system responses.

TickTock generates responses using keyword retrieval methods and is powered with conversational strategies. There are two versions of TickTock based on the modalities available for user input and system output. Multimodal-TickTock takes both audio and visual information from the user and replies the user with synthesized speech and an animated head with basic expressions. It is only available in a stand-alone version, which requires users to interact it in the lab face-to-face. The other version is Text-TickTock which takes user's typed text as input and reply with text as well. Through a web socket, I connect Text-TickTock to a webpage implemented in PHP. People can access TickTock through any browser and multiple people can talk to it simultaneously. Text-TickTock is created to collect conversations with less effort in order to expand system database and to improve system response appropriateness.

Figure 4.1 shows the architecture of the TickTock conversational agent. I use Google Automatic Speech Recognition (ASR), OpenFace and a purpose-build Natural Language Understanding (NLU) to process user input. I use a template-based Natural Language Generation (NLG), Flite Text to Speech (TTS) [Black and Lenzo, 2001] and an animated head to generate system response. An earlier version of a free-form conversational agent guides our design decisions [Marge et al., 2010]. The asynchronous message passing mechanism is implemented based on

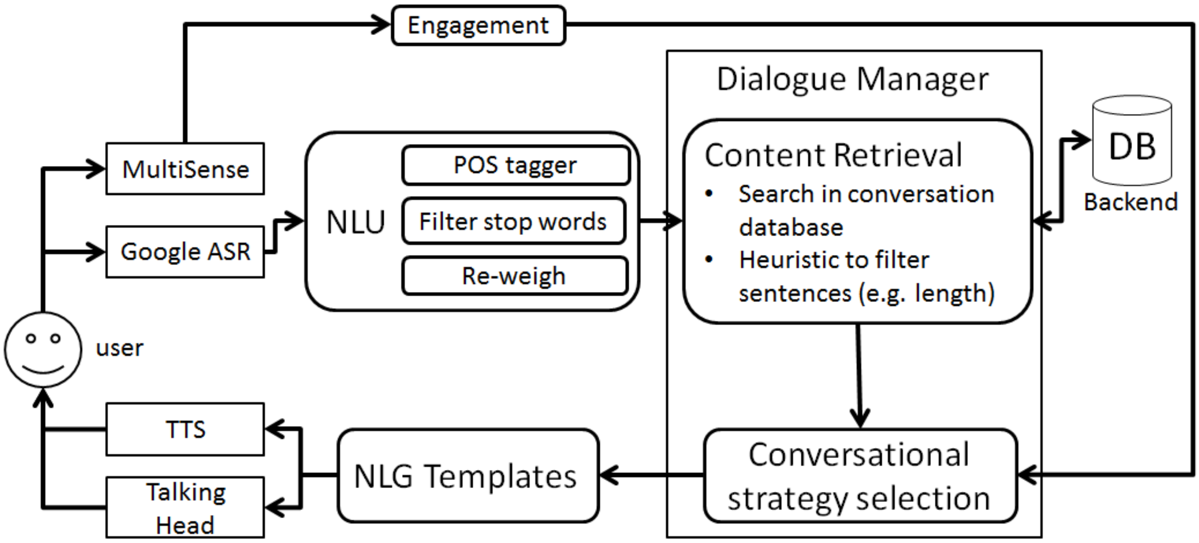


Figure 4.1: The architecture of TickTock

Olympus Dialog Framework [Bohus et al., 2007].

4.1 Database

In Text-TickTock 1.0 and Multimodal-TickTock 1.0, the database consists of question-response pairs from CNN Interview Transcripts from the "Piers Morgan Tonight" Show . The corpus has 767 Interviews in total and each interview is between 500 to 1,000 sentences. To construct our database, I used a rule-based question identification method, which simply means searching for tokens such as "?", "How", "Wh-", etc. to identify questions and then extracted the consecutive utterance of the other speaker as the response to that question. In Chapter 5, I expanded the database by including targeted human generated conversations via crowdsourcing for Text-TickTock 2.0. The later versions of TickTock in both the text or multimodal used the expanded dataset.

4.2 User Input Process

I perform automatic speech recognition on user speech using Google ASR API. Sphinx is used for user speech start and end detection. If the user interrupts the system, the system will stop speaking immediately and let the user finish first. I also used Sphinx to extract acoustic features from user speech, such as loudness of speech and pitch. These features are then directed to the engagement module to predict engagement in real time.

Human visual signals are captured by OpenFace [Scherer et al., 2012]. OpenFace is able to detect and track user's face. It outputs the 3D head position, the action unit of the face that later used for engagement prediction.

4.3 Response Retrieval

The ASR result is sent to the NLU component. I first performed POS tagging [Toutanova et al., 2003] and removed stop words; heuristics are used to determine weights for each POS tag groups e.g. nouns have higher weight compared to other verbs. The sets weight 2 to nouns ('NN', 'NNS','NNP' and 'NNPS') and 1 to prepositions ('PRP' and 'PRP$') and verbs ('VBD', 'VBG', 'VBN', 'VBN', 'VBP', and 'VBZ').

I use keyword matching [Martin, 2002] for content retrieval. The goal is to generate coherent responses efficiently without deep understanding of the context, which is useful in a non-task-oriented system, and is motivated by lexical cohesion in modeling discourse. The coherence can be reflected by the repetition of lexicon items. I first search for matched keywords in the database and calculate the weighted sum. Finally, I normalize the score by diving it by the length of the retrieved utterance and obtain the retrieval confidence. I also filter out inappropriate content, exclude utterance that is longer than 15 words and remove characters that do not exist in speech, such as punctuations.

4.4 Engagement Module

The engagement module tracks user's engagement state in real time, and updates the state every conversational turn. It takes multimodal information as input from the ASR, OpenFace and Audio processing component. An engagement model that is pre-trained to predict user engagement based on these information. In the end, the predicted user engagement state goes to the conversational strategy selection sub-module in the Dialog Manager component. I will explain how to train the engagement model in Chapter 8 and 9 in detail. The Multimodal-TickTock 4.0 and later versions activated this module, while all the previous versions are not. The engagement module could be replaced by any situation context that is critical, such as user attention.

4.5 Dialog Manger

The dialog manger has two components, the conversational strategies and the policy that selects among these conversational strategies. Dialog Manger takes two signals into consideration, the retrieval confidence and the engagement confidence. In TickTock 1.0, the engagement confidence is not applicable, thus the policy only considers the retrieval confidence when selecting conversational strategies.

I determine a threshold for the retrieval confidence score based on experimental heuristics. The threshold can be tuned to make the system appear more active or more passive. Higher thresholds correspond to more active systems. In Text-TickTock 1.0 and Multimodal-TickTock 1.0, I design two strategies for cases that are below and above the threshold. At each dialogue turn, I randomly chose between the two strategies. If the retrieval confidence score is low, meaning no good response is obtained, I use strategies to change the current topic by proposing a new topic, such as “sports” or“music” or I close the current topic using an open question, such as “I don’t know, could you tell me something interesting?” If the retrieval confidence score is higher than the threshold, I choose between two strategies, one where I return the retrieved response and another where I ask about the user’s opinion, for example: “What do you think?” These simple strategies are designed as baselines to sustain the flow of conversation. In Text-TickTock 2.0, I expand the conversational strategies by adding extra conversational strategies. I will describe these additional strategies in Chapter 8. In Text-TickTock 3.0, I develop a dialog policy to select among these conversational strategies to optimize system response appropriateness. I will describe the policy in detail in Chapter 9.

4.6 Text-to-Speech and Talking head

Flite TTS [Black and Lenzo, 2001] is used for speech synthesis for Chinese and Baidu TTS API is used for Mandarin. TickTock also uses a 2D talking head to communicate its internal state. There are in total four animated videos [Pappu et al., 2013] that represent internal state, such as smiling, confusing, sleep talking and sleeping. Figure 4.2 shows all four states of TickTock. Initially, when there is no user interacting with the system, the talking head is in a semi-active state and appears to be sleeping. Once it hears the user, it wakes up; the lips move when the agent is speaking. It looks confused when misunderstanding occurs. The head is designed to be gender and culture ambiguous. This design aims to avoid the uncanny valley dilemma, so that the users would not expect realistic human-like behaviors from the system.

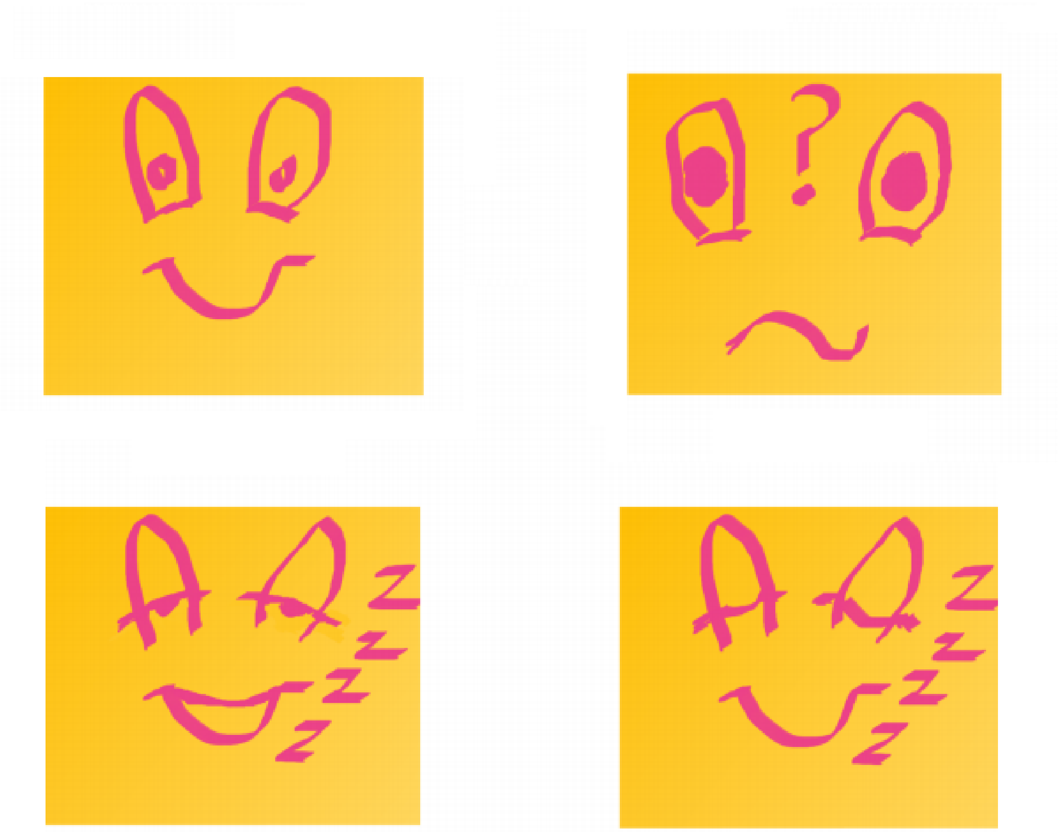


Figure 4.2: The smiley face of TickTock

4.7 TickTock in Mandarin

I created Chinese-Multimodal-TickTock 1.0 that speaks Mandarin, by changing the database, the ASR and the synthesis modules of the original version of Multimodal-TickTock 1.0. This version of TickTock is created to study whether people from different culture backgrounds use similar behaviors to express engagement or not.

The original TickTock's response generation model is trained on an open-domain dialog corpus formed by American popular social media, such as CNN interview corpus, TV show "Friends" and Reddit. Translated text usually appears unnatural in the target language, which may lead to less believability of the agent's culture identity. Thus, I used similar social media materials as the American version, but is originally created in Mandarin: Xinlang Aiwen (similar to Quora) and TV show "Love apartment". In total, the data set has around 1000 question-response pairs.

I replaced the Google Automatic Speech Recognizer (ASR) and Flite Text-to-speech (TTS) [Black and Lenzo, 2001] with Baidu ASR and Baidu TTS respectively to support automatic Mandarin recognition and synthesis. The change is due to the fact Google service is not applicable in mainland China and Flite does not have an advanced Chinese synthesis model. I chose Baidu's service because it provides relative good quality synthesis and is easy to integrate. I did not change the appearance of the agent for different cultural contexts, as the agent's appearance is designed to be culturally ambiguous.

4.8 Conclusion

The TickTock framework is designed to facilitate fast prototyping. It is implemented in python (except the multimodal processing module is implemented in C++). It is platform independent and easy to modify. Another advantage of TickTock is modular, so different users can work on different modules separately and integrate together in the end with little effort. The module within TickTock is also very flexible and can be replaced easily. For example, I can replace the keyword retrieval method with other response generation methods, such as sequence-to-sequence response generation easily. The modular and platform independent advantage makes TickTock the framework for CMU Dialog System Labs course in both 2015 and 2016, as well as in the Amazon Alexa Prize competition. The system is also one of the six chatbots in the shared tasks in the RE-WOCHAT workshop at two international conferences: LREC 2016 [D'Haro et al., b] and IVA 2016 [D'Haro et al., a].

Chapter 5

CrowdSourcing for Dialog Data Collection and Dialog System Evaluation

Last chapter, I described the basic system architecture of a chatbot system, TickTock. In this chapter, I describe an approach that starts from an existing conversation database and makes use of crowd-sourced data to augment the conversation database to improve the chatbot performance. Experiments found that the version with the expanded database is rated significantly better in terms of the response level appropriateness and the overall ability to engage users. In the process, I developed a pipeline that enables the user to interact with the chatbot via webpages, which makes recruiting a large number of users for system evaluation possible.

5.1 Introduction

Chatbots do not have any specific goal that guides the interaction. Consequently, traditional evaluation metrics, such as task completion rate, are no longer appropriate. The difficulty of evaluation is intrinsic as each conversation is interactive, and the same conversation will not occur more than once; one slightly different answer will lead to a completely different conversation. Moreover there is no clear sense of when such a conversation is "complete". It is not possible to design a pipeline to evaluate such systems in a batch mode, nor is it easy to equate participants on various dimensions that may influence their behavior.

In addition to the difficulty of evaluating a chatbot, another challenge is identifying an appropriate database to build chatbots. Ideally, it should be a database that has the same conversation content as the future users' conversations. However, It is not clear what is the best strategy for designing a database that works best for general usages.

Current chatbots use a variety of methods to generate responses, such as machine translation

[Ritter et al., 2011], retrieval based response selection [Banchs and Li, 2012], and recurrent neural network sequence generation [Vinyals and Le, 2015]. Yet, the databases they use to power their systems have very little variability. Some systems used micro-blogs, such as Twitter conversations [Ritter et al., 2011] and others used movie subtitles [Ameixa et al., 2014, Banchs and Li, 2012]. There is also research that used Twitter as a database but switched to ask the human to generate responses in the crowdsourcing platform in real time when the database failed to have an appropriate response [Bessho et al., 2012]. Most of the work reported above have no real user evaluation or a small group of people for evaluation. Only two kinds of databases have been used, movie subtitles and micro-blogs. In this work, I focus on how to generate appropriate databases for chatbots and conduct evaluations for chatbots by leveraging crowdsourcing resources.

5.2 Methodology

The method that I used involves designing three crowdsourcing tasks to expand TickTock's database. The first task is "the conversation generation task", in which a user interacts with TickTock via typing. The second task is "the conversation rating task", in which the user rates how appropriate TickTock's response is per conversational turn. The third task is "the conversation correction task", in which the user generates appropriate responses for TickTock. I only selected the conversational turns that were rated not appropriate in the second task to be corrected by users.

I recruited participants on the Amazon Mechanical Turk Platform with criteria of: a higher than 95% life time approval rate, completed more than 50 tasks, and are located in the United States. The TickTock system is implemented in Python, making it platform independent. (Please refer to Chapter 3 for details of TickTock system implementation.) In this work, I used Text-TickTock for the experiment as it can be easily accessed via web browser.

After I collected sufficient data from the above three tasks, I expanded our database by adding the responses that have high appropriateness scores and the ones that are generated by users in the third task. I named the system with the expanded database Text-TickTock 2.0. Then I put it on the Amazon Mechanical Turk Platform to collect more conversation from users for evaluations. In the end, I calculated the appropriateness rating distribution of the two versions to see if Text-TickTock 2.0 outperforms Text-TickTock 1.0 or not.

I also collected user subjective ratings towards the two versions. I asked users to report how engaged they felt during the interaction. I also asked users who have interacted with both versions which version they prefer.

Label	Definition	Example
Inappropriate	Not coherent with the user utterance	*Participant*: How old are you? *TickTock*: Apple.
Interpretable	Related and can be interpreted	*Participant*: How old are you? *TickTock*: That's too big a question for me to answer.
Appropriate	Coherent with the user utterance	*Participant*: How is the weather today? *TickTock*: Very good.

Table 5.1: Appropriateness rating scheme.

5.2.1 Mechanical Turk Study Design

The three tasks I designed are explained in detail below.

Conversation Generation Task

I asked the user to interact with the system via the web page for more than 10 conversational turns. I also asked the user what they liked and disliked about the task for the purpose of improving the task quality. I collected 50 conversations for each version. Forty-four people interacted with Text-TickTock 1.0 (I allow people to do the same task multiple times), resulting in 589 conversational turns. The conversation length on average is 11.9(SD=1.9). Twenty-nine people interacted with Text-TickTock 2.0, resulting in 590 conversational turns. The conversation length on average is 11.8(SD=4.2).

Conversation Rating Task

Once users finished their conversations with TickTock, I ask them to rate the appropriateness of each version's responses with respect to their utterances. Table 5.1 describes the annotation scheme. The user can see the entire dialog on the web page. They can choose from: “Inappropriate”, “Interpretable” and “Appropriate”.

An expert assesses the quality of the user rating by sampling 10% of the data and re-annotating them. If, I collapsed the “Inappropriate” and “Interpretable” categories and found that the agreement between the participant's self-rated appropriateness and the expert-rated appropriateness is relatively high (Kappa=0.73). Therefore, the user ratings are proved to be reliable. Table 5.2 shows an example dialog with appropriateness ratings.

Participant	TickTock(Appropriateness)
Hello how are you?	I'm feeling quite good. (3)
Anything new going on with you?	Of course. What do you think? (3)
I am not sure. What's on your mind?	Well, you know, I'm not sure. let me tell you something, Politicians and diapers have one thing in common. They should both be changed regularly, and for the same reason. (3)
Wow. That funny and insightful.	Uhh, I can't answer this. Can I talk about something else? (2)
How about movies. Have you seen any good movies?	Of course. What do you think? (2)
What have you seen?	Yes. (2)
I love super hero movies. Have you seen Deadpool?	Uh-huh, I do. (2)

Table 5.2: An example conversation with TickTock

Conversation Correction Task

I showed the user three consecutive utterances: the user utterance, TickTock's reply to the user utterance, the user's reply to TickTock's reply. I asked the user to pretend to be TickTock and respond to the participant's last reply. The original TickTock's response is not shown to the user to avoid influencing the user's response. In total, 28 Turkers participated in this task.

To assess the quality of the generated response, I randomly sampled 10% of the corrected conversational turns and asked an expert to rate their appropriateness. I found that 82.8% of the responses are appropriate, and the inappropriate responses are irregular words produced by people gaming the system. I filtered these bad responses out using a simple regular expression. The percentage of appropriate response increased to 100% after filtering. I corrected this problem by putting a checking in the script to not allow user to submit irregular responses. The reason that this task is easy for users is due to their experience in conducting conversations with other humans.

5.2.2 Experiments and Results

We conducted the experiment by first deploying the Text-TickTock 1.0 to collect 50 conversations. Then I conducted the three tasks mentioned above to expand the conversation database. Finally, I created Text-TickTock 2.0 by only replacing the conversation database of Text-TickTock 1.0.

After performing the experiment, I found our method to expand dataset does improve the performance of the chatbot. Table 5.3 shows the response appropriateness distribution of the two versions. The inappropriateness turn ratio dropped from 55% to 34% by expanding the

Version	Inappropriate	Interpretable	Appropriate
1	321 (55%)	138 (23%)	130 (22%)
2	200 (34%)	242 (41%)	148 (25%)

Table 5.3: The turn appropriateness distribution in data collected using Text-TickTock 1.0 and 2.0

database with appropriate question-response pairs. The average self-reported engagement score is 2.4 (out of 5) in the Text-TickTock 1.0 experiment and 3.6 (out of 5) in the Text-TickTock 2.0 experiment. I observe that Text-TickTock 2.0 performs better than Text-TickTock 1.0 on the per-turn appropriateness measure and the per-interaction user engagement measure. There are only around 10% of the users (8 out of 73), who interacted with both versions of the system. Among them, five of them preferred the second version, two of them thought the two versions are the same, and one of them was not sure.

In addition to appropriateness and engagement measurements, I analyzed the lexical information in user responses. After filtering the stop words, I found that none of the top 50 most used words is content word. Most of the top ranked words are functional words, such as “you”, “I”, “is”, “do”, “are”, “what”, etc. This is expected because similar phenomena could be found in casual conversations among humans as well. Among all the words in the utterances, 9.4% of them are pronouns, which is expected in casual conversations as well. I also found the top ranked nouns are “music”, “movie”, “game”, and “sense”. “Music” and “movies” are popular because TickTock would change to these topics when it’s retrieval confidence is low. “Sense” is popular because some participants got annoyed and said “You don’t make sense.” or “It doesn’t make sense.”

I randomly selected 10% of the data and found that the quality of the users’ inputs is not always very satisfying. Because some users fail to provide content information. Some ask Ticktock simple questions, some go along with whatever TickTock says instead of pushing the conversation topic deeper. It’s difficult to elicit high quality user utterance , currently no chatbot is able to achieve that. I propose to include other measurements than system appropriateness, such as user utterance quality, conversation depth, to improve the overall conversation quality in the future.

5.3 Conclusion

In this chapter, I described a semi-automatic method to expand the chatbot database via crowdsourcing. I proved that with the expanded database, the chatbot achieves more turn level appropriateness and better overall engagement. I developed a software pipeline that enables the user to interact with the chatbot via webpages, which makes recruiting a large number of users for system evaluation possible.

I also found that users like to express their opinions in a everyday chatting setting, for example, “I like movies a lot”. Thus pronouns, such as, “you” and “I” appear frequently in these conversations. I also found, similar to human-human conversations, people use a lot of functional words when talking to a dialog system about everyday topics. In the future, I intend to not only improve the appropriateness of the system, but also design methods to illicit more topic consistent information from the users, in order to improve conversation quality.

Chapter 6

Understanding Engagement

In social chatting, user engagement is an important aspect of situation context. We first established an annotation scheme to quantify engagement in a non-task-oriented conversational setting. Then, we used automatically harvested features to approximate human behaviors, such as smiles, that were suggested by previous human-human communication literature as correlated with user engagement. Then we performed statistical analysis to uncover how these behaviors are related to engagement in different conversational culture setting. We found that people from different culture backgrounds express engagement differently when talking to a chatbot of their own cultures. For example Chinese compared to American users, their engagement reduced if the system took some time to respond. Though, we found Chinese and Americans have differences in expressing engagement, we still found a set of behaviors that are statistically different among high engagement turns and low engagement turns in both cultures with the same trend. In the next chapter, we will use these behaviors to model engagement. The different results among different cultural contexts also suggest that we need to model engagement differently for different users with different cultural backgrounds.

6.1 Introduction and Related Work

Many researchers cared about engagement when it comes to designing interactive systems. Some think engagement is correlated with immersiveness [Lombard and Ditton, 1997], for example how immersed you are in the interaction plays a key role in evaluating the interaction quality. Some think engagement is related to the level of psychological presence (e.g. focus) during a certain task [Abadi et al., 2013], for example how long the user are attracted to the robot [Moshkina et al., 2014]. Some define engagement as the value a participant in an interaction attributes to the goal of being together with the other participant(s) and of continuing the interaction [Peters et al., 2005]. They also think engagement is closely linked with interest, which in turn causes attention. Some care about the process, [Sidner et al., 2004] defined engagement as "the process by which two (or more) participants establish, maintain and end their perceived connection during interactions they jointly undertake". In this thesis, we investigate social engagement in non-task-oriented conversations. Another difficulty is to isolate users from the social context or surroundings completely and only focus on the conversation itself. Thus we try to conduct the experiments in a controlled lab environment.

[Gatica-Perez, 2009] define another similar construct "interest" as "people's internal state related to the degree of engagement displayed, consciously or not, during social interaction. Such displayed engagement can be the result of many factors, ranging from interest in a conversation, attraction to the interlocutor(s), and social rapport." Another similar concept that has been investigated extensively is "involvement", which is close to engagement but usually targets specifically for analyzing conversations that involve more than two participants (multiparty conversations). The complexity of analysis multiplied when more participants are involved as there will be multiple possible addressees to attend to for the speaker in multiparty conversations. In this thesis, we focus on conversations that only involves two participants (dyadic conversations), but we believe the findings we discover in dyadic conversations will also shed lights on analyzing multiparty conversations.

Many work attempted to quantify the presence or the amount of engagement the user has in an interaction. [Bohus and Horvitz, 2009] focused on quantifying engagement as whether or not a user intends to engage in an interaction with a system. [Levitski et al., 2012] operationalize engagement by whether the participant is actively gaze at another participant. [Bonin et al., 2012] used human behaviors to define involvement. They think involvement can be expressed through means such as mimicry, gesture or speech character. [Oertel et al., 2010] quantified involvement through a detailed coding scheme with a Likert scale from one to ten. Each level has a very detailed description related to the person's contribution to the group interaction. However, none of these work analyzed engagement across different culture groups, let alone computationally model them.

Many studies found that people from different cultures behave differently during conversations. The CUBE-G project is one of the most extensive data-driven efforts to study German and Japanese culture comparatively. [Rehm et al., 2009] collected a cross-cultural multimodal corpus of dyadic interactions and found in most Japanese conversations, participants discussed the experimental setting while German subjects talked significantly more about social topics such as their studies or friends. [Khaled et al., 2006] focused on cultural differences in persuasion strategies found that for short-term oriented cultures a stronger focus on the task itself can be expected, whereas for long-term oriented cultures a slower and more exhaustive way of problem solving can be expected, where every opinion is taken into account and harmony is at stake resulting in an increased frequency of contributions that are related to the communication management. [Leffler et al., 1982]. suggested there are differences in spatial behavior and in use of verbal facilitators like “yeah” or “mhmm” of people in different cultures when they are interacting with people in higher statuses. [Matsumoto, 2006] found that people from Arab cultures gaze much longer and more directly than Americans. In general, collective cultures, such as Arabian culture engage in more gazing and have more direct orientation when interacting with others.

Previous studies found that people in different cultures behave differently towards a task-oriented virtual agent as well. In a direction giving task, Arabic and English native speakers interact with a virtual agent differently [Gedawy et al., 2012]. English natives had a higher frequency of using cardinals, pauses and intermediate information while Arab natives used units of distance, left/right turns and error corrections more frequently than English natives. Another comparison study suggested that Arabians trust more on the Arabian speaking robot who speaks language that is rhetorically well formulated, while the rhetorically factor is less important for Americans when they talk to an English speaking robot [Andrist et al., 2015]. These interesting study between different cultures motivated us to explore if there is cultural differences between different cultural groups in expressing engagement when users are interacting with a social chatbot.

6.2 Experimental Setting

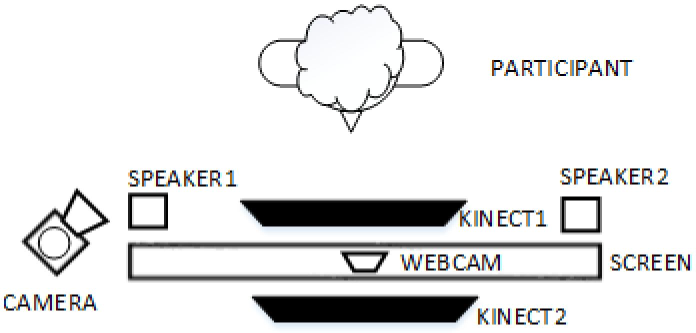


Figure 6.1: TickTock experiment setting

To test our hypothesis, we invited participants to our lab to talk to TickTock with multimodal output and input face to face. The experiment involves participants freely interacting with TickTock more than five minutes, one at a time. The setting is shown in Figure 6.1. The participant stands in front of the system, with the talking head displayed on a screen suspended at the participant's eye-level. A separate video recording is also made, and used as the prompt for the engagement annotation task. The system collects different channels of speech data, including the segmented utterances, the corresponding Google ASR output, the TTS output, as well as timing information. For visual data, we use CML-Z [Baltrusaitis et al., 2012] head tracker to collect 66 facial features, face orientation and location.

6.3 Databases

We describe two databases: English TickTock Database (ETDb) and Chinese TickTock Database (CTDb). ETDb is a collection of conversations between a human and TickTock speaking in English language, while CTDb is conversations between a human and TickTock in Mandarin Chinese. All the conversations are non-task-oriented that participants discuss everyday topics such as movies, music, etc. All the datasets are public available upon request.

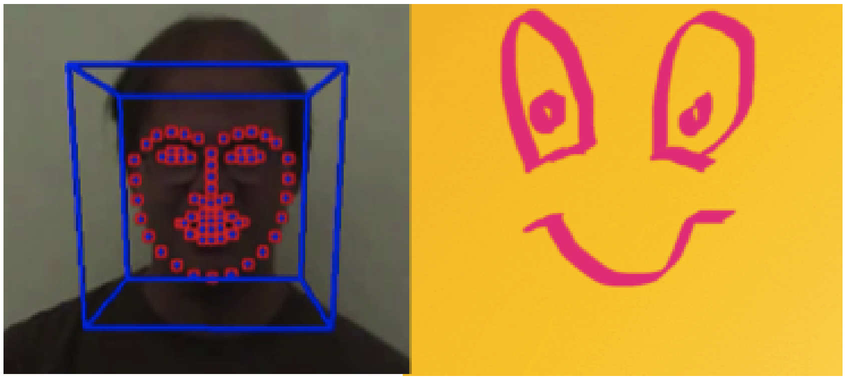


Figure 6.2: Screen shot of example conversation recording in the English TickTock Database (ETDb)

6.3.1 English TickTock Database (ETDb)

The English TickTock Database (ETDb) is an audiovisual database containing conversations of people interacting with the baseline version of TickTock face to face in English language. We collected 24 interactions and each conversation is about five minutes long. Subjects are students and faculty members in the United States. The database include both genders (6 females, 18 males) and varied ethnicity (11 American, 5 Chinese, 5 Indian, 1 British, 1 Korean and 1 Singaporean). None of the participant has interacted with TickTock before, however, they have varied familiarity with dialog systems in general, which may influence their behaviors when

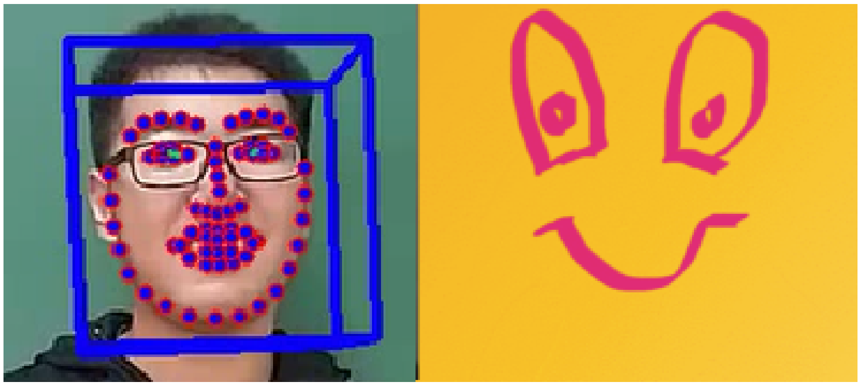


Figure 6.3: Screen shot of example conversation recording in the Chinese TickTock Database (CTDb)

interacting with the system. A total of 881 conversational turns are labeled with engagement by experts. In addition, we manually annotated behaviors, such as smiles and head nods. See Figure 6.2 for an example data snapshot. Table 6.1 shows an example conversation.

Participant	TickTock
Your noise is a question mark, I think that's interesting.	Of course, what's your idea?
I think it fits you perfectly.	Let me see, his nose is still broke.
Do you want me to make your nose broken?	Ah, yes, why did you ask me for this?
Because I thought you might mind.	Well, I am not getting into specifics about what doctor treated him, how he was treated.
You have not been treated yet. That is in the future.	Let me see, by the way, are you interested in movies?
Yes, I watch movies sometimes.	Great.

Table 6.1: An example conversation between a human subject and TickTock in English

6.3.2 Chinese TickTock Database (CTDb)

We used the TickTock-Mandarin version (see Chapter 4 for details) to collect the data. We recruited 21 people (12 males) in total in Beijing China to conduct the study. All participants are university students who were born and raised in Mainland China. No participants interacted with a virtual agent before and some have experience with speech activated dialog systems. There are 353 conversational turns labeled with user engagement levels by experts in this dataset. In addition, we manually labeled smiles in the dataset . See Figure 6.2 for an example data snapshot. An example conversation between the Mandarin version of TickTock and the user is shown in Table 6.2.

Partner	Utterance (Translation)
User:	那要不你给我唱首歌吧。(Then let me sing you a song)
TickTock:	让我想一下，就没有浪漫一点的反应啊？(Let me see, don't you have something romantic to say?)
User:	什么鬼啊? 我感觉受不了这个机器了。(What the heck, I feel I can't take the system anymore.)
TickTock:	你到底有没有爱过我？(Have you loved me at all?)
User:	什么情况？(What is the situation here?)
TickTock:	让我想一下，好吧。(Let me see, OK.)
User:	好吧。(OK.)
TickTock:	你有什么想说的吗？(Do you have anything to say?)
User:	爱过。([I] have loved you.)

Table 6.2: An example conversation between a human subject and TickTock in Mandarin

6.4 Annotation Scheme

We annotated the data for user engagement. Engagement is defined as "the **interest** to contribute, to continue the conversation" [Peters et al., 2005].

Level	Definition	Description
1	Strongly Disengaged	Shows no interest in the dialogue, engaged in other things.
2	Disengaged	Shows little interest to continue the conversation, passively interacts with his conversational partner.
3	Neither Disengaged nor Engaged	Interacts with the conversational partner, showing neither interest nor lack of interest to continue the conversation.
4	Engaged	Shows mild interest to continue the conversation.
5	Strongly Engaged	Shows a lot of interest to continue the conversation and actively contributes to the conversation.

Table 6.3: Engagement annotation scheme

Instead of using a fixed length of time, like most of the previous work [Gatica-Perez et al., 2005], we chose *conversational exchange* as the annotation unit. We define conversation exchange as "the start of the dialog system's utterance until the next time the system starts speaking again after another conversational partner has spoken." Conversational exchanges capture the micro-dynamics of discourse and are also the minimum planning point for the dialog manager. While for annotation scales, [Gatica-Perez et al., 2005] used a 1-5 scale without detailed description of each level, but a relative distinction of high and low and [Bonin et al., 2012] defined some cues for annotating involvement, such as involvement can be expressed through such means of mimicry, gesture and speech characteristics, besides the broad definition of engagement. We used a 5-point scale with detailed definition (see Table 6.3 for details).

Engagement, by its very nature, is difficult to annotate, as it reflects an internal state of the participant. Most of the previous work used third person annotations and [Oertel and Salvi, 2013] used self-reported ranking scores. In our work, in order to establish a baseline standard, we ask participants to watch the video recording of their sessions and mark the level of engagement for each conversational exchange. To assess whether another person can reliably judge engagement, we have a second expert annotator annotate the recording. There are two expert annotators who annotated both databases for engagement, and the inter-annotator agreement (kappa) is 0.93 and 0.74 in the ETDb and CTDb respectfully after collapsing the 5-point scale to 2-point scale (1,2 is disengagement, 3-5 is engagement). Experts can only rate data from their own culture, as people with different culture backgrounds may perceive engagement differently as well.

We run a Spearman's rho correlation analysis on self-reported ratings and expert ratings on seven interactions that we randomly sampled from the database and found there is idiosyncrasy in self-reported ratings. However, after normalizing ratings per rater, there is no statistical significant difference between these two sets of ratings over these seven interactions (t(223)=0.85, p=0.20). Therefore, for consistency, we used the expert annotations for all databases in our behavior analysis and in the machine learning training process in the coming chapter as well.

6.5 Human Behavior Quantification

Our goal of analyzing these behaviors in two different cultural settings is twofold: 1) to test if people from different cultures act differently when talking to a virtual chatbot; and 2) to identify differentiating behavior cues to predict engagement. Thus, we focus on observable behaviors mentioned in previous literature that correlate with engagement and are quantifiable in real time using automatically extractable audiovisual features. In this section, we denote automatically extracted features *Auto* and manually annotated features *Manual* and we quantify them per conversational exchange (turn).

We first investigated a list of multimodal behavioral cues that were used in modeling engagement in human-human communication introduced by [Gatica-Perez et al., 2005]. Most of them can be quantified using multimodal sensing techniques. This is also a step towards automatically detecting engagement using human behaviors in the next chapter. We grouped behaviors into four categories according to their income signal sources: verbal, acoustic, visual and dialog.

6.5.1 Verbal Behavioural Cues

Word count: is calculated using results from the automatic speech recognitions (ASR), as well as manual transcriptions. The word error rate is 34% and 39% in English and Chinese respectively.
Speech rate is calculated as the number of words spoken per second in the American data set and number of Chinese character spoken per second in the Chinese data set. They were calculated both over ASR results.

6.5.2 Acoustic Behavioural Cues

We performed both echo cancellations and noise suppression during audio collection. We used the voice detection in Sphinx 4 [Walker et al., 2004] to detect the start and end of the utterance. Then we detected the pause bounded units (PBU) automatically with a heuristic algorithm. It used a silence threshold of 30.0dB to calculate the average length of the phrases identified. If a phrase have an average length of more than two seconds, we increase our silence threshold by +3.0 dB and re-extract all the prosodic features, continuing to do so until the average phrase length is below two seconds in length. The two-second threshold is calculated using manual segmentation results of the four interactions in the data set.
Intensity mean and **Intensity variance** are computed over the independent pause bounded units (IPU) that separated by PBU for each turn.
Pitch mean and **Pitch variance** are calculated using the Sphinx4's pitch extractor over the IPUs [Walker et al., 2004].

6.5.3 Visual Behavioural Cues

Smile is quantified with manual annotations as well as an automatic detector. We manually annotated smiles in both data sets. In our annotation scheme, we defined a smile as "a facial expression formed by flexing the muscles near both ends of the mouth and by flexing muscles throughout the mouth, ignoring the true and fake smile distinction" [Yu et al., 2013a]. We only asked the annotator to judge whether a smile is present or not in a turn, not to annotate the specific start and end of the smile. Two annotators reached an inter-annotator agreement of 0.90 in kappa. Based on the annotated data, we trained a smile detector using a simple Support Vector Machine [Chang and Lin, 2011] with features from a face tracker, CML-Z [Impett et al., 2014]. The tracker produced the facial unit activation and it is believed that smiles are correlated with action unit 6 and 12. We trained the smile detector based on these two features mean and variance. We have a training set of 100 samples, in which the majority vote is 30%. The smile detector achieved 0.82 in accuracy in a five-fold cross validation setting on the annotated data set.

Laughter is defined as the state when the subject is smiling, not talking, but is vocalizing. We automatically quantify the raw features of laughter by finding smiles that don't occur with speech and have an audio intensity higher than 50dB. The model achieves 0.86 in precision.

6.5.4 Dialog Behavioural Cues

System Interruption is computed as a binary number that captures if the participant is interrupted by the system or not.
User Interruption is calculated in the same fashion which captures whether the agent is interrupted by the participant or not.
Human Response Time is calculated as the time it takes the user to respond to the system.
System Response Time is calculated as the time it takes the agent to respond to the user.
Time Since Start is calculated as the time elapsed since the conversation starts till the current turn starts.

6.6 Analysis and Results

In this section, we describe how human behaviors relate to user engagement with respect to different culture groups. We only include the users that are from North American culture in ETDb (11 users) for analysis in this section, resulting 349 exchanges (turns). In the Chinese database, we have 21 interactions and in total 353 exchanges (turns). We separate the conversation exchanges by their engagement annotations into two group, high engagement (1-2) and low engagement (3-5). We perform a two tail Student t-test on the two group with respect to each features. To deal with the multiple statistical tests problem, we applied Bonferroni correction (simply requiring the p value to be smaller than 0.05/N, here N is the number of tests, in this section, N=32). We report all the statistical significance test's p value in Table 6.4 in groups based on different modalities. One thing to notice is that conversational exchanges within one interaction may have correlations, thus the statistics we report may be influenced by that.

We found that in both culture groups, the engaged conversation exchanges have a larger number of words compared to disengaged conversation exchanges. Therefore, we can say the more the user speaks, the more engaged the user is in both cultural contexts. Interestingly, the raw features calculated from ASR output are even more predictive of engagement than features calculated from manual transcriptions. The ASR results sometimes miss portions of a user's utterance, which resulted in smaller value for average number of words per turn than the value calculated using manual transcriptions. In addition, ASR does not account for disfluencies such

Features	American	Chinese
Verbal Behavioural Cues		
Word Count (Manual)	0.002 (+)	**0.001** (+)
Word Count (ASR)	**0.001** (+)	**0.001** (+)
Speech Rate (ASR) Mean	0.04 (+)	0.10 (+)
Speech Rate Variance (ASR)	0.07 (+)	0.09 (+)
Acoustic Behavioural Cues		
Intensity Mean	**0.001** (+)	0.002 (+)
Intensity Variance	**0.001** (+)	**0.001** (+)
Pitch Mean	0.08 (+)	0.10 (+)
Pitch Variance	0.20 (+)	0.32 (+)
Visual Behavioural Cues		
Smile(Manual, Freq.)	**0.001** (+)	0.05 (+)
Smile(Auto,Freq.)	**0.001** (+)	0.05 (+)
Laughter (Freq.)	**0.001** (+)	**0.001** (+)
Dialog Behavioural Cues		
System Interruption	**0.001** (-)	0.09 (-)
User Interruption	0.12 (-)	0.11 (-)
Human Response Time	**0.001** (-)	0.24 (-)
System Response Time	0.01 (-)	**0.001** (-)
Time Since Start	**0.001** (-)	**0.001** (-)

Table 6.4: The difference between high engagement and low engagement exchanges with respect to different features in different culture groups. Results with statistically significant ($p < 0.0016$) are indicated by bold fonts. The positive sign indicates that exchanges with high engagement have larger values in this feature, while the negative sign indicates the opposite.

as “umm” and “ahh”. The appropriateness of the system response is highly dependent on the ASR results. The reason ASR results are more differentiable of engagement from disengagement than manual transcription is that the incomplete user information leads to shorter system perceived user utterances, which in turn makes the system unable to return meaningful content to engage users. Thus, the shorter word count calculated from ASR correlates to less engaged turns.

We also found that in both cultures, engaged turns have louder and more varied speech volume. However, we didn’t find significant difference in terms of pitch mean or pitch variance between engaged exchanges and disengaged exchanges in both culture settings. This finding differs from [Voigt et al., 2014] which found that pitch mean is positively correlated with engagement. This may be due to the fact that their conversations are monologues, which are different from ours. We also suspect this is caused by the background noise in the collected speech. As in the experiment, to make the user feel relaxed, instead of a close talk microphone, we placed a Microsoft Kinect in front of the user. In the future experiment, we wish to reduce the room noise and use a close-talk microphone to collect cleaner speech.

We conducted a qualitative analysis of the semantic contents of user responses and found 4 out of 9 Chinese female and 2 out of 12 Chinese male participants told the agent: “I like you, you are so cute.”, while none of the Americans expressed their likability towards the agent directly. We further looked into the smile features, and found that Chinese female users smile more frequently than the other users with statistical significance ($p < 0.03$). One possible explanation of the difference is that Chinese culture is a typical collective culture that seeks harmony between partners. Chinese may subconsciously treat the agent as one of their partner and try to harmonize the agent by using more positive affect. In addition, Chinese female were influenced by the culture to express more positive affect as a cultural norm. While American culture is a individualism culture according to Hofstede’s culture theory [Hofstede and Hofstede, 2001]. Americans would change less of their behaviors to harmonize their partners compared to Chinese.

Another difference we found between the two cultures is in terms of the system response time. There is a clear trend among Chinese users, the system response time is longer among conversational exchanges are disengaged. While, there is no difference among engaged and disengaged exchanges among American users. One explanation of such difference is that Americans are mostly individualists who tolerate the delay much more than Chinese who seek for harmonized dynamics in conversations all the time according to the Hofstede culture model. As Chinese is a more collective culture, a long pause from their partners makes Chinese participants less engaged.

6.7 Conclusion

To conclude, we described methods to automatically extract human behaviors in different modalities. People from different cultures express engagement differently. For example, compared to Americans, among Chinese users, the system response time is longer among conversational exchanges are disengaged. However, regardless of whether the user is from North American culture or Chinese culture, they speak more, speak louder and laugh more frequently in conversation exchanges that they appear more engaged. We will describe the role of these features in predicting engagement in the next chapter in details.

Chapter 7

Engagement Prediction

After identified a set of multimodal behavior features that have similar correlation trend with engagement regardless of users' culture background, we introduce a computational model that leveraged these features to predict user engagement in real-time systems. We specifically talked about the trade-off of predicting the future and prediction accuracy in real-time system. To facilitate modeling of computational models for cultures with litter labeled audio-video data, such as Chinese, we adopted a transfer learning method .

7.1 Introduction and Related Work

Different multimodal features and machine learning algorithms have been used to predict engagement or involvement in human-human conversations. [Gatica-Perez et al., 2005] and [Hsiao et al., 2012] both used HMMs (supervised machine learning methods) with audio features, such as intensity, fundamental frequency and speech rate, etc. [Gatica-Perez et al., 2005] combined audio features with some visual features such as hand blobs. [Bednarik et al., 2012] and [Oertel and Salvi, 2013] both used gaze features obtained by an eye tracker to model engagement and involvement in multiparty conversations. [Bednarik et al., 2012] used linear kernel SVMs and [Oertel and Salvi, 2013] used GMMs in modeling engagement.

[Bohus and Horvitz, 2009] used both visual and acoustic signals to predict subjects' engagement when interacting with a virtual agent in open space. Benkaouar and Dominique [Benkaouar and Vaufreydaz, 2012] also incorporate a user engagement detector that takes input from multiple sensors, in a robot companion design. However both of them represent engagement as a binary variable, 'engaged' or 'disengaged' and both systems are task based. Based on our analysis on culture difference in expressing engagement from last chapter, we selected predictive multimodal features and built a real time computational engagement model. The model can assess engagement at each dialogue turn and also predict the level of engagement at the following turn, both on a five point Likert scale. Table 8.3 in Chapter 8 showed the annotation scheme for engagement. We also encoded one turn interaction history to improve the prediction performance via adding features that represent the change between the current and previous turns with respect to selected feature groups.

7.2 Machine Learning Setting

All the mentioned related work formulated the computational engagement model as a binary classification problem. However, engagement is a user's mental state, which is a continues value that changes over time, modeling engagement as a discrete value is not appropriate. We modeled engagement computationally using regression models, in order to model engagement as continuous value. During the training process of the regression model, we used leave-one-interaction-out cross-validation to prevent the same participant appearing in both training and testing sets. Cross-validation also ensures the model's robustness towards distribution difference between the training and testing data. To further ensure the model's robustness, we also scaled each data point to zero mean and unit variance.

As mentioned in the last chapter, we quantified multimodal features per conversational exchange. We used early fusion to fuse these features together, which means concatenating features extracted from a conversational exchange into a vector. We used a support vector regression algorithm [Chang and Lin, 2011] to model user's engagement per conversational exchange. In the modeling process, we used a linear kernel instead of non-linear kernels in SVR to reduce the training and testing time, since our model needs to run in real time. The linear kernel also avoids overfitting, since we have relatively little data (e.g. 881 in ETDb). To evaluate the model's performance, we used correlation coefficient, a common evaluation metric for regression. In our setting, the correlation coefficient stands for the correlation between the predicted engagement value and the expert annotated label. .

7.3 Feature Sets

We experimented two sets of features in modeling user engagement in ETDb, one set with automatic features that correlate with engagement across both American and Chinese cultures in similar significant trend, the other set with automatic features selected through a stepwise regression method [Draper and Smith, 2014]. We named the first set of features as the cultural independence features (CIF), and the send set of features as the stepwise regression output features (SRF).

CIF consists of “word count (ASR)”, ”intensity variance”, “laughter (Auto, Freq)”, “time since start”, based on the results reported in Table 8.4. SRF consists of “word count (ASR)”, “intensity mean”, “speech rate variance”, “laughter (Auto, Freq)”, “system interruption” and “time since start”. These features are selected using a stepwise regression with the seeding feature as “word count (ASR)” as it is the non-sparse feature with the highest correlation in ETDb dataset. We constructed a baseline feature set which included all the automatically obtainable features described in Table 8.4, we named this complete set as “ALL”. We also constructed a set of modality-based feature groups, for example: “Acoustic” stands for all features from audio modality: mean and variance of power and pitch; “Visual” stands for all features from vision modality: smile (Auto, Freq), laugh (Auto, Freq) and head orientation; and “Verbal” stands for all features from text and dialog modality: mean and variance of word count and speech rate, system interruption, user interruption, human response time, system response time and time since start. The reason to combine them is that text and dialog features combined represents verbal communication cues. We also enumerate all the possible combination of these modalities to form different features, such as “Verbal + Acoustic” stands for automatically obtained features in both verbal and acoustic modalities.

Because conversation is a Markov process, the current conversation exchange’s engagement depends on its history. Thus we encoded one turn interaction history to improve the prediction performance via adding features that represent the change between the current and previous turns with respect to selected feature groups.

7.4 Results and Analysis

Table 7.1 shows the results of the SVR model with different feature groups on ETDb dataset. We found adding one turn of history improved the performance of the model with statistical significance for feature groups: CIF and SRF. We also tested on adding more than one turn history, and the results didn’t improve with a statistical significance.

Results on feature groups with different modality combinations shown that among all the modalities, text and dialog modality has the best performance in single modality model, and the modality leads to the second best performance alone is vision. Though the features in visual modality are very sparse, they are very indicative. Both the dialog and audio modalities produced low correlations alone in the model. One possible reason is that features, such as “pitch mean” introduced noise and confused the model. If we combine different modalities together, in general

Feature Group	Without History	With History
CIF	0.37	0.32
SRF	**0.35**	**0.30**
Verbal	0.39	0.38
Acoustic	0.45	0.42
Visual	0.38	0.37
Verbal+Acoustic	0.39	0.38
Verbal+Visual	0.38	0.37
Acoustic+Visual	0.39	0.37
ALL	0.38	0.36

Table 7.1: Engagement prediction performance respect to different feature sets on ETDb dataset

there is a small improvement over individual modality, however none of them are statistically significant. The possible reason is that each modality may have some features that are noisy and involving them would confuse the computational model. So combining modalities would still have the noise. Only when the features are clean and have complimentary information would combining them have improved the performance. Among all the feature combinations, the SRF feature group performed the best in both the condition of including or not including history information. This indicates feature selection methods, which leveraged the original dataset's distribution is useful. Even we only selected the features that have similar trends across user cultures, the performance is still better than including noisy features. The results implied that we should always perform feature selection in building computational methods for audiovisual tasks.

7.5 Time and Accuracy Trade-Off

In real-time systems, turn taking is critical. In human-huamn conversations, the time for one to respond to each other's speech only takes less then two second. For machines, within this period, it needs to perform a sequence of computations. First it needs to perform end-pointing detection, to realize if the user actually finished the utterance or is just pausing, this usually takes 500ms. After knowing the end of the speech, the automatic speech recognizer then pass the decoded user utterance to the language understanding component and the situation awareness component of the dialog systems. Simultaneously the visual and acoustic sensing component will send the information to the situation awareness component. With in the situation awareness component we will normalize and compute the statistics of these row features and combine them for the SVR model. Then based on the natural language understanding results and the situation awareness results, the system will generate an action based on the dialog policy. The action generation process usually takes longer than we hoped. One bottle neck is one of the system response generation method, information retrieval takes a long time to find the closest mapped sentence in a huge database with millions of entries. The same problem is true for other response generation methods such as sequence-to-sequence model. In the proposed TickTock architecture, one possible solution is to predict user's engagement ahead of time, so the policy could take that into account, and some cases, given the value of the user engagement, the system action will be certain conversational strategies instead of the generated response, thus there is no need to go through the time-consuming response generation process.

A lot of computation is needed in a short time window in order to minimize the turn taking latency and allocate time for planning. Therefore, we need to predict user engagement in advance to save time to form dialog action. We aim to build our engagement detector in a real time system, which requires producing the engagement score in real time as well, so we can reduce the latency of responding to users. In addition, the features can be extracted incrementally over time, we discuss the quantification window for each feature that are selected using the step-wise regressor.

For verbal behavioral cues, the *count of words* cannot be computed until users finish their utterances but *speech rate* can be computed mid utterance, given a discrete time window. For acoustic behavioural cues, *intensity mean* and *intensity variance* can also be computed mid utterance. For visual behavioural cues, *laughter*, *head shake* are both based on a trained predictor. They can both be computed mid utterance as well. For dialog behavioural cues, *turn length* is not possible to compute before users finish their utterances, but *time since start* can be computed very easily in mid utterance. In conclusion, *intensity mean*, *speech rate*, *laughter*, *head shake* and *turn length* are features that can be computed incrementally in real-time systems.

In the real-time implementation, each feature can be extracted with a separate thread, so there will be no delay in computation. However, we need to compute the statistics over these features, which requires waiting until the users finish their utterances. The speech end-point detector takes an additional 200-500ms (a parameter that can be tuned for different types of dialog systems) to ensure the user actually finished the utterance, rather than having a long pause in their speech. We take advantage of this additional period of time to ensure the result of the engagement predictor is ready right after the wait.

Each feature has its own sampling rate. For example, *laughter* takes the raw face tracking features in a 30-frame-per-second frame rate, while *intensity* is computed over a sampling rate of 32KHz. We use a unified time window of 200ms to compute statistics over these raw features. This window size is empirically decided to balance the computation load and the performance. We show the engagement predictor's performance over time in Figure 7.1. Predicting user's engagement every 200ms enables the dialog system to reduce the computation overhead and makes sure the system has the engagement predictor output right after the system gets the signal from the speech end-point detector that indicates that the utterance is finished. In Figure 7.1, we can see in the first 200ms, the engagement predictor's performance is not high, because there is very limited information in the first 200ms of the user's utterance. However the performance improves drastically when it reaches 1500ms. The reason is that most of the user utterances are between 1-2 seconds and important features such as "count of words" and "turn length" require the time window to reach the end of the turn to be accurate. The engagement predictor's performance converges once the time window covers the information of all the user utterances.

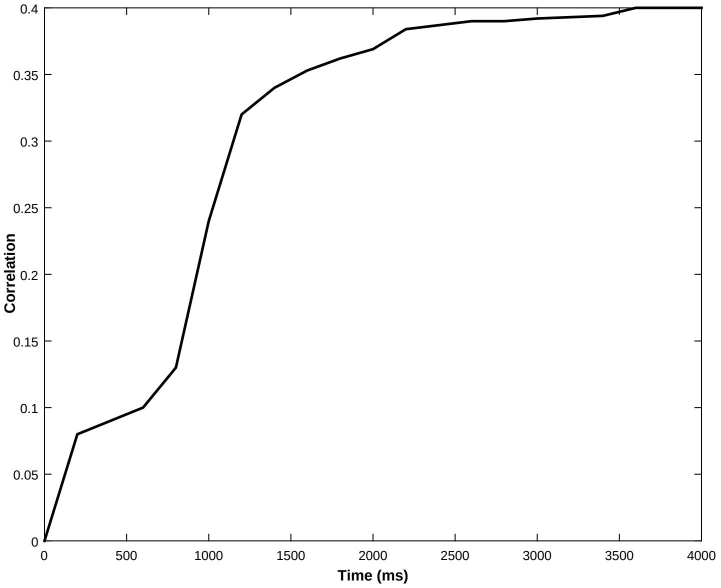


Figure 7.1: Engagement prediction performance over time

7.6 Culture Adaptation

There are very few datasets available for multimodal conversations with dialog systems. Most of the dialog systems available are designed for English speaking users, there is not enough data to build a model for other cultures. Thus to leverage the data from well studied culture to less studied cultures is a very important research question.

In the previous chapter, we found that users in different cultural contexts behave differently in expressing engagement. For example, similes are significantly correlated with engagement in American culture but less significantly in Chinese culture. Thus, the behavior feature distributions are different between the two cultures. We also found that using the model trained on American culture to predict engagement of Chinese users lead to bad performance, which is 0.504 in rmse, while the majority vote baseline is 0.547 (in this dataset, the majority value is 3 in terms of ground truth labeling, I used that as the constant prediction value for the baseline).

We propose to borrow the data from a well studied culture to improve user model performance for the less studied culture. We model the problem as a conventional mathematical problem, which is to leverage the source data (American cultural data) together with target data (Chinese cultural data) to build a model for the target domain (Chinese). We use an existing domain adaptation method, "frustratingly easy domain adaptation", which focuses on feature projections [Daumé III, 2009]. Essentially, the algorithm takes each feature in the original domain and makes three versions of it: a general version, a source-specific version and a target-specific version. The augmented source data will contain only general and source-specific versions. The augmented target data contains general and target-specific versions. We represent the data from the targeted domain as X_t and the data from the source domain as X_s, and the method simply transformed the targeted domain data into $< X_t, X_t, 0 >$ and the source data into $< X_s, 0, X_s >$. We use the set of features that achieved the best performance in Table 7.2 for all the models mentioned in this section. We use the v-Support Vector Regression for all the experiments. We found the domain adaptation method, which has projected features before regression reached the best performance in rmse (0.452), which is statistically better than the model that used the Chinese data alone (rmse = 0.486) (see Table 7.2 for details). A simple domain adaptive model captured the common information of the two data sets and accounted for the differences of the two data sets. The domain adaptive model makes the information additive in the source and the target domain.

Training Data	rmse
American Data	0.504
Chinese Data	0.486
American and Chinese Data No Adaptation	0.490
American and Chinese Data with Adaptation	**0.452**
Majority Vote	0.547

Table 7.2: Engagement prediction performance on the Chinese data set. The smaller the rmse is, the better the performance is.

7.7 Conclusion

First we used a step-wised regression to select a set of features for engagement prediction. Then using these features, we developed machine learning models for assessing the user engagement of the current conversation turn and for predicting the user engagement of the future turn via regression. In addition, we also analysed each modalities contribution to the overall performance. We also described some adjustments of the previous models to make them usable in real-time systems. Finally we also proved that using domain adaptation methods that borrows data from well studied culture improves the performance of the engagement model for a less studied culture that has limited data available.

Chapter 8

Conversation Strategy for Non-Task-Oriented Systems

I designed a set of conversational strategies to cooperate with situational context, to address the difficulties for language understanding and personalization.

8.1 Introduction

Researchers developed a variety of methods to generate responses for non-task-oriented systems, such as machine translation [Ritter et al., 2011], retrieval-based response selection [Banchs and Li, 2012], and sequence-to-sequence recurrent neural network [Vinyals and Le, 2015]. All of them aim to improve system coherence, but none of them focuses on the experience of the user. Conversation is an interaction that involves two parties, only improving the system side of the conversation is insufficient. In an extreme case, if the system always behave appropriately, but is a boring and passive conversational partner, users would not stay in the conversation or come back a second time. We propose to leverage the interactive property of dialog systems in the system. We approach it in three directions, through conversation strategies to cooperate with user's situation and, to resolve language understanding difficulties collaboratively with users and to provide personal experience.

As mentioned before, in this thesis, I focus on one type of situation context, user engagement. I will talk about how to cooperate with user engagement via conversational strategies designed based on communication literature. It is believed user engagement is a critical to maintain the conversation flow. In a non-task-oriented system, the user experience is crucial, because the ultimate goal is to keep users in the interaction as long as possible, and have them come back as frequently as possible. Previously systems have not worked on improving user experience, mostly because their systems are text-based, and cannot access to the user's behavior beyond text and user experience is better sensed with multimodal behavior information. I define user engagement as the **interest** to contribute, to continue the conversation in each turn as I mentioned in Chapter 5. Making the system aware of the user's engagement is considered crucial in creating user stickiness in interaction designs. As better user engagement leads to a better experience, and attracts repeat users. I argue that a good system should not only be coherent and appropriate but also engaging.

8.2 Related Work

Many experiments have shown that an agent that reacts to its user's behavior or internal state leads to better user experience. In an in-car navigation setting, a system that reacts to the user's cognitive load was shown to have better user experience [Kousidis et al., 2014a]. In a tutoring setting, a system that reacts to its user's disengagement resulted in better learning gain [Forbes-Riley and Litman, 2012]. In task-oriented systems, users have a concrete reason to interact with the system. However, in a non-task-oriented setting, user engagement is the sole reason for the user to stay in the conversation, making it an ideal situation for engagement study. So I focus on making the system reactive to user engagement in real time in an everyday chatting setting.

In human-human conversations, engagement has been studied extensively. Engagement is considered important in designing interactive systems. Some believe engagement is correlated with immersiveness [Lombard and Ditton, 1997]. For example, how immersed a user is in the interaction plays a key role in measuring the interaction quality. Some believe engagement is related to the level of psychological presence (i.e. focus) during a certain task [Abadi et al., 2013], for example how long the user is focused on the robot [Moshkina et al., 2014]. Some define engagement as "the value a participant in an interaction attributes to the goal of being together with the other participant(s) and of continuing the interaction" [Peters et al., 2005]. I define engagement as the interest to contribute, to continue the conversation. Because the goal of a non-task-oriented system is to keep the user interacting with the system voluntarily, making users to stay in the conversation is critical. I use a 5-point Likert scale for engagement annotation, higher score indicates higher engagement. I use three categories to annotate system appropriateness: "Appropriate", "Interpretable" and "Inappropriate". The engagement annotation scheme is introduced in chapter 6 and the system appropriateness annotation scheme can be found in chapter 5.

A lot of conversational strategies have been proposed in previous work to avoid generating incoherent utterances in non-task-oriented conversations, such as introducing topics, (e.g. "Let's talk about favorite foods!" in [Higashinaka et al., 2014]) and asking the user to explain out-of-vocabulary words [Maria Schmidt and Waibel, 2015]. I propose a list of active participation strategies focusing on handling user disengagement and in the same time avoiding possible system breakdowns. The probability of possible breakdowns can be calculated using different metrics according to different systems. For example, a neural network generation system [Vinyals and Le, 2015] can use the posterior probability to decide if the generated utterance would cause a system breakdown or not.

Previously, knowledge base was used to automatically induct semantic slots to facilitate language understanding in task-oriented dialogs [Chen et al.,], while our work is the first to use knowledge bases to synthesize conversation strategies (e.g. grounding) that can improve language understanding collaboratively with users through interactions. Our work on personalized strategies leveraged the information from dynamically constructed personalized knowledge bases to provides a personalized experience for each user .

8.3 Conversation Strategies

I describe two types of conversation strategies, active participation strategies, knowledge-base strategies. With in the second category, I have two sub-groups: grounding strategies and personalized strategies.

8.3.1 Active Participation Strategy

I designed six active participation strategies to maintain and to improve user's engagement based on communication literature. All the strategies were derived from the concept of active participation introduced in [Wendler, 2014]. Active participation actions tend to lead the conversation and make partners easy to respond to. These six strategies also have effect in controlling conversation flow. The first three strategies change the current topic and the last three strategies keep users in the same topic.

1. **Switch a topic (Switch).** The chatbot proposes a new topic, such as "sports", other than the current topic. For example: "Let's talk about sports." If this strategy is executed, the system updates the tracked topic.
2. **End topics with an open question (Open).** The chatbot closes the current conversation topic and asks an open question, such as " Sorry I don't know. Could you tell me something interesting?".

3. **Refer back to a previously engaged topic (Refer back).** This strategy refers back to the previous engaging topic. I keep a list of utterances that have resulted in high user engagement in a personalized knowledge base. This strategy will refer the user back to the most recently engaged turn. For example: "Previously, you said 'I like music', do you want to talk more about that?" This strategy is also considered to be personalized stratey as it leverages the knowledge in the personalized database. This strategy is only available if the system could track users engagement, as it needs to store user's engagement history.
4. **Initiate topic related activities (Initiation).** The chatbot invites the user to an activity. Each invitation is designed to match the conversation topic. For example, under the politics topic, the system would ask: "Do you want to see the latest Star Wars movie together?"
5. **Tell a joke (Joke).** The chatbot tells a joke under the conversation topic, such as: "Politicians and diapers have one thing in common. They should both be changed regularly, and for the same reason".
6. **Elicit more information (More).** The chatbot asks the user to say more about the current topic, such as " Could we talk more about that?".

Each strategy has a set of surface forms to choose from in order to avoid repetition. For example, the Switch strategy has several forms, such as, "How about we talk about sports?" and "Let's talk about sports."

8.3.2 Grounding Strategy

I designed three type of grounding strategies that leverage the general knowledge-base (such as Google Knowledge Graph) to assist language understanding. These grounding strategies resolve ambiguous named-entities and unknown words via a combination of different natural language processing algorithms, such as named-entity detection and statistical language generation. However, grounding strategies are not limited to grounding strategies. Knowledge bases could be used to design other strategies to assist language understanding and generation. For example, I can use knowledge bases to address the co-reference problem in conversations to assist language understanding. An example strategy could be if user mentioned "SIGDIAL" and then later mentioned "the academic conference" in an immediate follow-up utterance. Then, with knowledge base information stating SIGDIAL is an academic conference, I could infer that "SIGDIAL" and " the academic conference" refer to the same entity. In this thesis, I only started to tap into part of the powerful usage of knowledge bases via designing grounding strategies, which are essential for communication [Clark and Brennan, 1991]. In the future, I wish to expand our reportorial of grounding strategies.

1. **Ground on named entities.(GroundEntity)** I perform a shallow parsing to find the named entity in the sentence, and then retrieve a short description of the named entity in a knowledge base. Finally I use several templates to generate sentences using the obtained short description. An example grounding strategy for entity disambiguation is when users mentioned “Clinton”. “Clinton” can refer to “Bill Clinton” or “Hillary Clinton”, thus the system asks “Which Clinton are you talking about? Bill Clinton, the 42rd president of the United States or Hillary Clinton, the secretary of the state?” Users feel like they are understood when this strategy is triggered correctly. In addition, I make sure to never ground the same named-entity twice in single conversation.
2. **Ground on out of vocabulary words.(GroundOOV)** If the user says a word that is out of the system’s vocabulary, such as “confrontational”. Then the chatbot asks: “What is confrontational?” I expand our vocabulary with the new user-defined words continuously, so I will not ask for grounding on the same word twice.
3. **Ground on single-word sentences.(GroundSingle)** If user types in meaningless single word such as ‘d’, ‘dd’, or equations such as ‘1+2=’, that is not in a standard knowledge base. Then the chatbot replies: “Can you be serious and say things in a complete sentence?” to deal with such condition.

8.3.3 Personalized Strategy

Personal knowledge base expands along the conversation with respect to each individual user. An example entry of personal knowledge is a user’s preferred conversation topic, such as movies. I used information extraction methods to extract personalized knowledge from conversation history to form the personal knowledge base with respect to each user. Strategies that leverage personalized knowledge provide users with a proactive, effective and personalized experience. I designed two types of personalized strategies.

1. **Don’t repeat.(NoRepeat)** If the user repeat themselves, the system confronts the user by saying:“You already said that!”.
2. **Personalized Suggestion.(PersonalSuggest)** This strategy refers to actions that embed user’s personal knowledge and is designed to achieve a personalized user experience. An example personalized strategy is to suggest “Captain America 2” given that a user’s favorite movie type is superheroes.

8.4 Experiment

To test these strategies' effects, I conducted two sets of user study. In both settings, I modified the TickTock baseline system to include all the strategies mentioned. I used keyword retrieval method to generate responses. Please refer to Chapter 4 for details.

Experiment 1 I used TickTock's typing interface to collect data with a lower cost. In this experiment, I only randomly select an active participation strategy to maintain user engagement, but not to handle user disengagement. Because user's engagement cannot be detected accurately using only text features according to results shown in Chapter 6. I recruited users on crowd-source platform, Amazon Mechanical Turk and controlled the users to have an approval rate that is higher than 98% and are located in the U.S. In total, I collected 50 conversations from unique users.

Experiment 2, I used TickTock's multimodal interface, so I could test active participation strategies' effect on improving user engagement as well. I designed two systems: *Maintain-Only* and *Maintain-Improve*. The Maintain-Only system randomly selects among all the active participation strategies above to maintain user engagement whenever the generated response's confidence is low and none of the grounding and personalized strategies are applicable. The Maintain-Improve system randomly selects among all the active participation strategies to maintain user engagement, as well as to improve user engagement. In other words, the Maintain-Improve system also uses the active participate strategies to react to user's disengagement. The user engagement is manually labeled in-real time by a wizard in the Maintain-Improve system. The reason to have a wizard in the experiment is to avoid engagement detection error's influence. In the next chapter, I will describe a system with automatic engagement predictor.

To avoid culture and language proficiency confounds, all participants recruited were born and raised in North America. I also balanced gender. I had 10 people (6 males) people interacted with the Maintain-Only system and 12 people (7 males) interacted with the Maintain-Improve system. Participants were all university students and none of them had interacted with a multimodal dialog system before. There were no repeat users in the two groups. I also collected how frequently participants use spoken dialog systems, such as Apple Siri, in the exist survey and found that 25% of them have used dialog systems frequently. An example dialog of a user interacting with the Maintain-Only system was shown in Table 12.1. In Turn 5, the system used the *switch* strategy to deal the condition that it was unable to retrieve a response with similar content to the user's utterance. An example dialog from the Maintain-Improve system was shown in Table 9.7. In Turn 3, the system used the *initiation* strategy to deal with the fact that the user was not engaged in the previous turn.

TurnParticipant		TickTock
1	Your noise is a question mark, I think that's interesting.	Of course, what's your idea?
2	I think it fits you perfectly.	Let me see, his nose is still broke.
3	Do you want me to make your nose broken?	Ah, yes, why did you ask me for this?
4	Because I thought you might mind.	Well, I am not getting into specifics about what doctor treated him, how he was treated.
5	You have not been treated yet. That is in the future.	Let me see, by the way, are you interested in movies?

Table 8.1: An example conversation produced with the Maintain-Only system

TurnParticipant		TickTock
1	How are you	I think so, any thoughts?
2	How are you doing?	I am fine, how are you, any thought?
3	Any thought?	Let me see, do you want to go to my friends' party tonight?
4	No I have a lot of work to do. I can't go.	Everyday?

Table 8.2: An example conversation conversation with the Maintain-Improve system.

8.5 Data Annotation

I used similar annotation methods for both experiments. I annotated each conversational exchange with user engagement and response appropriateness. One conversational exchange includes one speech segment from the system and the following speech segment from the user. User engagement is defined as "user's interest to continue, to contribute the conversation", and the system appropriateness as "the coherence and appropriateness of the system utterance given the context". See Chapter 6 and Chapter 5 for the engagement annotation scheme and the response appropriateness annotation scheme respectively. When users failed to respond, the engagement value is 0. I labeled the turns that the system asked the user to repeat to be 0 in appropriateness. I exclude these turns in the analysis in order to avoid adding system failure in the analysis. Appropriateness was annotated by user themselves after the interaction is finished while engagement was annotated by experts. The inter-annotator agreement between the two experts on text-only data is relatively lower (kappa =0.43). However, if I collapse the rating for "strongly disengaged" with "disengaged" and "very engaged" with "engaged", our agreement improves (kappa = 0.67).

8.6 Results and Analysis

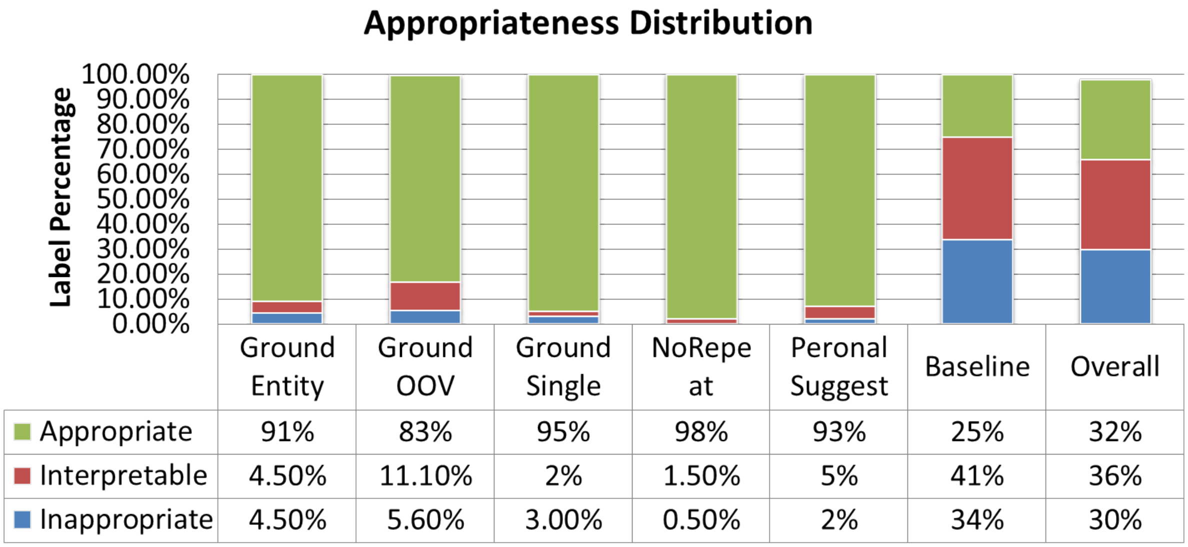
Based on the collected data and annotations, I conducted statistical analysis to report the effect of different strategies. Most strategies are proved to be beneficial to the appropriateness evaluation metric, except some active participation strategies have mixed results.

8.6.1 Grounding Strategies and Personalized Strategies

In Experiment 1, I tested the validity of all the grounding strategies and personalized strategies. Figure 8.1 shows all three grounding strategies lead to majority of appropriate responses. However, there were not appropriate response caused by errors in natural language precessing techniques that were used in generating these strategies. For example the errors in entity-detection lead to wrong entity being detected, and thus the GroundEntity strategy that generated based the wrong entity was considered inappropriate. The limitation of the knowledge base also contributed to some not appropriate responses. For example, some trendy Internet words (e.g. AMA means “ask me anything”) were not included in the Google Knowledge Graphs, thus the system treated them as out-of-vocabulary words and asked the user to explain them. The users on the other hand expect the system know these words, thus rated the system as being not appropriate for asking them for explanations. Failure of handling typos also contributed not appropriate responses in generated GroundOOV strategy. I did not include the function of automatic typo correction in our typing interfaces. I also didn’t have the typo converting process in our natural language understanding pipeline, thus I would consider those typos as out-of-vocabulary words and ask users to explain them. While users expect the system to understand typos, thus thought the system being not appropriate. Trendy Internet words and typos were also main factors that contributed to the inappropriateness for GroundSingle strategies, as sometimes user just say a single word and it happened to be a trendy Internet word or typo, then the GroundSingle strategy would prompt the user to be serious and say something in a complete sentences, while some users find them inappropriate. These two phenomenas were only found in the typing version, while the audiovisual version do not have them. Because users tend to use trendy Internet words in spoken language and speech doesn’t have typos. However, speech will introduce ASR errors in language understanding and I will introduce methods to address this issue in the next chapter.

I also found the two personalized strategies contributed to high appropriate responses ratio as well. The failure cases for NoRepeat were simply due to rare cases, users have the same answer to the two consecutive system utterances. Such as generic ones, I don’t like it or I am not sure. The failure cases for PersonalSuggest were mainly the failure of knowledge extraction methods in finding the correct relationship between entities. I wish to leverage recent works on knowledge base inferences to improve our personal knowledge base constructions.

As I mentioned in Chapter 6 and shown again in Figure 8.1 the baseline version of the TickTock have a low appropriateness ratio, after applying all the conversation strategies, including both active participation strategies and knowledge-base strategies, the improvement is statistically significant ($p < 0.01$). The appropriateness ratio improved from 25% to 32%. However, these knowledge-bases have trigger conditions, they were only applicable to a number of utterances, thus even with their above 80% appropriateness rate, the improvement on overall conversation only improve the ratio to 32% percent. One future work is to expand the set of conversation strategies to cover more user utterances.

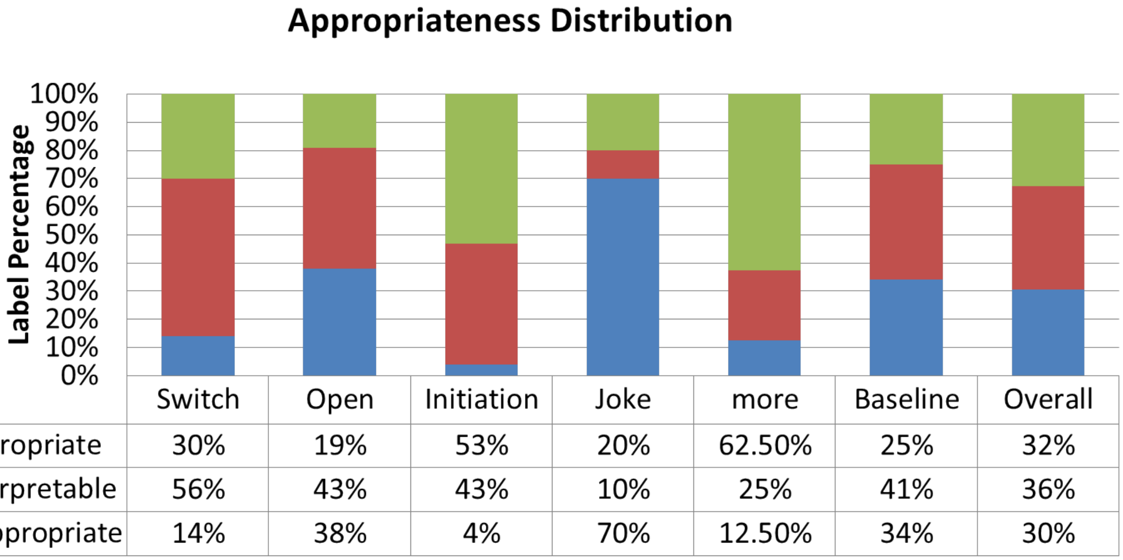


	Ground Entity	Ground OOV	Ground Single	NoRepe at	Peronal Suggest	Baseline	Overall
🟩 Appropriate	91%	83%	95%	98%	93%	25%	32%
🟥 Interpretable	4.50%	11.10%	2%	1.50%	5%	41%	36%
🟦 Inappropriate	4.50%	5.60%	3.00%	0.50%	2%	34%	30%

Figure 8.1: Appropriateness distribution over different grounding and personalized strategies

8.6.2 Active Participation Strategies for Engagement Maintenance

Figure 8.2 shows the active participation strategies have a mixed effect on appropriateness in Experiment 1. The “More” and “Initiation” strategies are producing more appropriate responses than not appropriate ones in general. However, the “Switch”, “Open” and “Joke” strategies are worse. This is because these active participation strategies were used only when the generated response’s confidence was low and none of the grounding and personalized strategies were applicable. In other words, if the system does not use active participation strategies, it will produce an inappropriate response anyway. Therefore, these strategies contributed to the reduction of inappropriate and interpretable responses (12% of total utterance). In the next chapter, I will describe algorithms that further reduce the number of incorporate and interpretable responses by



	Switch	Open	Initiation	Joke	more	Baseline	Overall
🟩 Appropriate	30%	19%	53%	20%	62.50%	25%	32%
🟥 Interpretable	56%	43%	43%	10%	25%	41%	36%
🟦 Inappropriate	14%	38%	4%	70%	12.50%	34%	30%

Figure 8.2: Appropriateness distribution over different active participation strategies

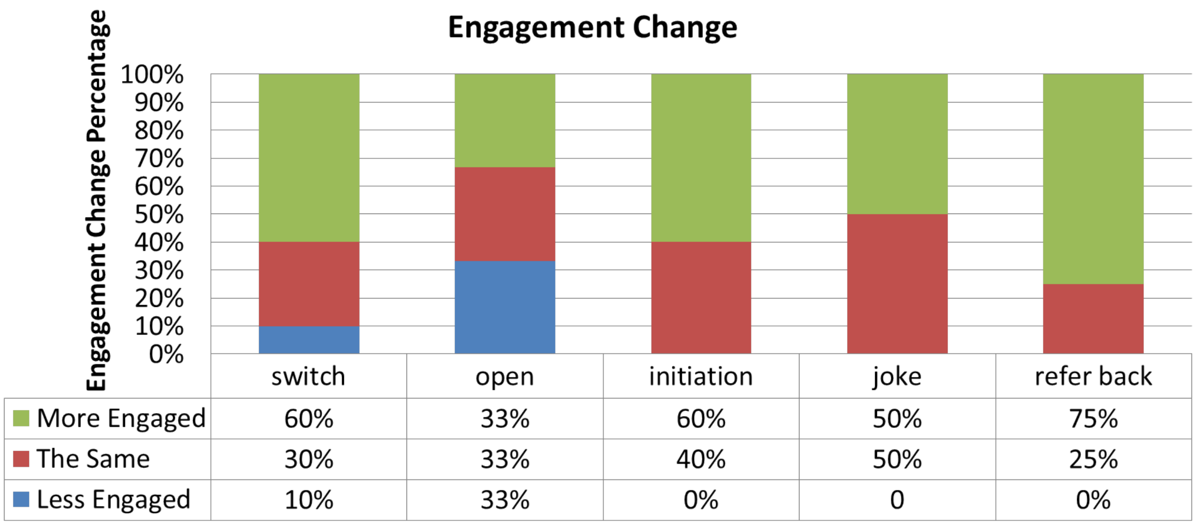
choosing appropriate active participation strategies with respect to conversation history.

8.6.3 Active Participation Strategies for Engagement Improvement

I further test the hypothesis if the active participation strategies could help users to improve their engagement state when they are disengaged. I described the results in Figure 8.3. I found that different strategies have a different effect in terms of improving the user engagement. For example, the Switch, Initiation and Refer back strategies were pretty effective in terms of improving users engagement, however the Open and Joke strategies were less effective, I speculate it was correlated with these strategies being appropriate. In next section, I will go deep into understanding the relationship between appropriateness and engagement. One caveat is that when conducting Experiment 2, the “More” strategy was not implemented by then, so I could not report the detailed effects of this strategy. However, I believe it should follow the same pattern. I believe the reason that these active participation strategies increased user’s engagement is because it provides the user with an active lead of what to do next, which motivates the user to continue the conversation.

8.7 Maintain-Only System VS. Maintain-Improve System

To test if Maintain-Only system or the Maintain-Improve system elicits better user engagement. I selected video recordings with participants who are not familiar with dialog systems. There



	switch	open	initiation	joke	refer back
🟩 More Engaged	60%	33%	60%	50%	75%
🟥 The Same	30%	33%	40%	50%	25%
🟦 Less Engaged	10%	33%	0%	0	0%

Figure 8.3: User engagement change distribution when system reacts to low user engagement.

are only five participants in the Maintain-Only dataset and nine participants in the Maintain-Improve dataset who meet this requirement. In order to balance the two sets, I randomly selected five participants from the nine in the Maintain-Improve. I picked one video from each dataset to form a A/B comparison study. In total there are 25 pairs, and I recruited three raters for each pair. Nobody rated the same pair twice. I ask them to watch the two videos and then compare them through a set of questions including “Which system resulted in a better user experience?”, “Which system would you rather interact with?” and “Which person seemed more enthusiastic about talking to the system”. In addition, I also included some factual question related to the video content in order to test if the rater had watched the video, which all of them had. Raters are allowed to watch the two videos multiple times. The limitations of such a comparison is that some system failures, such as ASR failure, may affect the quality of the conversation, which may be a confound. In the task, I specifically asked the users to overlook these system defects, but they still commented on these issues in their feedback. I will collect more examples in the future to balance the influence of system defects.

In our study, I found that a system that reacts to low user engagement and possible system breakdowns is rated as having better user engagement and experience compared to a system that only reacts to possible system breakdowns. This rating held true for both experts and non-experts. I performed an unbalanced Student’s t-test on expert-rated user engagement of turns in Maintain-Only and Maintain-Improve and found the engagement ratings are statistically different ($p < 0.05$). Maintain-Improve has more user engagement (Maintain-Only: Mean = 3.09 (SD = 0.62); Maintain-Improve: Mean = 3.51 (SD = 0.78). A t-test on utterances that are not produced by designed strategies shows the two systems are not statistically different in terms of user engagement ($p = 0.13$). This suggests that the difference in user engagement is mostly due to the utterances that are produced by strategies. Experts also rated the interaction for overall user experience and I found that Maintain-Improve interactions are rated significantly higher than Maintain-Only system overall ($p < 0.05$).

In Maintain-Improve, 37% of the strategies were triggered to react to low user engagement and 63% were used to deal with low generation confidence. Among the strategies that were triggered to react to the low user engagement, 72% of them lead to user engagement improvement. I believe the ability to react to low user engagement is the reason that Maintain-Improve has more user engagement than Maintain-Only. Another reason is that the Maintain-Improve has an extra strategy, *refer back*, which in general performs best in improving user engagement. In the user survey, one of the participants also mentioned that he likes the Maintain-Improve system because it actively proposes engaging topics.

For non-expert ratings, I designed 25 A/B comparison tasks. Each task had three raters, and majority vote was used to combine them. Non-experts rated the Maintain-Improve system as more engaging in 12 tasks, and the Maintain-Only system as more engaging in three tasks. In the rest of the ten tasks, non-experts expressed no difference between the two systems. The results indicated that non-experts believed the Maintain-Improve system elicited more user engagement than the Maintain-Only system in our small-sample study. This conclusion is also true when the systems are judged on which leads to a better user experience. I examined the three tasks on which the Maintain-Only system is rated higher than Maintain-Improve and found that two of them involved the same interaction produced by Maintain-Only. In that interaction, the user was very actively interpreting the system's utterance and was responding with interesting questions. Table 12.1 shows a part of that interaction.

Turn	Participant	TickTock
1	Your noise is a question mark, I think that's interesting.	Of course, what's your idea?
2	I think it fits you perfectly.	Let me see, his nose is still broke.
3	Do you want me to make your nose broken?	Ah, yes, why did you ask me for this?
4	Because I thought you might mind.	Well, I am not getting into specifics about what doctor treated him, how he was treated.
5	You have not been treated yet. That is in the future.	Let me see, by the way, are you interested in movies?
6	Yes, I watch movies sometimes.	Great.

Table 8.3: An engaging example conversation produced by the Maintain-Only system

8.8 System Appropriateness and User Engagement

In conversations produced using the Maintain-Only system, an unbalanced Student's t-test of engagement change between turns that are appropriate and ones that are inappropriate shows that turns that are appropriate (Mean =0.33, (SD=0.84)) have significantly ($p = 0.01$) better engagement change than turns that are inappropriate (Mean = -0.01, (SD=0.92)). Figure 8.4 shows a box plot of the resulting engagement change from appropriate and inappropriate responses. The figure suggests that having appropriate responses leads to better engagement change overall. However some inappropriate responses lead to positive engagement change as well. The same trend is found in conversations produced by the Maintain-Improve system.

I tested the hypothesis with respect to each strategy via an unbalanced Student's t-test. The hypothesis holds for the Switch, Initiation and Joke strategies. It did not hold for the Open strategy, but this is probably because there were very few examples of Open strategies were triggered and rated appropriate, making it hard to yield any statistical significance. In addition, across all responses, I find some outliers, where even though the system's response is appropriate the user's engagement decreased. This may happen when the system provides a simple 'yes' or 'no' answer, when the system interrupts the user, or when the user misunderstands the system. Some users are not familiar with synthetic voices and misheard the system, and thus thought the system was inappropriate.

I believe that in the future I can improve our system's turn-taking mechanism and try to tune the system retrieval method to prefer longer responses. This will help to overcome the issue that even appropriate answers can lead to a decrease in user engagement. Since appropriate

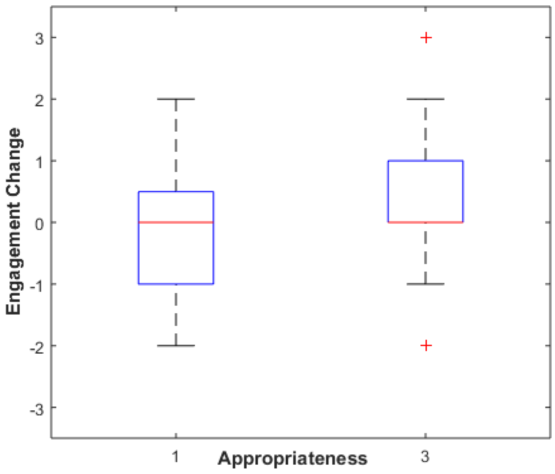


Figure 8.4: User engagement change with respect to system appropriateness in the Maintain-Only system.

system responses make users more engaged, are all the positive engagement changes the result of appropriate responses? I performed an unbalanced t-test of the appropriateness values between turns that have positive engagement change (Mean = 1.79 (SD = 0.82)) and turns that have negative engagement change (Mean = 1.53 (SD = 0.67)) and found that they are statistically significant ($p < 0.05$). I examined the recordings of conversations and found that there are other factors that contribute to the engagement change other than the system's appropriateness. For example, funny comments and provocative utterances on the part of the system can also increase user engagement. In Table 12.1, the system response in Turn 4 is only rated as "Interpretable," and yet it leads to an increase in user engagement. The speaker even smiled when replying to the system. In another interaction, "Don't talk to an idiot, because they will drag you down to the same level and beat you with experience." is rated as "Inappropriate" with respect to the previous user utterance. However the user reacted to it with increased engagement and asked the system: "Are you calling me an idiot, TickTock?". I conclude that being appropriate is important to achieve better user engagement, however it is not the only way.

8.9 Conclusion

I designed a set of conversational strategies to improve the system's response appropriateness and user engagement. All five knowledge-base strategies contributed to the improvement of the overall system appropriateness. active participation strategies have a mixed performance in terms of being appropriate when randomly selected to tackle the situation when the response generation confidence was low and knowledge-base strategies were not applicable. However with high rate they were able to achieve their purpose of avoiding inappropriate responses. During the analysis I also found interesting relationship between user engagement and system appropriateness. I found appropriate responses lead to better engagement change compared to inappropriate responses.

In a third-person study, both experts and non-experts rated the system that reacts to both low user engagement and low generation confidence as having more overall user engagement than the system that only reacts to low generation confidence. Thus I conclude that the improvement gained by reacting to user's engagement is generally recognizable.

Chapter 9

Global Planning Policy

After introducing and validating the positive effect of conversation strategies in promoting user engagement and system appropriateness. We designed several dialog policies to select among conversational strategies mentioned in the previous chapter. We propose a global planning policy that takes conversation history in consideration in selecting conversational strategies in order to maximize the long-term conversation performances. The global planning policy learned via reinforcement learning outperforms the random selection policy and the locally greedy policy in both the simulated and the real-world settings in different metrics. These metrics consider both the local and global quality of the conversation.

9.1 Introduction

There are a variety of methods to generate responses for non-task-oriented systems, such as machine translation [Ritter et al., 2011], retrieval-based response selection [Banchs and Li, 2012], and sequence-to-sequence recurrent neural network [Vinyals and Le, 2015]. However, these systems still produce utterances that are incoherent or inappropriate from time to time. To tackle this problem, we propose a set of conversational strategies in the previous chapter, such as switching topics, to avoid possible inappropriate responses (breakdowns) and improve user engagement. Another difficulty is to decide which strategy to select with respect to different conversational contexts. In a multi-turn conversation, if the same strategy is used repeatedly, the user experience will be affected. We experiment on three dialog policies: a random selection policy that randomly selects a policy regardless of the dialog context, a locally greedy policy that focuses on local dialog context, and a reinforcement learning policy that considers the entire dialog context. The conversational strategies and policies are applicable for non-task-oriented systems in general, regardless of the response generation method. We implement three policies in a keyword retrieval driven non-task-oriented system. We use the retrieval confidence as the criteria to decide whether a strategy is needed to be triggered to avoid system breakdowns.

Reinforcement learning was introduced to the dialog community two decades ago [Biermann and Long, 1996] and has mainly been used in task-oriented systems [Singh et al., 1999]. Researchers have proposed to design dialogue systems in the formalism of Markov decision processes (MDPs) [Levin et al., 1997] or partially observable Markov decision processes (POMDPs) [Williams and Young, 2007]. In a stochastic environment, a dialog system's actions are system utterances, and the state is represented by the dialog history. The goal is to design a dialog system that takes actions to maximize some measure of system reward, such as task completion rate or dialog length. The difficulty of such modeling lies in the state representation. Representing the dialog by the entire history is often neither feasible nor conceptually useful, and the so-called belief state approach is not possible, since we do not even know what features are required to represent the belief state. Previous work [Walker et al., 1998] has largely dealt with this issue by imposing prior limitations on the features used to represent the approximate state. In this paper, instead of focusing on task-oriented systems, we apply reinforcement learning to design a policy to select designed conversation strategies in a non-task-oriented dialog systems. Unlike task-oriented dialog systems, non-task-oriented systems have no specific goal that guides the interaction. Consequently, evaluation metrics that are traditionally used for reward design, such as task completion rate, are no longer appropriate. The state design in reinforcement learning is even more difficult for non-task-oriented systems because the action state space is much bigger. One slightly different answer would lead to a completely different conversation; moreover there is no clear sense of when a conversation is "complete". We simplify the state design by introducing expert knowledge, such as not repeating the same strategy in a row, as well as statistics obtained from conversational data analysis.

We implement and deploy a non-task-oriented dialog system driven by a statistical policy to avoid possible system breakdowns using designed conversation strategies. We evaluate the system on the Amazon Mechanical Turk platform with metrics that consider both the local and the global quality of the conversation.

9.2 Related Work

Many generic conversational strategies have been proposed in previous work to avoid generating incoherent utterances in non-task-oriented conversations, such as introducing new topics (e.g. "Let's talk about favorite foods!") in [Higashinaka et al., 2014], asking the user to explain missing words (e.g. "What is SIGDIAL?") [Maria Schmidt and Waibel, 2015]. They were triggered with pre-defined rules. No researcher has investigated thoroughly on which strategy to use in different conversational contexts. Compared to task-oriented dialog systems, non-task-oriented systems have more varied conversation history, which are thus harder to formulate as a mathematical problem. In this work, we propose a method to use statistical findings in conversational study to constrain the dialog history space and to use reinforcement learning for statistical policy learning in a non-task-oriented conversation setting.

To date, reinforcement learning is mainly used for learning dialogue policies for slot-filling task-oriented applications such as bus information search [Lee and Eskenazi, 2012], restaurant recommendations [Jurcicek et al., 2012], and sightseeing recommendations [Misu et al., 2010]. Reinforcement learning is also used for some more complex systems, such as learning negotiation policies [Georgila and Traum, 2011] and tutoring [Chi et al., 2011]. Reinforcement learning is also used in question-answering systems [Misu et al., 2012]. Question-answering systems are very similar to non-task-oriented systems except that they do not consider dialog context in generating responses. They have pre-existing questions that the user is expected to go through, which limits the content space of the dialog. Reinforcement learning has also been applied to a non-task-oriented system for deciding which sub-system to choose to generate a system utterance [Shibata et al., 2014]. In this paper, we used reinforcement learning to learn a policy to sequentially decide which conversational strategy to use to avoid possible system breakdowns.

Task completion rate is widely used as the conversational metric for task oriented systems [Williams and Young, 2007]. However, it is not applicable for non-task-oriented dialog systems, as they don't have a task. Response appropriateness (coherence) is a widely used manual annotation metric [Yu et al., 2016e] for non-task-oriented systems. However, this metric only focuses on the utterance level conversational quality and is not automatically computable. Perplexity of the language model is an automatically computable metric but is difficult to interpret [Vinyals and Le, 2015]. We propose three metrics: *turn-level appropriateness* (captures local coherence), *conversational depth* (captures topic consistency) and *information gain* (captures conversation variety), which assess both the local and the global conversation quality. Although only *Information gain* is automatically quantifiable, we use supervised machine learning methods to built automatic detectors for *turn level appropriateness* and *conversational depth*.

9.3 Dialog Policy Design

In previous chapter, we used a random selection policy that randomly chooses among active participation strategies whenever grounding and personalized strategies are not applicable. We find that the sentiment polarity of the utterance has an influence on which active participation strategy to select that leads to appropriate response. People tend to rate the *switch* strategy more favorably if there is negative sentiment in the previous utterances. For example:

TickTock: Hello, I really like politics. Let's talk about politics.
User: No, I don't like politics.
TickTock: Why is that?
User: I just don't like politics.
TickTock: OK, how about we talk about movies?

In another scenario, when all the previous three utterances are positive, the *more* strategy (e.g. Do you want to talk more about that?) is preferred over the *switch* strategy (e.g. Do you like movies?).

We set out to find the optimum strategy to deal with the user utterance given the sentiment polarities of its previous three utterances. We generated five different versions of the conversations by replacing the original used active participation strategy with other active participation strategies. We asked people to rate the strategy's appropriateness given its three previous utterances. For each conversation, we collected ratings from three different raters and used the majority rating as the final score. Then we constructed a table of a distribution that represents the system response's appropriateness regarding each strategy. We collect 10 ratings for each strategy under each context. We use the Vader [Hutto and Gilbert, 2014] sentiment predictor for automatic sentiment prediction. The sentiment predictor produces a label with three categories: positive (pos), negative(neg) and neutral (neu).

We found that the results of the rating task supports our hypothesis that different strategies are preferred with respect to different sentiment context. In Table 9.1, we show the distribution of the appropriateness ratings for all the active participation strategies in a context when all their previous utterances are positive. Users rated the *more* strategy more appropriate than the *end* strategy and the *switch* strategy. One interesting observation is that the *joke* strategy is rated poorly. We examine all the cases and find that the low appropriateness rate is mostly due to the fact that the joke is unexpected given the context. The *initiation* strategy can be appropriate when the activity fits the previous content semantically.

In another sentiment context, when there are consecutive negative utterances, the *switch* strategy and the *end* strategy are preferred. We can see that which strategy is appropriate is heavily dependent on the immediately sentiment context of the conversation. Sentiment polarity captures some conversational level information which is a discriminating factor. We use these findings to design the locally greedy policy. The system deal with user's utterance uses the strategy that is rated as the most appropriate given the utterance's three previous utterances sentiment polarity .

We conduct another Amazon Mechanical Turk study to test if sentiment context beyond three utterances would influence the preferred strategy or not. To reduce the work load, we test on one condition which is when the previous three utterances are all positive. We provide the complete conversation history of that dialog to the raters instead of only three previous utterances. We find that strategies used most recently are rated less favorably if used again. This motivates us to include information that relates to the usage of the previous strategy and a longer history to design policy that cares about global context.

Strategy	App	Inter	Inapp
switch	0.1	0.3	**0.6**
initiation	0.2	**0.4**	**0.4**
joke	0.1	0.2	**0.7**
end	0.1	0.3	**0.6**
more	0.4	**0.5**	0.1

Table 9.1: Appropriateness rating distribution when the previous three utterances are positive.

9.4 Reinforcement Learning

$$Q_{t+1}(s_t, a_t) \leftarrow Q_t(s_t, a_t) + \alpha_t(s_t, a_t) \cdot \Big(R_{t+1} + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t) \Big) \tag{9.1}$$

$$\text{Accumulated Appropriateness} * 0.3 + \text{Conversational depth} * 0.3 + \text{Information gain} * 0.4 \tag{9.2}$$

$$\begin{gathered}\text{Accumulated Appropriateness} * 0.2 + \text{Conversational depth} * 0.2 + \\ \text{Information gain} * 0.3 + \text{Overall User Engagement} * 0.3\end{gathered} \tag{9.3}$$

We modeled the conversation process as Markov Decision Processes (MDPs). Then we can use reinforcement learning to solve the it. The goal is to learn a conversation policy that makes sequential decisions considering conversation history. We used Q-learning, a model-free algorithm to tackle the conversation policy learning. The reason of choosing Q-learning is it handles discrete states well and since it generates a Q table after training, which brings the potential to understand the model in order to further provide accountability. Another advantage of Q-learning is that it makes encoding expert knowledge easier, as it is a model-free algorithm and the expert knowledge can be easily encoded in the transition functions. So Q-learning is a good choice for this setting.

In reinforcement learning setting, we formulated the problem as $(\mathbb{S}, \mathbb{A}, \mathbb{R}, \gamma, \alpha)$, where $\mathbb{S}$ is a set of states that represents the system's environment, in this case the conversation history so far. $\mathbb{A}$ is a set of actions available per state. In our setting, the actions are strategies available. By performing an action, the agent can move from one state to another. Executing an action in a specific state provides the agent with a reward (a numerical score), $\mathbb{R}(s, a)$. The goal of the agent is to maximize its total reward. It does this by learning which action is optimal to take for each state. The action that is optimal for each state is the action that has the highest long-term reward. This reward is a weighted sum of the expected values of the rewards of all future steps starting from the current state, where the discount factor γ is a number between 0 and 1 that trades off the importance of sooner versus later rewards. γ may also be interpreted as the likelihood to succeed (or survive) at every step. The algorithm therefore has a function that calculates the quantity of a state-action combination, $\mathbb{Q} : \mathbb{S} \times \mathbb{A} \rightarrow \mathbb{R}$. The core of the algorithm is a simple value iteration update. It assumes the old value and makes a correction based on the new information at each time step, *t*. See Equation (1) for details of the iteration function.

The critical part of the modeling is to design appropriate states and the corresponding reward function. We reduce the number of the states by incorporating expert knowledge and the statistical findings in our analysis. We introduce two reinforcement learning policies: Engagement Maintenance Policy (Maintain-Only Policy) and Engagement Maintenance and Improvement Policy (Maintain-Improve Policy). The first focuses on improving long-term conversation coherence and variety and could be used in typing interface, and the second focuses on not only long-term conversation coherence and variety but also long-term user engagement and was designed for audiovisual interface. We use another chatbot, A.L.I.C.E. [1] as a user simulator in the training process for both policies.

9.4.1 Engagement Maintenance Policy

This policy aims to maintain user's engagement through improving the system's appropriateness, consistency and variety. We include features: turn index, response generation confidence, times each strategy was executed previously, the previous executed conversation strategy and the sentiment polarity of the previous three utterances in the state design. We construct the reward table based on the statistics collected from the previous experiment. To make the reward table tractable, we impose some of the rules we constructed based on expert knowledge. For example, if certain strategy has been used before, then the reward of using it again is reduced. If the trigger condition of any knowledge-base strategies were meet, the system chooses them over all engagement strategies. This may result in some less optimum solutions, but reduces the state space and action space considerably.

[1] http://alice.pandorabots.com/

9.4.2 Engagement Maintenance and Improvement Policy

To realize the full potential the SI framework, we integrated user engagement in the reinforcement learning formulation. We designed another policy that not only deal with system inappropriateness but also low user engagement by adding real time user engagement as one of the state variable in the Q-learning function. Due to the inability of simulating human nonverbal behaviors, we only used text features to predict user engagement in the training process. As we mentioned in Chapter 6, only using general text features, such as “word count” is not sufficient to predict engagement accurately. Thus we incorporated semantic features to boost the performance. In particular, we included the word2vec features [Mikolov et al., 2013]. We also changed the original regression model to a binary classification model (Above or equal to 3 in a 5-point Liker scale is considered to be engagement and below 3 is considered to be disengaged). The performance of the “turn-level user engagement detector” is 82.2% in accuracy, while the majority vote baseline accuracy is 70.3%. By doing so, we lost the precision of user engagement, but boosted the accuracy for detecting lack of user engagement. Since the policy only reacts to user’s disengagement, so detecting disengagement is the priority. This is the reason that we converted the original regression model to a binary classification model in this policy. However, if another policy which will react to more fine-grind user engagement, then we would recommend adopting the regression model for user engagement. In this policy, the state included all the information included in the Engagement Maintenance Policy plus the real-time user engagement. We also added extra expert information to the reward function about preferring engagement strategy when user is detected as disengaged. We also include the overall user engagement of the entire conversation as a delayed reward at the end of the conversation, so the policy will guide the system to improve user engagement along with system coherence and variety.

9.5 Evaluation Metrics

To train the Engagement Maintenance Policy, we used a metric which was a linear combination of three metrics: the *accumulated turn-level appropriateness*, the *conversational depth* and the *information gain*. While for the Engagement Maintenance and Improvement Policy, we included an extra metric, the *overall user engagement*. We will describe each metric and the methods to obtain them automatically below.

Label	Definition	Example
Inappropriate (Inapp)	Not coherent with the user utterance	*Participant*: How old are you? *TickTock*: Apple. .
Interpretable (Inter)	Related and can be interpreted	*Participant*: How old are you? *TickTock*: That's too big a question for me to answer.
Appropriate (App)	Coherent with the user utterance	*Participant*: How is the weather today? *TickTock*: Very good.

Table 9.2: Appropriateness rating scheme.

9.5.1 Turn-Level Appropriateness

Turn-level appropriateness reflects the coherence of the system's response in each conversational turn. See Table 9.2 for the annotation scheme. The inter-annotator agreement between the two experts is relatively high (kappa = 0.73). We collapse the "Appropriate" and "Interpretable" labels into one class and formulate the appropriateness detection as a binary classification problem. Our designed policies and strategies intend to avoid system breakdowns (the inappropriate responses), so we built this detector to tell whether a system response is appropriate or not.

We annotated the appropriateness for 1256 turns. We balance the ratings by generating more inappropriate examples by randomly pairing two utterances. In order to reduce the variance of the detector, we use five-fold cross-validation and a Z-score normalizer to scale all the features into the same range. We use early fusion, which simply concatenates all feature vectors. We use a v-Support Vector [Chang and Lin, 2011] with a RBF Kernel to train the detector. The performance of the automatic appropriateness detector is 0.73 in accuracy while the accuracy of the majority vote is 0.5.

We use three sets of features: the strategy used in the response, the word counts of both the user's and TickTock's utterances, and the utterance similarity features. The utterance similarity features consist of a feature vector obtained from a word2vec model [Mikolov et al., 2013], the cosine similarity score between the user utterance and the system response, and the similarity scores between the user response and all the previous system responses. For the word2vec model, we trained a 100-dimension model using the collected data.

9.5.2 Conversational Depth

Conversational depth reflects the number of consecutive utterances that share the same topic. We design an annotation scheme (Table 9.3) based on the maximum number of consecutive utterances on the same topic. We annotate conversations into three categories: "Shallow", "Inter-

Conversational depth	Consecutive utterances
Shallow	< 6
Intermediate	[7, 10]
Deep	> 10

Table 9.3: Conversational depth annotation scheme

mediate" and "Deep". The annotation agreement between the two experts is moderate (kappa = 0.45). Users manually labeled 100 conversations collected using Text-TickTock 1.0 and 2.0. We collapse "Shallow" and "Intermediate" into one category and formulate the problem as a binary classification problem. We use the same machine learning setting as the turn level appropriateness predictor. The turn level appropriateness predict is trained on 100 conversations collected via the text-only TickTock. The performance of the automatic conversational depth detector is 72.7% accuracy, while the majority vote baseline accuracy is 63.6%. The conversational depth detector has three types of features:

1. The number of dialogue exchanges between the user and TickTock, and the number of times TickTock uses the *continue*, *switch* and *end* strategy.
2. The count of a set of keywords used in the conversation. The keywords are "sense", "something" and interrogative pronouns, such as "when", "who", "why", etc. "Sense" often occurs in sentence, such as "You are not making any sense" and "something" often occurs in sentence, such as "Can we talk about something else?" or "Tell me something you are interested in." Both of them indicate a possible topic change. Interrogative pronouns are usually involved in questions that probe users to express more on the current topic.
3. We convert the entire conversation into a vector using doc2vec and also include the cosine similarity scores between adjacent responses of the conversation.

Policy	Appropriateness	Conv depth	Info gain
Random Selection	62%	32%	50.2
Locally Greedy	72%	34%	62.4
Reinforcement Learning	82%	45%	68.2

Table 9.4: Performance of different policies in the simulated setting

9.5.3 Information Gain

Information gain reflects the number of unique words that are introduced into the conversation from both the system and the user. We believe that the more information the conversation has,

Policy	App	Inter	Inapp	Conv depth	Info gain
Random Selection	30%	36%	32%	30%	56.3
Locally Greedy	30%	42%	27%	52%	71.7
Reinforcement Learning	34%	43%	23%	58%	73.2

Table 9.5: Performance of different policies in the real-world setting.

the better the conversational quality is. This metric is calculated automatically by counting the number of unique words after the utterance is tokenized.

9.5.4 Overall User Engagement

This is a metric to evaluate users' overall user engagement. We adopted the same method described for the real-time turn-level engagement, except that we perform the prediction over an entire conversation instead of each conversational exchange, and we converted the regression formulation to binary classification. We treated user as engaged if his or her was rated above or equal to level 3 out 5, and as disengaged if below 3. The same 100 conversations used to train the turn level appropriateness predictor is used to train the overall user engagement. Since 100 data points was not sufficient to train the predictor, therefore, we augmented the data by expanding the dataset with similar data point. We generate similar data point by deleting the first one or two utterances, or the last and last two utterances. In all we trained our overall engagement predictor over 500 data points. The performance of the "overall user engagement detector" is 74.2% in accuracy, while the majority vote baseline accuracy is 66.2%. Previous study found that the conversation length is highly correlated to user overall engagement in chat dialogs [Yu et al., 2016f]. Thus in the future we would use the length of the conversation, which is a more objective measure to assess user overall engagement. Because in this study, we used a chatbot simulator, which is incapable of deciding when to quit. Thus it is not applicable in our study. However, if we could use real human users for the training process, then this would be applicable.

Since we used another chatbot to simulate users, making sure of a high overall conversation quality is critical. All metrics mentioned are related to each other but cover different aspects of the conversation. For the Engagement Maintenance Policy, we used a weighted score of the first three metrics mentioned for the learning process (see Equation 2 for details). While for the Engagement Maintenance and Improvement Policy, we used a weighted score of all four metrics mentioned. The coefficients are chosen based on empirical heuristics (see Equation 3 for details).

9.6 Experiment

During the training process, we constrain the simulated conversation to be 10 turns. The stepwise immediate reward is the system response appropriateness. In the transition function, we integrated an expert rule, penalizing repeated strategies in sequence. The delayed reward is a combination of the conversation depth and the information gain for the Maintain-Only Policy and a combination of the previous metrics plus overall user engaged for the Maintain-Improve Policy. It took 5000 conversations for this policy to converge.

9.7 Results and Analysis

Turn	Participant	TickTock
1	I am not sure. What's on your mind?	Well, you know, I'm not sure. Let me tell you something, politicians and diapers have one thing in common. They should both be changed regularly, and for the same reason.
2	Wow. That's funny and insightful.	Uhh, I can't answer this. Can we talk about something else?
3	How about movies. Have you seen any good movies?	Of course. What do you think?
4	I love super hero movies. Have you seen Deadpool?	Are you talking about Deadpool,the animation character?
5	Yes, exactly. They made a movie of him.	That sounds really interesting.

Table 9.6: An example conversation with TickTock in the real-world setting using Engagement Maintenance Policy

Turn	A.L.I.C.E.	TickTock
1	When do I do it?	I don't know.
2	That is interesting.	Do you want to go to my friend's party tonight?
3	Sure, just download me to your notebook or PDA and take me along.	Sounds great. Do you want to play board games together some time.
4	I like to play the Turing Game.	I don't know how to play it. Is it difficult to play?
5	Interesting gossip. It could be.	Possibly.

Table 9.7: An example conversation of TickTock in the simulated setting using Engagement Maintenance Policy

In the user experiments we found that reinforcement learning trained policy outperforms the random selection and local greedy algorithms. And the policy that considers both maintaining and improving user engagement outperforms the one only maintains user engagement.

9.7.1 Reinforcement Learning Policy VS. Non-Reinforcement Learning Policy

We evaluated the random selection, local greedy and Engagement Maintenance policies with respect to three evaluation metrics: *turn-level appropriateness*, *conversational depth* and *information gain*. We show the results in the simulated setting in Table 12.2 and the real-world setting in Table 12.3. In the simulated setting, users are simulated using a chatbot, A.L.I.C.E.. We show an example simulated conversion in Table 9.7. In the real-world setting, the users were people recruited on Amazon Mechanical Turk. We collected 50 conversations for each policy. We compute *turn-level appropriateness* and *conversational depth* using automatic predictors in the simulated setting and use manual annotations in the real-world setting.

The policy learned via reinforcement learning outperforms the other two policies in all three metrics with statistical significance ($p < 0.05$)in both the simulated setting and the real-world setting. The percentage of inappropriate turns decreases when the policy considers context in selecting strategies. However, the percentage of appropriate utterances is not as high as we hoped. This is due to the fact that in some situations, no generic strategy is appropriate. For example, none of the strategies can produce an appropriate response for a content-specific question, such as “What is your favorite part of the movie?” However, the Open strategy can produce a response, such as: “Sorry, I don’t know, tell me something you are interested.” which considered as “Interpretable”, and saved the system from a breakdown.

Both the reinforcement learning policy and the locally greedy policy outperform the random selection policy with a huge margin in conversational depth. The reason is that they take context into consideration in selecting strategies, while the random selection policy uses the Switch strategy randomly without considering the context. As a result, it cannot keep the user on the same topic for long. However, the reinforcement learning policy only outperforms the locally greedy policy with a small margin. Because there are cases when the user has very little interest in a topic, the reinforcement learning policy will switch the topic to satisfy the *turn-level appropriateness* metric, while the locally greedy policy seldom selects the Switch strategy according to the learned statistics. The reinforcement learning policy has the best performance in terms of information gain. We believe the improvement mostly comes from using the More strategy in the right context. The More strategy elicits more information from the user compared to the other active participation strategies.

In Table 9.7, we can see that the simulated user is not as coherent as a human user. In addition, the simulated user is less expressive than a real user, so the depth of the conversation is generally lower in the simulated setting than in the real-world setting.

We also conducted a face-to-face user study to test if the Maintain-Improve policy performs better than the Random Selection and Local Greedy policies. In this setting, the Random Selection policy would randomly select one engagement strategy if the user disengages and if the system cannot find other good actions to perform. The Local Greedy policy works the same as in the one for maintaining engagement, except that when users are not engaged, it will select the strategy based on the local context as well.

Policy	App	Inter	Inapp	Conv depth	Info gain
Random Selection	23%	37 %	40%	30%	43.3
Locally Greedy	25%	39%	36%	40%	50.70
Reinforcement Learning	30%	33%	34%	60%	53.20

Table 9.8: Performance of different policies for both user engagement maintain and improvement.

9.7.2 Engagement Maintenance and Improvement Policy VS Engagement Maintenance Policy

We also ran a user study to test whether the Maintain-Improve policy is better than the Maintain policy by considering user engagement in the statistical planing. For a fair comparison, we collected 10 face-to-face interactions with system's with each policy. Table 9.9 shows the results. The maintain-improve policy outperforms the maintain-only policy in all metrics, especially the expert annotated overall user engagement score, conversation depth, and conversation length (number of conversation exchanges). This indicates by caring about real-time exchange level engagement benefit the long-term planning.

In face-to-face interaction, the user engagement could be detected more accurately, however the ASR errors would also influence the response quality. Thus the inappropriateness response percentage in the face-to-face setting is always higher than the typing setting. Therefore, the next step would be to account for the ASR confidence in conversation policy planning.

Policy	App	Inter	Inapp	Conv depth	Info gain	Engagement	Length
Maintain-Improve	30%	33%	34%	60%	53.20	3.51	27.2
Maintain-Only	28%	33%	36%	54%	50.30	3.23	23.1

Table 9.9: Performance comparison of the Maintain-Improve Policy and Maintain-Only Policy in face-to-face communication.

9.8 Conclusion

We learned a policy that considers both the local and the global context of the conversation for conversation strategy selection using reinforcement learning methods. The policy learned by reinforcement learning outperforms the locally greedy policy and the random selection policy with respect to three evaluation metrics: turn-level appropriateness, conversational depth and information gain. The policy that cares both appropriateness and user engagement performed better than the policy that cares appropriateness only in a user study with a relative small pool of users.

Chapter 10

HALEF: A Task-Oriented Dialog System Framework

We present an open-source web-based multimodal dialog framework, HALEF, that integrates video conferencing and telephony abilities into the existing HALEF cloud-based dialog framework via the FreeSWITCH video telephony server and takes real-time predicted user engagement score in selecting dialog actions. Due to its distributed and cloud-based architecture, Multimodal HALEF allows researchers to collect video and speech data from participants interacting with the dialog system outside of traditional lab settings, therefore largely reducing the cost and labor incurred during the traditional audio-visual data collection process. The framework is equipped with a set of tools including a web-based user survey template, a speech transcription, annotation and rating portal, a web visual processing server that performs head tracking, gaze tracking, and action unit detection based on visual information, and a database that logs full-call audio and video recordings as well as other call-specific information.

10.1 Introduction

Previously, many end-to-end spoken dialog systems (SDSs) used close-talk microphones or handheld telephones to gather speech input [Eskenazi et al., 2008] [Zue et al., 2000] in order to improve automatic speech recognition (ASR) performance of the system. However, this limits the accessibility of the system. Recently, the performance of ASR systems has improved drastically even in noisy conditions [Hinton et al., 2012]. In turn, spoken dialog systems are now becoming increasingly deployable in open spaces [Bohus et al., 2014]. They can also interact with users remotely through the web without specific microphone requirements [McGraw et al., 2010], thus reducing the cost and effort involved in collecting interactive data.

Recently, multimodal sensing technologies such as face recognition, head tracking, etc. have also improved. Those technologies are now robust enough to tolerate a fair amount of noise in the visual and acoustic background [He et al., 2005] [Morency et al., 2005]. So it is now possible to incorporate these technologies into spoken dialog systems to make the system aware of the physical context, which in turn will result in more natural and effective conversations [Vinciarelli et al., 2009].

Multimodal information has been proven to be useful in dialog system design in driving both low level mechanics such as turn taking as well as high level adaptive strategies such as user attention regulation. Sciutti et al. [Sciutti et al., 2015] used gaze as an implicit signal for turn taking in a robotic teaching context. In [Yu et al., 2015a], a direction-giving robot used conversational strategies such as pause and restarts to regulate user's attention. Kousidis et al. [Kousidis et al., 2014b] used situated incremental speech synthesis that accommodates users' cognitive load in a in-car navigation task, which improved user experience but the task performance stays the same. However most multimodal systems suffer from not enough data for model training or evaluation, and they are not easy to access, since most of them require you to be physically co-present with the system. The community has been struggling with limited publicly available data for a long time. We propose a web-based multimodal dialog system, Multimodal HALEF, to tackle this issue. It integrates the video-enabled FreeSWITCH telephony server with an open-source distributed dialog system, HALEF. Multimodal HALEF records the remote user's video interaction and streams it to its servers by accessing the remote user's camera via a web browser. The HALEF source code is available at [1].

10.2 Foundational Frameworks

In this section, we describe the prior framework that Multimodal HALEF extends and builds upon. Figure 1 schematically depicts the overall architecture of the Multimodal HALEF framework.

10.2.1 The HALEF dialog framework

leverages different open-source components to form an SDS framework that is modular and industry-standard-compliant: Asterisk, a SIP (Session Initiation Protocol) and PSTN (Public Switched Telephone Network) compatible telephony server [Van Meggelen et al., 2007]; JVoiceXML,

[1] `https://sourceforge.net/projects/halef/`

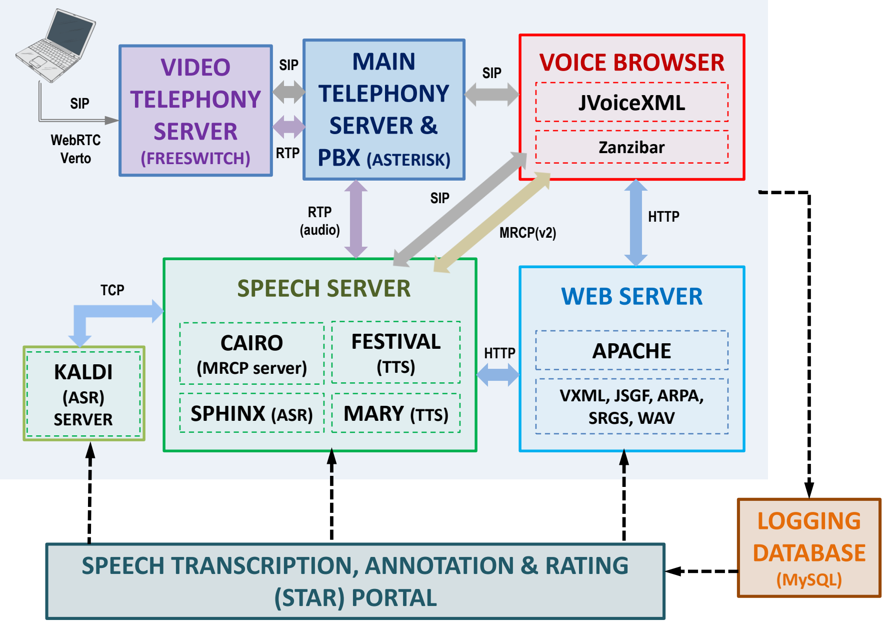


Figure 10.1: System architecture of the HALEF multimodal dialog system depicting the various modular open-source components.

an open-source voice browser that can process SIP traffic [Schnelle-Walka et al., 2013] via a voice browser interface called Zanzibar [Prylipko et al., 2011]; Cairo, an MRCP (Media Resource Control Protocol) speech server, which allows the voice browser to initiate SIP and RTP (Real-time Transport Protocol) connections between the speech server and the telephony server [Prylipko et al., 2011]; the Sphinx automatic speech recognizer [Lamere et al., 2003] and the Kaldi ASR system; Festival [Taylor et al., 1998] and Mary [Schroeder and Trouvain, 2003] text to speech synthesis engines; and an Apache Tomcat-based web server that can host dynamic VoiceXML (VXML) pages and serve media files such as grammars and audio files to the voice browser. OpenVXML allows designers to specify the dialog workflow as a flowchart, including details of specific grammar files to be used by the speech recognizer and text-to-speech prompts that need to be synthesized. In addition, dialog designers can insert "script" blocks of Javascript code into the workflow that can be used to perform simple processing steps, such as basic natural language understanding on the outputs of the speech recognition. In order to react to user's engagement, OpenVXML could retrieve the engagement score of the user utterance in real time and make decisions based on this. The entire workflow can be exported to a Web Archive (or WAR) application, which can then be deployed on a web server running Apache Tomcat that serves Voice XML (or VXML) documents.

Note that unlike a typical SDS, which consists of sequentially-connected modules for speech recognition, language understanding, dialog management, language generation and speech synthesis, in HALEF some of these are grouped together forming independent blocks which are hosted on different virtual machines in a distributed architecture. For further details on the individual blocks as well as design choices, please refer to [Ramanarayanan et al., 2015] [Mehrez et al., 2013].

10.2.2 FreeSWITCH

Specifically the 1.6 Video version[2], is a scalable open source cross-platform telephony framework designed to route and interconnect popular communication protocols using audio, video, text or any other form of media. It supports various communication technologies such as Skype, SIP and WebRTC. FreeSWITCH builds natively and runs standalone on several major operating systems, such as Linux, Windows and Mac OS. FreeSWITCH has been previously integrated with a dialog system in [Pappu and Rudnicky, 2013] that allows access via a web browser. However, this system can only handle audio input. FreeSWITCH is experimenter friendly. The experimenter can modify interaction settings, such as the number of people who can call in at any given time, whether to display the video of the user on the webpage, the resolution of the video, sampling rate of the audio, etc. FreeSWITCH also allows users to choose between different I/O devices for recording. They can switch between different microphones and cameras connected to their computers by selecting appropriate options on the web-based graphical user interface.

10.2.3 Engagement Module

The engagement module takes information from a visual processing service and produce an engagement score using the information. The visual processing service is a standalone Linux server for automatic head tracking via Cambridge Head Tracker [Baltrusaitis et al., 2012]. It can track a user's head movement and also estimate the head pose (e.g. 15 degrees to the left) given an input video with a clear view of the user's face. It also supports online feature extraction. We use a web socket to pass the captured raw image from FreeSWITCH to the Cambridge Head Tracker. It outputs the head tracking information to the engagement module, thus making multimodal human behavior information available in the dialog strategy selection module for decision making. This makes the dialog system aware of the user's behaviors so it can act accordingly. This is important since previous literature suggests that "computer systems with capabilities to sense

[2] https://freeswitch.org/confluence/display/FREESWITCH/FreeSWITCH+1.6+Video

agreement and attention that are able to adapt or respond to these signals in an appropriate way will likely be perceived as more natural, efficacious and trustworthy" [Vinciarelli et al., 2009]. Visual information has also been shown to be critical in assessing the mental states of the users in other systems as well [Yu et al., 2013a, Yu et al., 2015a]. So we include the visual processing service in our framework as well.

After the engagement module receives the real-time features, it performs three types of quantification simultaneously for engagement detection, visual engagement computed using visual features, acoustic engagement computed using audio features (currently not enabled) and verbal engagement computed using text features. The system processes the mean and variance of the head pose change to determine the frequency of users changing their head pose. It processes the mean and variance of the action unites that relates to smiles, in order to calculate the frequency of the user smiles. It processes the mean and variance of the gaze direction, to see the frequency of users shifting their gaze. We quantify these features in a time window (such as 500 ms) which is empirically determined based on the task. Then we combine these six features with a pre-set weights to obtain a score that represents the visual engagement of the user. Then the module outputs the visual engagement score to the database in order for the dialog manager (OpenVXML) to query later. We push the visual engagement score of the user every 500ms. In the future, we will integrate the audio information of the user to the engagement module as well in order to make the engagement prediction more accurate. The verbal engagement score is computed inside of the dialog manager as it can only be computed when the previous utterance is finished. We use features such as the length of the ASR to form the verbal engagement score. In the end, the dialog manger integrates all the modality-wise engagement score with a set of weights determined empirically. Once we have the engagement score, the dialog manager selects the conversation branch based on it as well as the SLU results. The weights used to combine different modality-wise engagement are determined by analyzing the previous data collected without the engagement model. They are different in different tasks. In the future, ideally we will make all the computation of the final engagement score within the engagement module, so as to make the system modular.

10.3 Framework Integration

We integrated FreeSWITCH into HALEF, so the dialog system have both video and audio as input from remote users through a webpage. Initially we planned to set up Asterisk to support video via WebRTC [3]. Unfortunately Asterisk does not support video yet. Thus we looked for

[3] http://www.webrtc.org

other alternatives and found that FreeSWITCH released the FreeSWITCH 1.6 Video version which supports video as of May 2015.

We followed the FreeSWITCH documentation to set up FreeSWITCH 1.6 on Debian 8 (Jesse). We decicde to use Verto [4] for achieving the in-browser video access capability. Verto is a FreeSWITCH module included the default FreeSWITCH configuration. Using Verto, rather than SIP/RTP over WebRTC, offered a few advantages. First it has a working webpage based conference call demo that we easily modified to allow a caller to interact with HALEF's audio dialogs. Second it is capable of recording audio and video as it is streamed to/from the caller. The alternative, SIP/RTP over WebRTC has more web-based clients such as sip.js [5], sipml5 [6], and jssip [7], but these required more work to get them to be functional on Chrome and Firefox. The problems with these clients are likely rooted in the fact that WebRTC and Web Audio are continuously evolving, experimental APIs. Therefore we choose not use them for our current framework.

The following is a description of how video and audio data flow to/from the Multimodal HALEF system. First the human user navigates to a custom webpage using their web browser and selects an extension to call. Different extensions are associated with different dialog system instances that have different task contents (for example, an interview for a job at a pizza restaurant,...). This webpage application is written in HTML, CSS, and javascript which leverages experimental web browser APIs (WebRTC and Web Audio). The Web Audio API enable access to the audio and video devices (e.g. cameras, microphones, speakers, etc.) via the web browser. The Verto protocol, which leverages WebRTC, is then used via javascript to send video/audio to FreeSWITCH and to receive audio from FreeSWITCH. The Verto teleconference webpage demo also uses the FreeSWITCH teleconference module. Overall, this is a teleconference that has two participants: HALEF and the user. When the call comes in from the user, HALEF starts the dialog with an audio prompt that flows out of HALEF system via Asterisk over SIP/RTP to FreeSWITCH. FreeSWITCH then sends the audio to the web browser over Verto. The user then gives a response to the system that is transported using Verto protocols to FreeSWITCH and then over SIP/RTP to HALEF's front end (Asterisk). During the teleconference, the user's video and audio interactions are continuously streamed and recorded on the FreeSWITCH server, while HALEF audio prompts are streamed to the user.

Within the HALEF system, once the interaction starts, Asterisk makes a SIP call to the Voice Browser (JVoiceXML) to get the specific dialog started based on the extension number dialed.

[4] https://freeswitch.org/confluence/display/FREESWITCH/mod_verto

[5] http://sipjs.com

[6] http://sipml5.org

[7] http://tryit.jssip.net

The Voice Browser then gets various resource files from the Web Server via HTTP that will allow it to control the call flow or call logic. For example, if the user says “yes” or “no” to a question then the Voice Browser will tell HALEF the next audio prompt to send accordingly, based on the dialog management instructions specified in the VXML application. The Voice Browser uses this information to communicate certain information to and from the Speech Servers. First, it tells the Speech Server to interact with Asterisk with regards to inbound and outbound audio (over SIP/RTP). Second, it tells the Speech Server to send the transcriptions of the audio to itself. And finally it sends instructions to receive text from the Speech Sever that will be synthesized to audio output to the user. The Voice Browser communicates with the Speech Server via MRCP and SIP.

There are other endpoints that are supported or likely can be supported with a little work. An endpoint is defined as a device that lives at the network edge (e.g. a telephone or a client application that acts like a telephone) that usually one user uses. HALEF (non-Multimodal) already supports audio-only dialogs with endpoints other than Verto. For example, we have used PSTN (public switched telephone network) endpoints, i.e., land line telephones and cell phones, to place calls to HALEF. We did this by using a PSTN/SIP proxy such as ipKall [8]. We also used SIP over WebRTC, and SIP/WebRTC clients like sipml5, jssip, etc to connect to HALEF directly thru Asterisk as well as via webrtc2sip `http://webrtc2sip.org/` to Asterisk. Note that we suggest using webrtc2sip, because it handles differences in implementations in Chrome and Firefox (in regards SRTP types and transcoding audio formats). As for Multimodal HALEF, we used Verto, but it is likely that with slightly modifications SIP/WebRTC clients could be supported.

The engagement module is linked with FreeSWITCH through sockets, as FreeSWITCH sends the raw image to the engagement module in real time via sockets. The engagement module doesn’t connect to HALEF directly, it pass the engagement information to the database first. Then HALEF would retrieve the engagement information whenever it needs from the database.

10.4 Supporting Modules

We introduced four supporting modules that assist researchers in conducting scientific research on interactive dialog systems and human behavior: a relational database that stores all call-log information, a survey webpage that collects users’ feedback of the interaction and pushes the information to the database, a web-based call viewing and rating portal.

[8] http://www.ipkall.com/

10.4.1 Database

We use the open-source database MySQL for our data warehousing purposes. All modules in the Multimodal HALEF connect to the database and write their log messages to it. We then post process this information with stored procedures into easily accessible views. Metadata extracted from the logs include information about the interactions such as the duration of the interaction, the audio/video recording file names, the caller IP, etc. Additionally, we store participants' survey data and expert rating information. All the modules connected to the database have been implemented such that all information will be available in the database as soon as the interaction, the survey, or the rating task is completed.

10.4.2 Participant Web-Survey

We created a survey webpage that participants were required to fill out upon completion of their video call. The survey was embedded along with the experimental instructions for the participant. Once the participant finishes the interaction, they are directed to fill out the survey regarding their interaction experience as well as some demographic information. Once the participant clicks the submit button the information is pushed to the appropriate table in the MySQL database.

10.4.3 STAR Portal

We developed an interactive rating portal, dubbed the Speech Transcription Annotation and Rating (STAR) portal, in order to facilitate annotation, transcription and rating of calls. It is mainly implemented using PHP and and the JavaScript framework jQuery. It has the advantage of accessing meta-data from the data warehouse and the audio/video data from the server as well. It provides a nice display of different information of the interaction. It also allows the experimenter to design rating questions that correspond to different components of the dialog framework or targeting the participant's performance, such as user speech input, system TTS, etc. It supports different types of questions, such as multiple choice questions, open questions, etc. Thus the rating task can not only be simple perception feedback to the interaction, but also detailed human transcription of the entire dialog. The tool also lets the experimenter manage raters by assigning different interactions for different users for the rating task. All of the information will be stored in the database for further analysis. The webpage supports playing both audio and video recordings of the collected interaction data.

10.5 Conclusion

We have designed and implemented an open-source web-based engagement reactive multimodal dialog framework, Reactive Multimodal HALEF, by integrating the existing dialog framework, HALEF, a video conferencing framework, FreeSWITCH and an engagement module. The framework also includes a database, a rating portal and a survey. It allows an experimenter to collect video and audio data of users interacting with the dialog system outside of a controlled lab environment, and largely reduces the cost and labor in collecting audio-visual data. The framework is designed to facilitate scientific research on how humans interact with reactive dialog systems, among other purposes.

Chapter 11

Situated Intelligent Interview Training Systems

In complex conversation tasks, people react to their interlocutor's state, such as uncertainty and engagement to improve conversation effectiveness [Forbes-Riley and Litman, 2009]. If a conversational system reacts to a user's state, would that lead to a better conversation experience? To test this hypothesis, we designed and implemented "Reactive Multimodal HALEF", a dialog system that tracks and reacts to a user's state, such as engagement, in real time. We designed and implemented a conversational job interview task based on the proposed framework. The system acts as an interviewer and reacts to user's disengagement in real-time with positive feedback strategies designed to re-engage the user in the job interview process. Experiments suggest that users speak more while interacting with the engagement-reactive version of the system as compared to a non-reactive version. Users also reported the former system as being more engaging and providing a better user experience.

11.1 Introduction and Related Work

Recently, multimodal sensing technologies such as face recognition, head tracking, etc. have improved. Those technologies are now robust enough to tolerate a fair amount of noise in the visual and acoustic background [He et al., 2005, Morency et al., 2005]. So it is now possible to incorporate these technologies into spoken dialog systems to make the system aware of the user's behavior and state, which in turn will result in more natural and effective conversations [Vinciarelli et al., 2009].

Multimodal information has been proven to be useful in dialog system design in driving both low level mechanics such as turn taking as well as high level mechanics such as conversation planning. Sciutti et al. [Sciutti et al., 2015] used gaze as an implicit signal for turn taking in a robotic teaching context. In [Yu et al., 2015a], a direction-giving robot used conversational strategies such as pause and restarts to regulate the user's attention. Kousidis et al. [Kousidis et al., 2014b] used situated incremental speech synthesis that accommodates users' cognitive load in a in-car navigation task, which improved user experience but the task performance stays the same. In Yu et al. [Yu et al., 2016b], a chatbot reacts to the user's disengagement by generating utterances that actively invite the user to continue the conversation.

Thus we propose Reactive Multimodal HALEF, a task-oriented dialog system framework that senses and reacts to a user's state, such as engagement, in real time. The framework is built on top of the HALEF open-source cloud-based standards-compliant multimodal dialog system framework [Ramanarayanan et al., 2016, Yu et al., 2016d]. Reactive Multimodal HALEF extracts multimodal features based on data that is streamed via the user's webcam and microphone in real time. Then the system uses these multimodal features, such as gaze and spoken word count to predict a user's state, such as engagement, using a pre-built machine learning model. Then the dialog manager takes the user's state into consideration in generating the system response. For example, the system could use some conversational strategies, such as positive feedback to react to the user's disengagement state.

With the advantage of being accessible via web-browser, Reactive Multimodal HALEF enables users interact with the system whenever and wherever in their comfortable environment, thus making the data collection and system evaluation process much easier and economical. The Reactive Multimodal HALEF is also open-source [1].

11.2 The Reactive Multimodal HALEF Framework

In this section, we describe the sub-components of the framework. Fig 11.1 schematically depicts the overall architecture of the HALEF framework.

11.2.1 The Multimodal HALEF Framework

FreeSWITCH, specifically versions above 1.6 [2], is a scalable, open-source and cross-platform telephony framework designed to route and interconnect popular communication protocols using audio, video, text or any other form of media. FreeSWITCH allows the experimenter to modify interaction settings, such as the number of people who can call in at any given time, whether to

[1] https://sourceforge.net/projects/halef/

[2] https://freeswitch.org/confluence/display/FREESWITCH/FreeSWITCH+1.6+Video

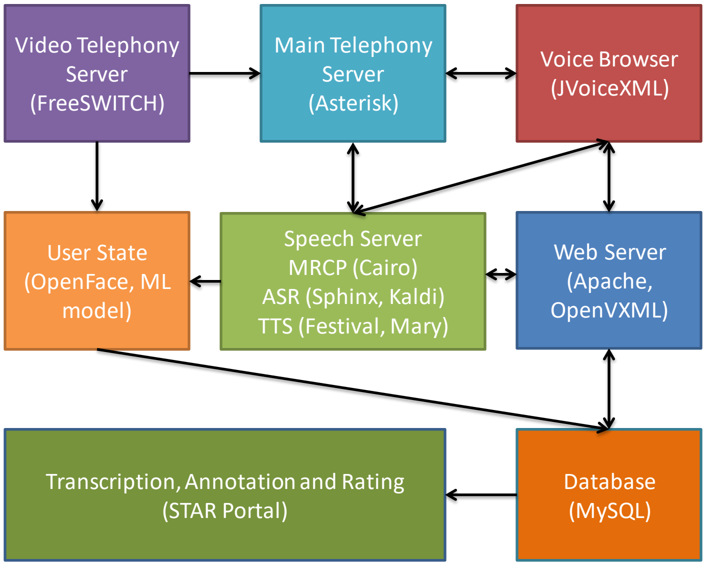

Video Telephony Server (FreeSWITCH)
Main Telephony Server (Asterisk)
Voice Browser (JVoiceXML)
User State (OpenFace, ML model)
Speech Server MRCP (Cairo) ASR (Sphinx, Kaldi) TTS (Festival, Mary)
Web Server (Apache, OpenVXML)
Transcription, Annotation and Rating (STAR Portal)
Database (MySQL)

Figure 11.1: System architecture of the Reactive Multimodal HALEF dialog system

display the video of the user on the webpage, the resolution of the video, sampling rate of the audio, etc. The FreeSWITCH Verto protocol also allows users to choose between different I/O devices for recording. They can switch between different microphones and cameras connected to their computers by selecting appropriate options on the web-based graphical user interface. We use FreeSWITCH Verto to connect user to HALEF via web browsers.

HALEF leverages different open-source components to form a SDS framework that is modular and industry-standard-compliant: Asterisk, a SIP (Session Initiation Protocol) and PSTN (Public Switched Telephone Network) compatible telephony server [Van Meggelen et al., 2007]; JVoiceXML, an open-source voice browser that can process SIP traffic [Schnelle-Walka et al., 2013]; Cairo, an MRCP (Media Resource Control Protocol) speech server, which allows the voice browser to request speech recognition, speech synthesis, audio playback and audio recording from the respective components; the Sphinx automatic speech recognizer [Lamere et al., 2003] and the Kaldi ASR system; Festival [Taylor et al., 1998] and Mary [Schroeder and Trouvain, 2003] text to speech synthesis engines; and an Apache Tomcat-based web server that can host dynamic VoiceXML (VXML) pages and serve media files such as grammars and audio files to the voice browser. OpenVXML allows designers to author the dialog workflow as a flowchart, including details of specific grammar files to be used by the speech recognizer and text-to-speech prompts that need to be synthesized. In addition, dialog designers can insert "script" blocks of Javascript code into the workflow that can be used to perform simple processing steps, such as creating HTTP requests to make use of natural language understanding web services on speech recognition output. In order to react to the user's engagement, these "script" blocks retrieve and act upon the engagement score of the user in real time. The entire workflow can be exported to a Web Archive (or WAR) application, which can then be deployed on an Apache Tomcat web server that serves Voice XML documents.

We use the open-source database MySQL for our data warehousing purposes. All modules in the Multimodal HALEF connect to the database and write their log messages into it. We then post-process this information with stored procedures into easily accessible views. Metadata extracted from the logs include information about the interactions such as the duration of the interaction, the audio/video recording file names, the caller IP, etc. Additionally, we store participants' survey data and expert rating information. All the modules connected to the database have been implemented such that all information will be available in the database as soon as the interaction, the survey, or the rating task is completed.

11.2.2 User State Module

The user state module is linked with FreeSWITCH via sockets to a standalone Linux server for automatic head tracking using OpenFace [Baltru et al., 2016]. The head tracker receives raw images from FreeSWITCH and performs tracking of the user's head movement, gaze direction and facial action units. Visual information has been shown to be critical in assessing the mental states of the users in other systems as well [Yu et al., 2013a]. So we include visual information in predicting the user state. The system uses the pre-trained machine learning model to predict the user's state. The user state module doesn't connect to HALEF directly, it passed the engagement information to the database first and then the application retrieves the user state information from the database. The application selects the conversation branch based on the user state information as well as the spoken language understanding results.

11.3 Example Application: Job Interview Training/Practice

In this conversation task, the system acts as the interviewer for a job in a pizza restaurant. The system first asks the user some basic personal information and then proposes two scenarios about conflicts that may happen in the workplace and asks the user how he/she would resolve them. We designed the task to assess non-native speakers' English conversational skills, pragmatic appropriateness of responses, and their ability to comprehend the stimulus materials and respond appropriately to questions posed during naturalistic conversational settings.

11.3.1 User Engagement Modeling

We first collected a set of data using a non-reactive multimodal interviewer version. We then used this dataset to build supervised machine learning models to predict engagement in real time, as well as a baseline for the reactive version of the system. We collected 200 video recordings in all from crowdsourced participants recruited via the Amazon Mechanical Turk platform. To train the engagement model we randomly selected 30 conversations which satisfied the following two quality criteria: (i) the face recognition system detects a face with high confidence (80%) throughout the interaction, and (ii) the automatic speech recognition output of the user utterances is not empty. There are in total 367 conversational exchanges in total over 30 conversations (note that we use the term conversational exchange here to denote a pair of one system turn and one user turn). We asked three experts to annotate user engagement for every conversational exchange based on the video and audio recordings. We adopted the engagement definition and annotation scheme introduced in [Yu et al., 2016b]. For the purposes of our study, we defined engagement as the degree to which users are willing to participate in the task along three dimensions – behaviorally (staying on task and following directions), emotionally (for instance, not being bored by the task) and cognitively (maximizing their cognitive abilities, including focused attention, memory, and creative thinking) [Whitehill et al., 2014]. Ratings were assigned on a 1-5 Likert scale ranging from very disengaged to very engaged. We collapsed 1-2 ratings into a "disengaged" label, and 3-5 into an "engaged" label, because the system is designed to react to a binary signal in the conversational flow. The threshold was chosen because we would like to only regulate the extreme cases in our task, in order to keep the conversation to be effective. For other tasks, we recommend setting the threshold as an experimental parameter that decided through user preference. There are in total three annotators involved and they had an inter-annotator Cohen's κ agreement value of 0.82 on average. In the modeling, we used the average label from all annotations as ground truth. Among all the conversation exchanges, 75% of the them are labeled as "engaged" and 25% are labeled as "disengaged". We would like to design policy that takes finer grained engagement scale in the future as well.

We used vision features to train a vision-based engagement predictor. We extracted the following vision features: head pose, gaze and facial action units. After the user state module receives the real-time features, it performs three types of quantification simultaneously for engagement detection [Baltru et al., 2016]. It processes the mean and variance of the head pose change to determine the frequency of users changing their head pose. It also extracts the mean and variance of the action units that relate to smiles, in order to calculate the frequency of user smiles. It further computes the mean and variance of the gaze direction, to capture how frequency users shift their gaze. These features are quantified per conversational exchange to form the feature set for engagement predictor training. We used a leave-one-conversation out cross validation and a Support Vector Machine with liner kernel to train the model. The result is 0.89 in F1 measure and the majority vote baseline is 0.72 in F1. The failure cases are mainly due to the system turn-taking errors, such as system interrupts the user which results a shorter user response, which then leads to a low engagement score in the verbal channel. The relative high value of the baseline is due to the skewness of the data, as there are more engaged conversational exchanges.

We also take the verbal information into account for engagement prediction. In this interview training task, there is a fixed number of conversation states, thus we calculated the mean value of the word count of all the conversations that are labeled as disengaged in each state. We use this value as the threshold to decide if the user is disengaged or not for each state. The verbal engagement score is computed over the ASR output as soon as the user utterance is finished.

In the testing process, we quantify vision features in a time window which is empirically determined by the engagement predictor's performance based on the task (we chose 2s, which happens to be the mean of the conversational exchange duration). Then we combine these six features with weights obtained from the machine learning model to obtain a score that represents the visual engagement of the user.

In the end, the dialog manger receives a score which is the sum of all the modality-wise engagement score with a set of weights determined empirically first. We then collected 54 conversations using the engagement-reactive interview training application. Among them we found 23 recordings that satisfied the two quality criteria mentioned earlier as well as an additional criterion of analyzing data from unique speakers. Experts rated the engagement at each conversational exchange where the system is required to make a dialog management decision based on the user's predicted engagement state.

We then used the collected data to retrain the weight we assigned for the vision modality and verbal modality with a linear regression model in Equation (1), here x_{i1} stands for the visual-engagement value, x_{i2} stands for the verbal-engagement value and y_i stands for the ground truth label. We performed a simple linear regression analysis which performed least-squares optimization of the following cost function:

$$min_{\alpha\beta}\sum_{i=1}^{n}(y_i - \alpha x_{i1} - \beta x_{i2})^2 \tag{11.1}$$

We estimated optimal regression coefficients of α= 0.63 and β = 0.37 and adjusted the weights in the model accordingly. We then collected another set of 50 conversations with adjusted weights. We found the F1 score for the trained engagement classifier was 0.86, a significant improvement over the majority vote baseline method of 0.74.

11.3.2 Conversational Strategy Design

The communication and education literature indicate a set of conversational strategies which are useful to improve user engagement, such as active participation, mention of shared experience [Wendler, 2014], positive feedback and encouragement [Lehman et al., 2012]. Particularly, in job interview literature, researchers find that infusing positive sentiments into the conversation could lead to more self-disclosure from interviewees. With this in mind, we designed a set of positive feedback strategies with respect to different dialog states. For example in one dialog state, we asked about the interviewee's previous experience and provide positive feedback about his experience, for example saying: "It is good to know that you have some previous experience about the job."

11.3.3 Reactive Policy

We implemented a local greedy policy to react to the user's disengagement in this interview training task. Once the dialog manager receives the signal from the end-pointing module reporting that the user has finished the turn, it queries the most recent engagement score from the database. If the user is sensed as disengaged, the positive strategy that is designed with respect to that conversational state will trigger, otherwise the conversation goes into next dialog state.

11.3.4 Results

We asked the user to fill out a survey after interacting with the system. We asked them how engaged they feel overall during the interaction in a 1-5 Likert scale and their overall conversation experience overall as well in a 1-5 Likert scale. We compared the user responses of the 30 conversations that are collected using the non-reactive interview training method and the 32 conversations that are collected using the engagement-reactive version, and found that the engagement-reactive system receives statistically higher overall ratings from the users in terms of both overall engagement and user experience (see Fig 11.2 for details).

Though the engagement-reactive version has more system utterances than the non-reactive one, the extra utterances are all statements(e.g. I think the manager would do that.) instead of questions, so no extra user utterances were elicited. Thus we calculated the number of unique tokens of all the uses' utterances based on the ASR output and found that users who interacted with the engagement-reactive version expressed significantly more information than users who interacted with the non-reactive version (see Fig 2 for details).

We also found there are three users who repeatedly interacted with the engagement-reactive interviewer. We found that their assessed average engagement improved (from 2.3 to 4.0) after interacting with the system several times. This indicates that by interacting with our system the users are able to improve their ability in engaging in a job interview conversation.

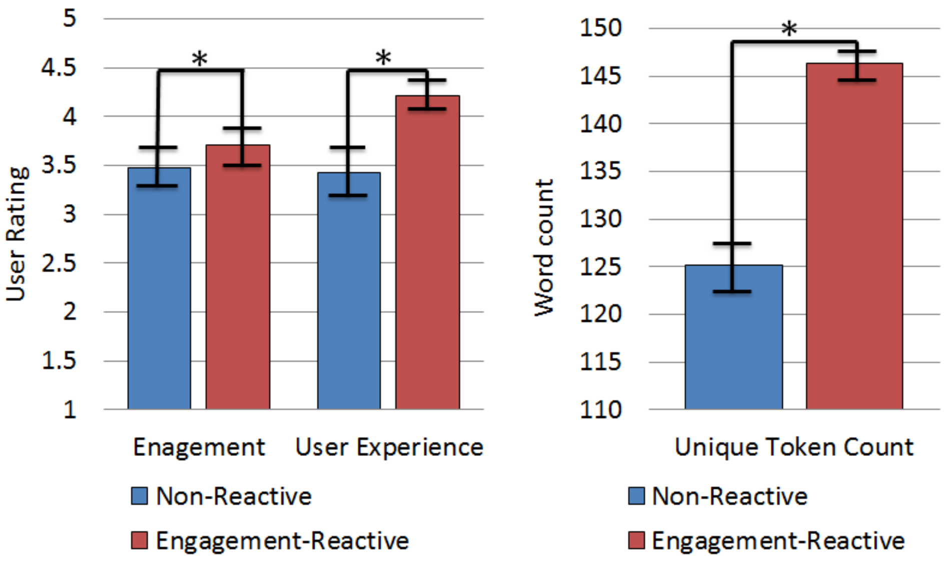


Figure 11.2: Experiment results between non-reactive and engagement-reactive job interviewer

11.4 Conclusion

We proposed and implemented a real-time user-reactive system framework for task-oriented conversations. We implemented an example application based on the framework, engagement-reactive interview training task. From the data collected using both non-reactive version and engagement-reactive version of the interview training task, we found that the engagement-reactive version is rated as more engaging and having better user experience in the job interview training setting compared to a non-reactive system. In addition, we do find that our job interview training task has helped some users improving their ability to engage in a job interview conversation.

In the future, we wish to design and implement other reactive systems that are able to tackle other tasks, such as language ability training. We also wish to design better conversation policy to take context into consideration in conversation flow planning. We will also integrate audio information of the user to the engagement module in order to make the engagement prediction more accurate.

Chapter 12

Conversational Systems that Interleave Task and Non-Task Content

Task-oriented dialog systems have been applied in various tasks, such as automated personal assistants, customer service providers and tutors. These systems work well when users have clear and explicit intentions that are well-aligned to the systems' capabilities. However, they fail if users intentions are not explicit. To address this shortcoming, we propose a framework to interleave non-task content (i.e. everyday social conversation) into task conversations. When the task content fails, the system can still keep the user engaged with the non-task content. We trained a policy using reinforcement learning algorithms to promote long-turn conversation coherence and consistency, so that the system can have smooth transitions between task and non-task content. To test the effectiveness of the proposed framework, we developed a movie promotion dialog system. Experiments with human users indicate that a system that interleaves social and task content achieves a better task success rate and is also rated as more engaging compared to a pure task-oriented system.

12.1 Introduction

The most familiar communication setting for people is talking to another human. Therefore, normal users would transfer their human-human behavior patterns and expectations to interactions with a system. For example, though users quickly learned that Microsoft Cortana (a personal assistant) could not handle social content, 30% of the total user utterances addressing it are social content [Jiang et al., 2015]. Therefore, one possible way to improve conversational system performance is to imitate human behaviors. Countless observations suggest that human conversations usually interleave social content with task content [Schegloff, 1968]. For example, we usually open a conversation with "How are you doing?"; we also divert to social topics during meetings; and we would most likely end our workday conversations with chitchat of our weekend plans. However, traditional conversational systems are mainly task-oriented. These systems can complete tasks such as booking airline tickets [Zue et al., 1994], searching for bus information [Raux et al., 2005], and making restaurant reservations [Jurcicek et al., 2011]. These systems do not involve social content mainly because the task is relatively simple and the user intention and system capability are well calibrated.

To move our current systems to tackle tasks that are more complex and especially those where most users do not have clear intentions, we propose a dialog framework to fuse task and non-task conversation content. To achieve the content transition smoothness, we trained dialog policies using reinforcement learning algorithms. We built an example dialog system, a movie promotion system that promotes a specific movie according to users' interests and uses social conversation to engage users to complete the task. There are several types of audience research conducted by film distributors in connection with domestic theatrical releases [Martin, 2002]. Such audience research can cost $1 million per movie, especially when scores of TV advertisements are tested and re-tested. Therefore, we argue that having conversational system to elicit audience information voluntarily to replace paid surveys would reduce the cost and improve the survey quality.

We published the source code of the software implementation of the framework, an example movie promotion system, and the conversation data collected with human users.[1] The framework is general and applicable in different domains, such as political surveying, language learning, and public health education. The theoretical framework and the software implementation enables researchers and developers to build example dialog systems in different domains, hopefully leading to big impact beyond discourse and dialog research.

12.2 Related Work

Current task-oriented dialog systems focus on completing a task together with the user. They can perform bus information search [Raux et al., 2005], flight booking [Zue et al., 1994], direction giving [Yu et al., 2015a], etc. However, these systems can only focus on one task at a time. The famous personal assistants, such as Apple's Siri are composed of many of these single-task systems. These single-task systems' underlying mechanisms are mainly frame-based or agenda-based [Rudnicky and Xu, 1999]. The architecture of traditional dialog systems is slot-filling, which pre-defines the structure of a dialog state as a set of slots to be filled during the

[1] omitted for anonymity purposes, see supplementary materials

conversation. For an airline booking system, an example slot is “destination city”. An example corresponding system utterance generated from that slot is ”Which city are you flying to?” Recently researchers have also started to look into end-to-end learning for task-oriented systems. Though the progress is still preliminary [Bordes and Weston, 2016], the premise of having a learning method that generalizes across domains is appealing.

Differing from task-oriented systems, non-task-oriented systems do not have a stated goal to work towards. Nevertheless, they are useful for social relationship bonding and have many other use cases, such as keeping elderly people company [Higashinaka et al., 2014], facilitating language learning [Jia and Ruan, 2008], and simply entertaining users [Yu et al., 2016b]. Because non-task systems do not have a goal, they do not have a set of restricted states or slots to follow. A variety of methods were therefore proposed to generate responses for them, such as machine translation [Ritter et al., 2011], retrieval-based response selection [Banchs and Li, 2012], and sequence-to-sequence models with different structures, such as, vanilla recurrent neural networks [Vinyals and Le, 2015], hierarchical neural models [Serban et al., 2015], and memory neural networks [Dodge et al., 2015].

However, there is no research on combining these two types of dialog systems so far. Therefore, our work is the first attempt to create a framework that combines these two types of conversations in a natural and smooth manner for the purpose of improving conversation task success and user engagement. Such a framework is especially useful to handle users who do not have explicit intentions.

To combine these two types of conversation systems smoothly, we trained a response selection policy with reinforcement learning algorithms. Reinforcement learning algorithms have been used in traditional task-oriented systems to track dialog states [Williams and Young, 2007]. They have also been used in non-task oriented systems. The Q-learning method was used to choose among a set of statistical templates and several neural model generated responses in [Yu et al., 2016g], while the policy gradient method was used in [Li et al., 2016b]. Different from these pure task or pure non-task systems, we applied reinforcement learning algorithms to train policies that choose among task and non-task candidate responses to optimize towards a coherent, consistent and informative conversation with respect to different users.

12.3 Framework Description

The framework has four major components: a *language understanding module*, a *task response generator*, a *non-task response generator* and a *response selection policy*. Figure 12.1 shows the information flow among these components. A user utterance is sent to both the language

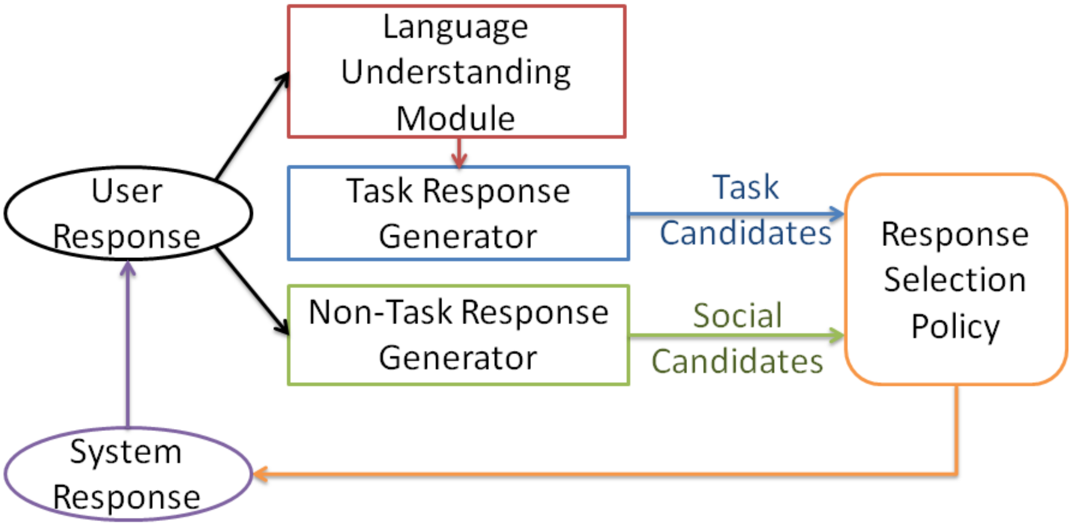


Figure 12.1: Framework Architecture. A user utterance is sent to both a language understanding module and a non-task response generator. The understanding model then extracts useful information to help a task response generator to produce task-oriented candidates. Simultaneously, the non-task response generator also produces non-task candidates. Finally, a response selection policy selects among all the candidates to produce a system response.

understanding module and the non-task response generator. The understanding model then extracts useful information to help the task response generator to produce task-oriented candidates. Simultaneously, the non-task response generator also produces non-task candidates. Finally, the *response selection policy* selects among all the candidates to produce a system response. We will discuss each component in details below.

The language understanding module extracts information required by the task from user responses, and sends the information to the task response generator. There are many approaches to perform language understanding. A rule-based keyword-matching method is sufficient for simple tasks [He and Young, 1998], while learning-based methods perform better in complex tasks. The understanding task can be formulated as different learning problems: for example a supervised classification problem that can be solved by algorithms, such as Support Vector Machines [Williams et al., 2015]; or a sequence labeling problem that can be solved by algorithms, such as recurrent neural networks [Mesnil et al., 2013].

The task response generator produces a list of task response candidates considering the information passed from the language understanding module. The most common method to generate these candidates is pre-defined statistical templates that are used in traditional slot-filling task oriented systems [Rambow et al., 2001]. Each template intends to elicit information required by the task. While the non-task response generator produces a list of social response candidates.

This component could be implemented by various methods mentioned in the related work, such as machine translation, sequence-to-sequence, keyword retrieval, or an ensemble of these methods. The non-task content intends to make the conversation more coherent and engaging, thus keeping users in the conversation. Therefore, users would have more chance to complete the task.

The response selection policy sequentially chooses among all the response candidates (non-task or task candidates) to optimize towards natural and engaging interactions. Different reinforcement learning algorithms could be applied to train the policy, such as Q-learning [Sutton and Barto, 1998], SARSA [Sutton and Barto, 1998], and policy gradient [Sutton and Barto, 1998]. Different reinforcement learning algorithms would be preferred with respect to other framework components' implementation. For instance, if the number of response candidates are limited and the dialog states can be represented in tabular forms, then Q-learning is sufficient. While if the state representation is continuous, then the policy gradient method is preferred. In the next section, we will use a movie promotion task as an example to describe example algorithms in details.

12.4 Example Dialog Task: Movie Promotion

The movie promotion task has a goal to promote a movie taking account of users' preferences. To imitate human-human conversation, the framework suggests to open conversations with a non-task conversation topic related to the task. Therefore, the movie promotion system starts a conversation saying: "Hello, I really like movies. How about we talk about movies?" We then describe the implementation details of each component.

12.4.1 Language Understanding Module

Because the user responses' sentence structure is relatively simple and the information to extracted is mainly 'yes/no' and named entities, we used a shallow parser to provide features, and then a set of pre-designed rules for each task template. For example, for the response to the "If Seen Movie?" template, we simply used a key-word matching rule to classify the utterance to 'yes' or 'no' categories. The extracted information is then fed to the task response generator. The collected information is also useful for task analysis.

12.4.2 The Task Response Generator

The task response generator produces response candidates using a sets of pre-defined language generation templates considering the information received from the language understanding module. We designed the following eight templates to approach the goal of promoting a movie, such as “Captain America: Civil War”.

- **Elicit movie type**: The system elicits the user’s preferred movie type, e.g. “Do you like superhero movies or Disney movies?”.
- **Introduce favorite superhero**: The system expresses its favorite superhero is Captain America, in order to lead to the movie for promotion, e.g. “My favorite superhero is Captain America.”
- **Ground on superhero**: We crawled wiki-webpages to build a superhero knowledge database that includes all the superheroes’ details, such as real name, eye colors, origin, etc. If the user mentions any superhero, the system will talk about some attributes of that superhero, e.g. “I really like Iron Man’s blue eyes.”
- **Discuss relevant movie**: The system talks about a relevant movie the user mentioned before, e.g. “I really like the first Avenger movie, have you seen it before?”
- **Discuss movie detail**: The system further elaborates on the details of the mentioned relevant movie, e.g.“I really liked the first Avenger movie. When Iron Man came back alive, I cried.”
- **Saw the movie**: The system asks the user if they saw the movie for promotion, e.g.“Have you seen the new superhero movie, ‘Captain America: Civil War’?”
- **Promote the movie**: The system promotes the intended movie. e.g. “One of my friends just saw ‘Captain America: Civil War’. He told me it is a really nice movie, much better than the previous Captain America movie.”
- **Invite to the movie**: The system suggests to see the promoted movie together, e.g. “Do you want to see Captain America: Civil War together?”

12.4.3 Non-Task Response Generator

The non-task generator provides three types of candidate responses via three methods described below.

- A keyword retrieval method trained on a CNN interview corpus [Yu et al., 2015b].
- A skip-thought vector model [Kiros et al., 2015] trained on the Movie Subtitle dataset

[Lison and Tiedemann, 2016]

- A set of conversation strategies that generated via statistical templates that emulate human-human conversation strategies, in order to foster user coordination, understanding and adaptation. In particular, we designed three types of conversation strategies following [Yu et al., 2016g].
  - *Active participation strategies* engage users by actively contributing to the conversation, such as asking more information on the current topic [Wendler, 2014].
  - *Grounding strategies* assist open-domain natural language understanding. Grounding strategies were automatically synthesized via leveraging knowledge-base (e.g. Google Knowledge Graph) information and natural language processing algorithms, such as named-entity detection and statistical language generation.
  - *Personalized strategies* support adaptions to users by leveraging automatically extracted information from individual user's conversation history. An example personalized strategy is to suggest talking more about a certain topic knowing that the user was previously engaged in that topic.

12.4.4 Response Selection Policy

The response selection policy is designed to select candidates provided by the two response generators. Conversation processes can be modeled as Markov Decision Processes (MDPs) [Williams and Young, 2007]. Therefore, we used reinforcement learning algorithms to train a policy that optimizes towards conversations with long-term coherence, consistency, variety, and continuity. Specifically, we used Q-learning, a model-free reinforcement learning algorithm. Because it handles discrete states well and learns a Q table that supports a model that makes both debugging and interpretation easier. Another advantage of Q-learning is that it also makes encoding expert knowledge easier, as it is a model-free algorithm. By encoding expert knowledge, the search space can be reduced, thus making the algorithm converge faster. In all, Q-learning is a good choice with respect to the implementation choice of other components of the system.

In a reinforcement learning setting, we formulate the problem as $(\mathbb{S}, \mathbb{A}, \mathbb{R}, \gamma, \alpha)$, where $\mathbb{S}$ is a set of states that represents the system's environment, in this case the conversation history so far. $\mathbb{A}$ is a set of actions available per state. In our setting, the actions are strategies available. By performing an action, the agent can move from one state to another. Executing an action in a specific state provides the agent with a reward (a numerical score), $\mathbb{R}(s, a)$. The goal of the agent is to maximize its total reward. It does this by learning which action is optimal to take for each state. The action that is optimal for each state is the action that has the highest long-term reward. This reward is a weighted sum of the expected values of the rewards of all future steps starting from the current state, where the discount factor $\gamma \in (0, 1)$ trades off the importance of sooner versus later rewards. γ may also be interpreted as the likelihood to succeed (or survive) at every step. The algorithm therefore has a function that calculates the quantity of a state-action combination, $\mathbb{Q} : \mathbb{S} \times \mathbb{A} \rightarrow \mathbb{R}$. The core of the algorithm is a simple value iteration update. It assumes the old value and makes a correction based on the new information at each time step, *t*. The critical part of the modeling is to design appropriate states, actions and a corresponding reward function.

State and Action Design It is difficult to design states for conversational systems, as one slight change from a user response may lead to a completely different conversation. We reduced the state space by incorporating extra knowledge, the statistics obtained from conversational data analysis. Following [Yu et al., 2016g], we include features: turn index, number of times each strategy executed, sentiment polarity of all previous utterances, coherence confidence of the response, and most recently used strategy. We constructed the reward table based on the statistics provided in [Yu et al., 2016g] . We utilized expert knowledge to construct rules to constrain the reward table. For example, if certain strategies have been used before, then the reward of using it again immediately is heavily penalized. Please see a detailed list of constrains in the appendix. These constraints may result in less optimal solutions, but reduce the state and action search space considerably. The actions are simply all the response candidates produced by both task and non-task generators.

Reward Function Design We designed the reward function to be a linear combination of four metrics: *turn-level appropriateness (App)*, *conversational depth (ConvDepth)*, *information gain (InfoGain)*, and *conversation length (ConvLen)*. Except for the first metric which is an immediate reward, all the others are delayed rewards. We first initiated weights associate to each metric manually, and refine them in training later. One difficulty for interactive tasks is that during training, we can not interrupt the interaction flow by asking users to give turn-by-turn immediate feedback. To solve this problem, we used pre-trained predictors to approximate these metrics, which is similar to inverse reinforcement learning. We will describe all the metrics along with methods to approximate them in the details below.

- **Turn-level appropriateness (App)**: reflects the coherence of the system's response in each conversational turn. For later experiments, we adopted the same annotation scheme in [Yu et al., 2016e] and used the appropriateness predictor provided in [Yu et al., 2016g], which trained on 1256 annotated turns. The performance of the automatic appropriateness detector is 0.73 in accuracy (majority vote is 0.5 in accuracy).

- **Conversation depth (ConvDepth)**: reflects the number of consecutive utterances that share the same topic. We followed the same annotation scheme in [Yu et al., 2016g]: labeling conversations having ten or more consecutive turns to be deep and others as shallow. We used the predictor trained on 100 conversations from [Yu et al., 2016g]. It has 72.7% in accuracy, while the majority vote baseline accuracy is 63.6%.
- **Information gain (InfoGain)**: reflects the number of unique words that are introduced into the conversation from both the system and the user. We believe that the more information the conversation has, the better the conversational quality is. This metric is calculated automatically by counting the number of unique words after the utterance is tokenized.
- **Conversation Length (ConvLen)**: reflects how long the user want to stay in the conversation. We approximated it by the number of turns in the overall conversation. The assumption is that the more the users want to interact with the system, the better the system is.

12.5 Experiments

We built three systems for the movie promotion task.

- ***Task-Global***: This does not have a non-task response generator and therefore can only output task responses. Its response selection policy is trained with the entire conversation history.
- ***Mix-Local***: This has both the task and non-task generators, and a response selection policy considers three previous turns as interaction history in the reinforcement learning algorithm.
- ***Mix-Global***: This has both the task and non-task generators, and a response selection policy trained with the entire interaction history in the reinforcement learning algorithm.

We used another chatbot, A.L.I.C.E.[2], which is powered by rules, as a user simulator to train the response selection policy for all systems. During training, we restart conversations if the user simulator repeats the same utterance. It took 200, 1000, and 8000 conversations respectively for the Task-Global, Mix-Local, and Mix-Global systems to converge.

Besides testing the system with the user simulator, we also recruited human users on crowdsourcing platforms (Amazon Mechanical Turk and Crowd Flower). We recruited crowd workers that are located in the U.S. and have a previous task approval rate that is higher than 95%. We

[2] http://alice.pandorabots.com/

asked them to interact with the system for as long as they want. They were only allowed to interact with one of the three systems once, thus preventing them exploiting the task. Within two days, 150 crowd workers participated and produced 50 conversations for each system. The users also reported their gender and age range after the interaction. In addition, we asked them to rate how engaged they were throughout the conversation in a 1-5 Likert scale (a higher score indicates more engagement). Table 12.1 shows an example conversation produced by the Mix-Global system. We consider the task to be successful if the user received all the task responses. Apart from the task success rate, the self-reported user engagement score is also included for evaluation.

12.6 Results and Analysis

From experiments, we found that the system that interleaves non-task content and considers the entire conversation history in policy training performs the best in both simulated and real-world settings. We discuss detailed statistics below:

Finding 1: Involving non-task content makes the user more engaged and at the same time increases the task success rate. Figure 12.2 shows that the Mix-Global system outperformed the Task-Global system with respect to both the task success rate and the user self-reported engagement rating with statistical significance (t-test, $p < 0.001$). The average user self-reported engagement is 2.3 (SD=0.42) for the Task-Global system, and 4.4 (SD=0.21) for the Mix-Global system. The task success rate of the Mix-Global system outperformed (78%) the Task-Global system (23%) with a big margin. The good performance comes from the fact that in the Mix-Global system, the non-task responses handled the user utterances that task responses could not handle. Therefore the user could remain in the conversation, providing the system with more opportunities to complete the task.

Table 12.1 shows an example conversation between the Mix-Global system and a human user. Utterances 7, 9, and 11 are system non-task responses. They were selected by the policy for yielding coherent and consistent conversations. For example, utterance 9 “What I meant to say was, what is it that you hate?” was chosen to promote topic consistency and at the same time to preserve local coherence. Non-task content also contributes to content variety. Overall, the policy trained with reinforcement learning algorithms makes the transition between non-task and task smoother. We suspect that is the major reason that users find the system that interleaves non-task and task content more engaging . However, the Mix-Global system still fails to engage some users who do not have any interest in superhero movies. For example, one user replied:

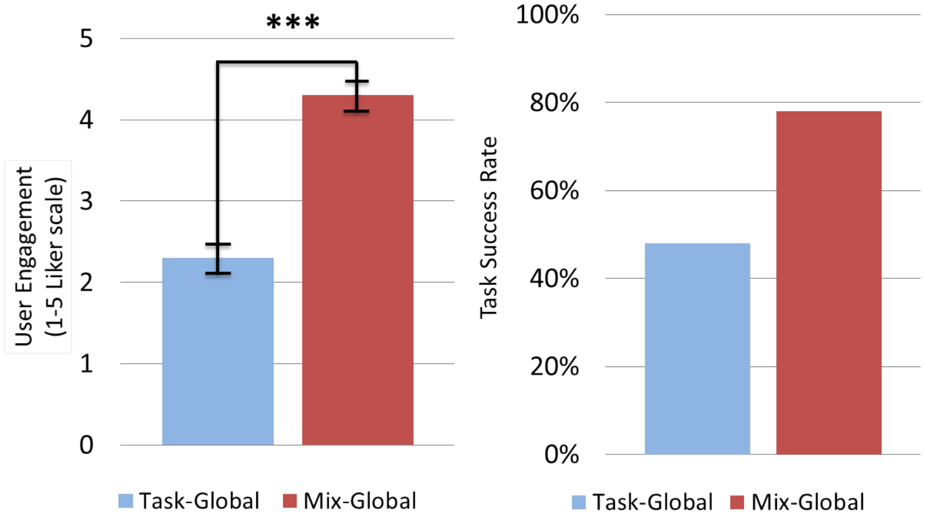


Figure 12.2: The Mix-Global system outperformed the Task-Global system on both user engagement and task success.

“No, I haven’t seen [Captain America: Civil War], I am not a stupid teenager or a stupid robot.” and left the conversation early.

Finding 2: Incorporating longer interaction history engages users better and improves task success rate. Figure 12.3 shows that the user self-reported engagement of the Mix-Local system (4.0 (SD=0.32)) outperformed the Mix-Global system (4.4 (SD=0.21)) with a moderate statistical significance ($p<0.05$). The Mix-Global system also outperformed the Mix-Local system on task completion rate (70% VS. 78%). Therefore, statistics suggest that a system considers the entire conversation history in planning is more engaging and completes the task more often, compared with a system that only takes the previous three utterance history. The small margin is caused by the fact that the task conversation is not very long (8 responses in total). Therefore, we expect a stronger performance impact for tasks that are more complex that require longer conversations.

Finding 3: We also found that the system interleaves social content and is powered by a policy trained with the entire conversation history performs best in all the individual metrics in the reward function in both simulated and real-world (interacting with human users) settings (see Table 2 and 3 for details). We follow the convention in [Yu et al., 2016g] to consider conversations that have 10 or more consistent turns on the same topic to be deep conversations. Because the Task-Global system only produces conversations with 8 turns, we do not compare it on the

1 *Non-Task*: Hello, I really like movies. How about we talk about movies?
2 *User*: I like watching movies too.
3 *Task*: Do you like superhero movies or Disney movies?
4 *User*: I like superhero movies.
5 *Task*: My favorite superhero is Captain America.
6 *User*: I like Spider-man.
7 *Non-Task*: Do you watch them with your kids?
8 *User*: I don't have any children.
9 *Non-Task*: What I meant to say was, what is it that you hate?
10 *User*: I hated the last Fantastic Four movie.
11 *Non-Task*: Are you talking about Fantastic Four, the 2005 film.
12 *User*: Yes. I am.
12 *Task*: I really like the first Avenger movie, have you seen it before?

Table 12.1: An example conversation between a human user and the Mix-Global system. Utterances 7, 9, and 11 are social responses that selected by the policy to keep the conversation coherent, consistent and informative.

conversation depth metric. Among all the metrics, we found response coherence (App) improved most by incorporation social content, indicating the smooth transition has the biggest impact on the overall performance improvement. However, we also found that all systems' performance is always lower in the real-world setting compared to the simulated setting. Probably because the user simulator is limited and it was powered by an extensive set of rules.

System	App	ConvDepth	InfoGain	ConvLen
Task-Global	32.5%	NA	34.5	5.7
Mix-Local	76.3%	77%	52.4	13.7
Mix-Global	80.1%	88%	66.8	17.4

Table 12.2: The Mix-Global system performed the best when interacting with a simulated user.

System	App	ConvDepth	InfoGain	ConvLen
Task-Global	31.3%	NA	39.3	5.3
Mix-Local	73.0%	71%	62.4	13.0
Mix-Global	76.7%	79%	67.2	15.8

Table 12.3: The Mix-Global system performed the best when interacting with human users.

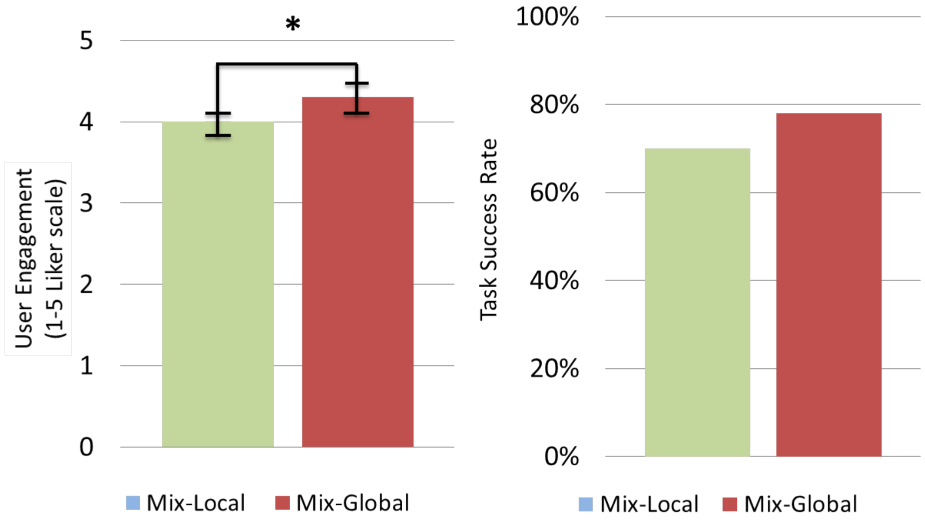


Figure 12.3: The Mix-Global system outperformed the Mix-Local system on both user engagement and task success.

12.7 Movie Promotion Task Data Analysis

To validate the usability of the example movie promotion system, we conduct a brief analysis on the information collected by the system. We found three interesting suggestive phenomena that would be of interest to the film industry, though the user sample pool is relatively small (150 participants) and biased (crowd workers located in the U.S.).

Participants prefer superhero movies over Disney movies. We found that 42% of participants preferred superhero movies, 22% of participants preferred Disney movies, and 36% of participants liked both or neither. Overall, superhero movies are more popular than Disney movies among our participants. One possible explanation is that we surveyed only U.S. residents.

Men prefer superhero movies. Among the 150 participants, 102 (57 men) of them were asked if seen the promoted movie and 42 (41%) of them did. Among them 57.9% of men saw the film, while only 20.0% of women did. 90 participants were asked if they would like to see the promoted movie, and 80% of them said yes. Men were more likely to go along with the invitation than women (77.8% of men and 68.3% of women said yes). Both findings indicate that men compared to women are more interested in the promoted movie.

Twenty-year-olds like superhero movies most. We separate the participants into five different age groups: below 20, 20-30, 30-40, 40-50, and 50 above. People who had seen the promoted movie spread across all age groups, and mostly concentrate in the 20-30 age group (27 out of 42). Even though our participants are mostly in this age group (78 out of 150), we still found that there are more participants in their 20s who have seen the promoted movie compared to other groups with statistical significance (t-test $p < 0.05$). In addition, compared to participants in other age groups, participants in their 20s are more likely to accept the invitation to see the promoted movie together (39 out of 42).

Statistics reported by the Motion Picture Association of America shows a similar trend. They found that females accounted for only 42% of the audience for “Captain America: The Winter Soldier”, and 18-25 age group attended movie most per capita is the 18-25 in 2015 [MPAA, 2015].

12.8 Conclusion and Future Work

We proposed a framework to develop conversational systems that interleave task content with non-task content in a natural manner, in order to improve conversation task success and user engagement. The framework is mainly powered by a statistical policy that selects among task and non-task candidate responses to optimize towards long-term conversation effectiveness. This framework is general enough to apply to many tasks, and is especially useful for tasks where users often do not have any concrete intentions. We also designed an example system under the framework that promotes movies to validate the framework. Through experiments conducted with both a simulator and real human users, we found the system that interleaves non-task content achieved better task success and user engagement. We also validated the systems usage by finding some interesting phenomena from the data collected that correlates with statistics reported by theatrical associations. Although our participant sample size is small, we believe these statistics could still help movie makers understand their markets better.

While the example task centers on movie promotion, there is nothing specific to the task design which could not be transferred immediately to other domains such as product recommendations, public health education, customer services and political surveys. Future work would further investigate different types of representations for reinforcement learning algorithms so as to make the framework generalize well across domains.

Chapter 13

Conclusion and Future Work

With the development of open-domain ASR and SLU components, we are able to approach the next generation dialog systems, that could handle multiple complex tasks, that could be adapt to uses to provide personal experience. One possible architecture that suits this goal best is the non-state-based system architecture. In contrast to the state-based conversational systems, non-state-based systems do not have a pre-defined agenda to guide and restrict the conversation. With its flexibility in conversation flow, none-state-based systems have potential to handle multiple tasks in one conversation and to cooperate with users in real time, in order to provide a complete, proactive and personalized user experience. The proposed SI framework is most effective in these state-less conversational systems due to the joint effect of its situation awareness, conversational strategy and global planning components. First, through modeling different situation contexts, such as user engagement with respect to the user's cultural background and the task content, the system can prepare itself to coordinate with users. Conversational strategy are a set of system actions with special functions designed by leveraging social linguistic knowledge and knowledge bases. Different types of conversational strategies have different capabilities, such as coordinating with situated contexts, assisting language understanding, providing personal experience and proactive making suggestions. The global planning policy takes conversation history into consideration to select among conversational strategies in order to optimize long-term system performance. The global planning strategy enables non-state-based conversational systems to transit smoothly from different conversion topics or tasks.

Apart from the algorithms we designed, we also applied them to build end-to-end systems that users could interact with and benefit from the interaction. In particular, we build an example system in varies conversation type: non-task-oriented, task-oriented and conversations interleaves task and non-task content. In non-task-oriented systems, we built a social chatbot to talk with users about everyday topics. A chatbot with the SI framework is rated as more engaging than one without and is able to elicit longer conversations as well. We also designed a task-oriented system, a job interview training system, in which the system acts like an interviewer to train users for interviews. We applied the SI framework to it as well with adjustment, as it is a state-based system. The version with the SI framework is more engaging and elicited more user self-disclosure than the version without the framework. We have also observed that users' interview performance improved after they interacted with the system repeatedly. Finally, we proposed a new type of systems that interleaves task content with social content. These systems could elicit information without users being biased by the task purpose. It is especially useful for social science studies, in which the study intent was often hidden as to avoid biases. We built a film-promoting system that is able to collect user's opinion towards a film and promote it to people who may have the interest. The SI framework is especially useful in these systems, as it managers to transition from task and non-task parts of the conversation smoothly. The film-promoter that has a global planning strategy in deciding the transition in and out of task contents was rated more engaging than a system that had a rule for the transition.

To sum up, this thesis pioneered the work of empowering situation intelligent to interactive systems. We developed both theoretical concepts and practical algorithms for the situated intelligent framework for different types of conversational systems. We also implemented, evaluated and deployed example end-to-end systems for each conversation type and found the SI framework is useful in improving system performance and engaging user.

In the future, I plan to further improve interactive systems' effectiveness and naturalness by incorporating *human-in-the-loop* technologies. I also want to apply the SI framework to other interdisciplinary areas for social good.

Interactive Machine Learning. Modeling human-machine interaction is difficult because user actions depend on interaction history, and a small change could lead to an entirely different action sequence. Therefore, I propose to use *interactive machine learning*, which iteratively leverages human feedback to improve model performance at a faster pace [Williams et al., 2015], to make interaction modeling efficient. The performance of interactive learning hinges on good user interface design to enable people to provide more effective and accurate feedback, and powerful online learning methods to incorporate human feedback effectively.

Human-Machine Hybrid Systems. Trust is essential in high risk areas, such as health care and education. Therefore, I propose to design hybrid systems that combine automatic systems and crowd-powered systems [Lasecki et al., 2013] to make interactive systems more accountable. Hybrid system research hinges on how to combine automatic and crowd-powered systems. These systems can automate simple user requests and leverage human knowledge to solve complex requests in order to achieve a balance between costs and benefits. Therefore, the key research problem is to build better computation models to select system actions. The interaction history of these systems also provides more training data for further automation.

Domain Generalization. Interactive systems require domain knowledge to achieve various goals, thus they are hard to generalize across domains. To address this challenge, I propose to first use information extraction and knowledge inference methods to automatically learn domain knowledge structures from data. I will then use statistical generation and planning methods to encode the learned knowledge structures in system action and policy. The recent work on unsupervised dialog semantic slot induction is the first step towards leveraging knowledge bases to learn domain specified structures from unlabeled conversations [Chen et al.,]. Additionally, the recent development of structural sequence-to-sequence networks, such as long-short-term-memory (LSTM) [Yu et al., 2016d] with attention models [**?**], have provided building blocks for developing advanced algorithms for automatic domain knowledge extraction and encoding in conversations.

Applications I also want to expand the applications of the framework in various fields of science for social good, such as **health care**, **education** and **robotics**. I have worked with Prof. Louis-Philippe Morency on designing virtual agents to facilitate clinical depression and PTSD assessment [Yu et al., 2013b]. The SI framework is general and could make an impact on assessments and therapies for other health issues, such as dementia, aphasia etc. Intelligent interactive systems that detect, coordinate and facilitate users' mental states with conversational strategies designed according to clinical knowledge can benefit society tremendously. Another interesting application field is the use of Artificial Intelligence in education (AIED). I believe by providing a situated intelligent interactive system that communicates with students about course content will facilitate interactive learning at scale, especially for Massive Online Open Courses (MOOCs). Together with Eric Horvitz and Dan Bohus from Microsoft research, I designed an attention-coordinated direction-giving robot (Aldebaran Nao, a humanoid robot) under the SI framework [Yu et al., 2015a]. Due to the physical embodiment of robots, the system actions are richer, but the user-system interaction dynamics will be harder to model. Though difficult, the outcomes are extremely promising. Imagine you can interact with your in-home robot to cook and clean collaboratively; interact with your assistive robot to complete your physical therapy; you can interact with your companion robot to monitor your physical and mental health. I believe the SI framework is general enough to benefit all areas of the sciences.

Bibliography

[Abadi et al., 2013] Abadi, M. K., Staiano, J., Cappelletti, A., Zancanaro, M., and Sebe, N. (2013). Multimodal engagement classification for affective cinema. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, ACII 2013, Geneva, Switzerland, September 2-5, 2013*, pages 411–416. 27, 48

[Ameixa et al., 2014] Ameixa, D., Coheur, L., Fialho, P., and Quaresma, P. (2014). Luke, i am your father: dealing with out-of-domain requests by using movies subtitles. In *Intelligent Virtual Agents*, pages 13–21. Springer. 22

[Andrist et al., 2015] Andrist, S., Ziadee, M., Boukaram, H., Mutlu, B., and Sakr, M. (2015). Effects of culture on the credibility of robot speech: A comparison between english and arabic. In *Proceedings of the HRI*, pages 157–164. ACM. 29

[Apple, 2015] Apple (2015). Siri. http://www.apple.com/ios/siri/. 1

[Baltru et al., 2016] Baltru, T., Robinson, P., Morency, L.-P., et al. (2016). Openface: an open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10. IEEE. 88, 89

[Baltrusaitis et al., 2012] Baltrusaitis, T., Robinson, P., and Morency, L.-P. (2012). 3D constrained local model for rigid and non-rigid facial tracking. In *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2610–2617. IEEE. 30, 78

[Banchs and Li, 2012] Banchs, R. E. and Li, H. (2012). Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the ACL 2012 System Demonstrations*, pages 37–42. Association for Computational Linguistics. 10, 22, 47, 61, 97

[Bednarik et al., 2012] Bednarik, R., Eivazi, S., and Hradis, M. (2012). Gaze and conversational engagement in multiparty video conversation: an annotation scheme and classification of high and low levels of engagement. In *Proceedings of the 4th Workshop on Eye Gaze in Intelligent Human Machine Interaction*, pages 10:1–10:6. ACM. 39

[Benkaouar and Vaufreydaz, 2012] Benkaouar, W. and Vaufreydaz, D. (2012). Multi-sensors engagement detection with a robot companion in a home environment. In *Workshop on Assistance and Service Robotics in a Human Environment at IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 45–52. 39

[Bessho et al., 2012] Bessho, F., Harada, T., and Kuniyoshi, Y. (2012). Dialog system using real-time crowdsourcing and twitter large-scale corpus. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 227–231. Association for Computational Linguistics. 22

[Bickmore et al., 2009] Bickmore, T. W., Pfeifer, L. M., and Jack, B. W. (2009). Taking the time to care: empowering low health literacy hospital patients with virtual nurse agents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1265–1274. ACM. 13

[Bickmore and Picard, 2005] Bickmore, T. W. and Picard, R. W. (2005). Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(2):293–327. 13

[Biermann and Long, 1996] Biermann, A. W. and Long, P. M. (1996). The composition of messages in speech-graphics interactive systems. In *Proceedings of the 1996 International Symposium on Spoken Dialogue*, pages 97–100. 62

[Black and Lenzo, 2001] Black, A. W. and Lenzo, K. A. (2001). Flite: a small fast run-time synthesis engine. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*. 15, 18, 20

[Bohus and Horvitz, 2009] Bohus, D. and Horvitz, E. (2009). Learning to predict engagement with a spoken dialog system in open-world settings. In *Proceedings of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 244–252. Association for Computational Linguistics. 5, 28, 39

[Bohus et al., 2007] Bohus, D., Raux, A., Harris, T. K., Eskenazi, M., and Rudnicky, A. I. (2007). Olympus: an open-source framework for conversational spoken language interface research. In *Proceedings of the workshop on bridging the gap: Academic and industrial research in dialog technologies*, pages 32–39. Association for Computational Linguistics. 16

[Bohus et al., 2014] Bohus, D., Saw, C. W., and Horvitz, E. (2014). Directions robot: in-the-wild experiences and lessons learned. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 637–644. International Foundation for Autonomous Agents and Multiagent Systems. 75

[Bonin et al., 2012] Bonin, F., Bock, R., and Campbell, N. (2012). How do we react to context? Annotation of individual and group engagement in a video corpus. In *Proceedings of the International Conferences on Social Computing (SocialCom) and Privacy, Security, Risk and Trust (PASSAT)*, pages 899–903. IEEE. 28, 32

[Bordes and Weston, 2016] Bordes, A. and Weston, J. (2016). Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*. 97

[Chang and Lin, 2011] Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27. 34, 40, 68

[Chen et al.,] Chen, Y.-N., Wang, W. Y., and Rudnicky, A. I. Unsupervised induction and filling of semantic slots for spoken dialogue systems using frame-semantic parsing. In *ASRU, 2013 IEEE*. IEEE. 49, 111

[Chi et al., 2011] Chi, M., VanLehn, K., Litman, D., and Jordan, P. (2011). Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. *User Modeling and User-Adapted Interaction*, 21(1-2):137–180. 63

[Clark and Brennan, 1991] Clark, H. and Brennan, S. (1991). Grounding in communication. *Perspectives on socially shared cognition*. 50

[Daumé III, 2009] Daumé III, H. (2009). Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*. 13, 45

[D'Haro et al., a] D'Haro, L., Shawar, B. A., and Yu, Z. 2rd re-wochat share task report. In *RE-WOCHAT: 2nd Workshop on Collecting and Generating Resources for Chatbots and Conversational Agents, IVA 2016*, page 39. 2, 12, 20

[D'Haro et al., b] D'Haro, L. F., Shawar, B. A., and Yu, Z. 1st re-wochat share task report. In *RE-WOCHAT: Workshop on Collecting and Generating Resources for Chatbots and Conversational Agents, LREC 2016*. 2, 12, 20

[Dodge et al., 2015] Dodge, J., Gane, A., Zhang, X., Bordes, A., Chopra, S., Miller, A., Szlam, A., and Weston, J. (2015). Evaluating prerequisite qualities for learning end-to-end dialog systems. *arXiv preprint arXiv:1511.06931*. 97

[Draper and Smith, 2014] Draper, N. R. and Smith, H. (2014). *Applied regression analysis*. John Wiley & Sons. 40

[Eskenazi et al., 2008] Eskenazi, M., Black, A. W., Raux, A., and Langner, B. (2008). Let's go lab: a platform for evaluation of spoken dialog systems with real world users. In *Ninth Annual Conference of the International Speech Communication Association*. 75

[Forbes-Riley and Litman, 2009] Forbes-Riley, K. and Litman, D. J. (2009). Adapting to student uncertainty improves tutoring dialogues. In *AIED*, pages 33–40. 85

[Forbes-Riley and Litman, 2012] Forbes-Riley, K. and Litman, D. J. (2012). Adapting to multiple affective states in spoken dialogue. In *Proceedings of the SIGDIAL 2012 Conference, The 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 5-6 July 2012, Seoul National University, Seoul, South Korea*, pages 217–226. 6, 13, 48

[Gatica-Perez, 2009] Gatica-Perez, D. (2009). Automatic nonverbal analysis of social interaction in small groups: A review. *Image Vision Comput.*, 27(12):1775–1787. 28

[Gatica-Perez et al., 2005] Gatica-Perez, D., McCowan, I. A., Zhang, D., and Bengio, S. (2005). Detecting group interest-level in meetings. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, number EPFL-CONF-83257. 32, 33, 39

[Gedawy et al., 2012] Gedawy, H., Ziadee, M., and Sakr, M. (2012). Variations in giving directions across arabic and english native speakers. In *Qatar Foundation Annual Research Forum*. 29

[Georgila and Traum, 2011] Georgila, K. and Traum, D. R. (2011). Reinforcement learning of argumentation dialogue policies in negotiation. In *INTERSPEECH*, pages 2073–2076. 63

[Goleman, 2007] Goleman, D. (2007). *Social intelligence*. Random house. 1

[Gumperz, 1982] Gumperz, J. J. (1982). *Discourse strategies*, volume 1. Cambridge University Press. 6

[He and Young, 1998] He, A. W. and Young, R. (1998). Language proficiency interviews: A discourse approach. *Talking and testing: Discourse approaches to the assessment of oral proficiency*, 14:1–24. 12, 98

[He et al., 2005] He, X., Yan, S., Hu, Y., Niyogi, P., and Zhang, H.-J. (2005). Face recognition using laplacianfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):328–340. 76, 85

[He and Young, 2003] He, Y. and Young, S. (2003). A data-driven spoken language understanding system. In *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, pages 583–588. IEEE. 1

[Higashinaka et al., 2014] Higashinaka, R., Imamura, K., Meguro, T., Miyazaki, C., Kobayashi, N., Sugiyama, H., Hirano, T., Makino, T., and Matsuo, Y. (2014). Towards an open-domain conversational system fully based on natural language processing. In *COLING*, pages 928–939. 49, 62, 97

[Hinton et al., 2012] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97. 75

[Hofstede and Hofstede, 2001] Hofstede, G. H. and Hofstede, G. (2001). *Culture's consequences: Comparing values, behaviors, institutions and organizations across nations*. Sage. 6, 37

[Hsiao et al., 2012] Hsiao, J. C.-y., Jih, W.-r., and Hsu, J. Y. (2012). Recognizing continuous social engagement level in dyadic conversation by using turntaking and speech emotion patterns. In *Proceedings of the Activity Context Representation Workshop at AAAI*. 39

[Hutto and Gilbert, 2014] Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International AAAI Conference on Weblogs and Social Media*. 64

[Impett et al., 2014] Impett, L., Robinson, P., and Baltrusaitis, T. (2014). A facial affect mapping engine. In *Proceedings of the companion publication of the 19th international conference on Intelligent User Interfaces*, pages 33–36. ACM. 34

[Jia and Ruan, 2008] Jia, J. and Ruan, M. (2008). Use chatbot csiec to facilitate the individual learning in english instruction: A case study. In *International Conference on Intelligent Tutoring Systems*, pages 706–708. Springer. 97

[Jiang et al., 2015] Jiang, J., Hassan Awadallah, A., Jones, R., Ozertem, U., Zitouni, I., Gurunath Kulkarni, R., and Khan, O. Z. (2015). Automatic online evaluation of intelligent assistants. In *Proceedings of the 24th International Conference on World Wide Web*, pages 506–516. ACM. 9, 95

[Jurcicek et al., 2011] Jurcicek, F., Keizer, S., Gasic, M., Mairesse, F., Thomson, B., Yu, K., and Young, S. (2011). Real user evaluation of spoken dialogue systems using amazon mechanical turk. In *Proceedings of INTERSPEECH*, volume 11. 96

[Jurcicek et al., 2012] Jurcicek, F., Thomson, B., and Young, S. (2012). Reinforcement learning for parameter estimation in statistical spoken dialogue systems. *Computer Speech & Language*, 26(3):168–192. 63

[Khaled et al., 2006] Khaled, R., Biddle, R., Noble, J., Barr, P., and Fischer, R. (2006). Persuasive interaction for collectivist cultures. In *Proceedings of the 7th Australasian User interface conference-Volume 50*, pages 73–80. Australian Computer Society, Inc. 28

[Kiros et al., 2015] Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing* *systems*, pages 3294–3302. 11, 100

[Kousidis et al., 2014a] Kousidis, S., Kennington, C., Baumann, T., Buschmeier, H., Kopp, S., and Schlangen, D. (2014a). A multimodal in-car dialogue system that tracks the driver's attention. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 26–33. ACM. 5, 48

[Kousidis et al., 2014b] Kousidis, S., Kennington, C., Baumann, T., Buschmeier, H., Kopp, S., and Schlangen, D. (2014b). A multimodal in-car dialogue system that tracks the driver's attention. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 26–33. ACM. 76, 86

[Lamere et al., 2003] Lamere, P., Kwok, P., Gouvea, E., Raj, B., Singh, R., Walker, W., Warmuth, M., and Wolf, P. (2003). The cmu sphinx-4 speech recognition system. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003), Hong Kong*, volume 1, pages 2–5. Citeseer. 77, 87

[Lasecki et al., 2013] Lasecki, W. S., Wesley, R., Nichols, J., Kulkarni, A., Allen, J. F., and Bigham, J. P. (2013). The goal is to give people a solid conclusion by adding some nlp and ml techniques in the conversational systems, if users find the system easier to talk to, like to talk to the system more? : a crowd-powered conversational assistant. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM. 110

[Lee and Eskenazi, 2012] Lee, S. and Eskenazi, M. (2012). Pomdp-based let's go system for spoken dialog challenge. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 61–66. IEEE. 63

[Leffler et al., 1982] Leffler, A., Gillespie, D. L., and Conaty, J. C. (1982). The effects of status differentiation on nonverbal behavior. *Social Psychology Quarterly*, pages 153–161. 29

[Lehman et al., 2012] Lehman, B., D'Mello, S., and Graesser, A. (2012). Interventions to regulate confusion during learning. In *International Conference on Intelligent Tutoring Systems*, pages 576–578. Springer. 91

[Levin et al., 1997] Levin, E., Pieraccini, R., and Eckert, W. (1997). Learning dialogue strategies within the markov decision process framework. In *Automatic Speech Recognition and Understanding, 1997. Proceedings. 1997 IEEE Workshop on*, pages 72–79. IEEE. 62

[Levitski et al., 2012] Levitski, A., Radun, J., and Jokinen, K. (2012). Visual interaction and conversational activity. In *Proceedings of the 4th Workshop on Eye Gaze in Intelligent Human Machine Interaction*, page 11. ACM. 28

[Li et al., 2016a] Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, B. (2016a). A diversitypromoting objective function for neural conversation models. *NAACL*. 10

[Li et al., 2016b] Li, J., Monroe, W., Ritter, A., Galley, M., Gao, J., and Jurafsky, D. (2016b). Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*. 97

[Lison and Tiedemann, 2016] Lison, P. and Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*. 101

[Lombard and Ditton, 1997] Lombard, M. and Ditton, T. (1997). At the heart of it all: The concept of presence, journal of computer mediated-communication. *Journal of Computer Mediated Communication*, 3(2). 27, 48

[Macan, 2009] Macan, T. (2009). The employment interview: A review of current studies and directions for future research. *Human Resource Management Review*, 19(3):203–218. 13

[Marge et al., 2010] Marge, M., Miranda, J., Black, A. W., and Rudnicky, A. I. (2010). Towards improving the naturalness of social conversations with dialogue systems. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 91–94. Association for Computational Linguistics. 15

[Maria Schmidt and Waibel, 2015] Maria Schmidt, J. N. and Waibel, A. (2015). Towards an open-domain social dialog system. In *Proceedings of the 6th International Workshop Series on Spoken Dialog Systems*, pages 124–129. 49, 62

[Martin, 2002] Martin, J. R. (2002). *Meaning beyond the clause: area: self-perspectives*. Annual Review of Applied Linguistics 22. 17, 96

[Matsumoto, 2006] Matsumoto, D. (2006). Culture and nonverbal behavior. *Handbook of nonverbal communication*, pages 219–235. 29

[McGraw et al., 2010] McGraw, I., Lee, C.-y., Hetherington, I. L., Seneff, S., and Glass, J. (2010). Collecting voices from the cloud. In *LREC*. 75

[Mehrez et al., 2013] Mehrez, T., Abdelkawy, A., Heikal, Y., Lange, P., Nabil, H., and Suendermann-Oeft, D. (2013). Who discovered the electron neutrino? a telephony-based distributed open-source standard-compliant spoken dialog system for question answering. *Proc. of the GSCL, Darmstadt, Germany*. 78

[Mesnil et al., 2013] Mesnil, G., He, X., Deng, L., and Bengio, Y. (2013). Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Interspeech*, pages 3771–3775. 98

[Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*. 67, 68

[Misu et al., 2012] Misu, T., Georgila, K., Leuski, A., and Traum, D. (2012). Reinforcement learning of question-answering dialogue policies for virtual museum guides. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 84–93. Association for Computational Linguistics. 7, 63

[Misu et al., 2010] Misu, T., Sugiura, K., Ohtake, K., Hori, C., Kashioka, H., Kawai, H., and Nakamura, S. (2010). Modeling spoken decision making dialogue and optimization of its dialogue strategy. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 221–224. Association for Computational Linguistics. 63

[Morency et al., 2005] Morency, L.-P., Sidner, C., Lee, C., and Darrell, T. (2005). Contextual recognition of head gestures. In *Proceedings of the 7th international conference on Multimodal interfaces*, pages 18–24. ACM. 76, 85

[Moshkina et al., 2014] Moshkina, L., Trickett, S., and Trafton, J. G. (2014). Social engagement in public places: A tale of one robot. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-robot Interaction*, HRI '14, pages 382–389, New York, NY, USA. ACM. 27, 48

[MPAA, 2015] MPAA (2015). Theatrical market statistics 2015. Motion Picture of American Association. 108

[Oertel et al., 2010] Oertel, C., Looze, C. D., Scherer, S., Windmann, A., Wagner, P., and Campbell, N. (2010). Towards the automatic detection of involvement in conversation. In *Analysis of Verbal and Nonverbal Communication and Enactment. The Processing Issues - COST 2102 International Conference, Budapest, Hungary, September 7-10, 2010, Revised Selected Papers*, pages 163–170. 28

[Oertel and Salvi, 2013] Oertel, C. and Salvi, G. (2013). A gaze-based method for relating group involvement to individual engagement in multimodal multiparty dialogue. In *Proceedings of the International Conference on Multimodal Interaction (ICMI)*, pages 99–106. ACM. 33, 39

[Pappu and Rudnicky, 2013] Pappu, A. and Rudnicky, A. (2013). Deploying speech interfaces to the masses. In *Proceedings of the companion publication of the 2013 international conference on Intelligent user interfaces companion*, pages 41–42. ACM. 78

[Pappu et al., 2013] Pappu, A., Sun, M., Sridharan, S., and Rudnicky, A. (2013). Situated multiparty interaction between humans and agents. In *Human-Computer Interaction. Interaction Modalities and Techniques*, pages 107–116. Springer. 18

[Pentland, 2005] Pentland, A. S. (2005). Socially aware, computation and communication. *Computer*, 38(3):33–40. 1

[Peters et al., 2005] Peters, C., Pelachaud, C., Bevacqua, E., Mancini, M., and Poggi, I. (2005). A model of attention and interest using gaze behavior. In *Intelligent Virtual Agents, 5th International Working Conference, IVA 2005, Kos, Greece, September 12-14, 2005, Proceedings*, pages 229–240. 10, 28, 32, 48

[Prylipko et al., 2011] Prylipko, D., Schnelle-Walka, D., Lord, S., and Wendemuth, A. (2011). Zanzibar openivr: an open-source framework for development of spoken dialog systems. In *Text, Speech and Dialogue*, pages 372–379. Springer. 77

[Ramanarayanan et al., 2015] Ramanarayanan, V., Suendermann-Oeft, D., Ivanov, A. V., and Evanini, K. (2015). A distributed cloud-based dialog system for conversational application development. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 432. 78

[Ramanarayanan et al., 2016] Ramanarayanan, V., Suendermann-Oeft, D., Lange, P., Mundkowsky, R., Ivanou, A., Yu, Z., Qian, Y., and Evanini, K. (2016). Assembling the jigsaw: How multiple w3c standards are synergistically combined in the halef multimodal dialog system. In *Multimodal Interaction with W3C Standards: Towards Natural User Interfaces to Everything*, page to appear. Springer. 86

[Rambow et al., 2001] Rambow, O., Bangalore, S., and Walker, M. (2001). Natural language generation in dialog systems. In *Proceedings of the first international conference on Human language technology research*, pages 1–4. Association for Computational Linguistics. 98

[Raux and Eskenazi, 2004] Raux, A. and Eskenazi, M. (2004). Using task-oriented spoken dialogue systems for language learning: potential, practical applications and challenges. In *InSTIL/ICALL Symposium 2004*. 13

[Raux et al., 2003] Raux, A., Langner, B., Black, A. W., and Eskenazi, M. (2003). Let's go: Improving spoken dialog systems for the elderly and non-native. In *in Eurospeech03*. Citeseer. 1, 12

[Raux et al., 2005] Raux, A., Langner, B., Bohus, D., Black, A. W., and Eskenazi, M. (2005). Let's go public! taking a spoken dialog system to the real world. In *in Proc. of Interspeech 2005*. Citeseer. 96

[Rehm et al., 2009] Rehm, M., André, E., Bee, N., Endrass, B., Wissner, M., Nakano, Y., Lipi, A. A., Nishida, T., and Huang, H.-H. (2009). Creating standardized video recordings of multimodal interactions across cultures. In *Multimodal corpora*, pages 138–159. Springer. 28

[Ritter et al., 2011] Ritter, A., Cherry, C., and Dolan, W. B. (2011). Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural* *language processing*, pages 583–593. Association for Computational Linguistics. 22, 47, 61, 97

[Rudnicky and Xu, 1999] Rudnicky, A. and Xu, W. (1999). An agenda-based dialog management architecture for spoken language systems. In *IEEE Automatic Speech Recognition and Understanding Workshop*, volume 13. 96

[Schegloff, 1968] Schegloff, E. A. (1968). Sequencing in conversational openings1. *American anthropologist*, 70(6):1075–1095. 14, 95

[Scherer et al., 2012] Scherer, S., Marsella, S., Stratou, G., Xu, Y., Morbini, F., Egan, A., Morency, L.-P., et al. (2012). Perception markup language: towards a standardized representation of perceived nonverbal behaviors. In *Intelligent virtual agents*, pages 455–463. Springer. 17

[Schnelle-Walka et al., 2013] Schnelle-Walka, D., Radomski, S., and M"u"hlh"a"user, M. (2013). Jvoicexml as a modality component in the w3c multimodal architecture. *Journal on Multimodal User Interfaces*, 7(3):183–194. 77, 87

[Schroeder and Trouvain, 2003] Schroeder, M. and Trouvain, J. (2003). The german text-to-speech synthesis system mary: A tool for research, development and teaching. *International Journal of Speech Technology*, 6(4):365–377. 77, 87

[Sciutti et al., 2015] Sciutti, A., Schillingmann, L., Palinko, O., Nagai, Y., and Sandini, G. (2015). A gaze-contingent dictating robot to study turn-taking. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, pages 137–138. ACM. 76, 86

[Serban et al., 2015] Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. (2015). Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808*. 97

[Shibata et al., 2014] Shibata, T., Egashira, Y., and Kurohashi, S. (2014). Chat-like conversational system based on selection of reply generating module with reinforcement learning. In *Proceedings of the 5th International Workshop Series on Spoken Dialog Systems*, pages 124–129. 63

[Sidner et al., 2004] Sidner, C. L., Kidd, C. D., Lee, C., and Lesh, N. (2004). Where to look: a study of human-robot engagement. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 78–84. ACM. 28

[Singh et al., 1999] Singh, S. P., Kearns, M. J., Litman, D. J., and Walker, M. A. (1999). Reinforcement learning for spoken dialogue systems. In *Nips*, pages 956–962. 62

[Skantze et al., 2015] Skantze, G., Johansson, M., and Beskow, J. (2015). Exploring turn-taking cues in multi-party human-robot discussions about objects. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 67–74. ACM. 5

[Sordoni et al., 2015] Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. *NAACL*. 10

[Sutton and Barto, 1998] Sutton, R. S. and Barto, A. G. (1998). *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge. 99

[Taylor et al., 1998] Taylor, P., Black, A. W., and Caley, R. (1998). The architecture of the festival speech synthesis system. 77, 87

[Toutanova et al., 2003] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics. 17

[Van Meggelen et al., 2007] Van Meggelen, J., Madsen, L., and Smith, J. (2007). *Asterisk: the future of telephony*. " O'Reilly Media, Inc.". 76, 87

[Vinciarelli et al., 2009] Vinciarelli, A., Pantic, M., and Bourlard, H. (2009). Social signal processing: Survey of an emerging domain. *Image and Vision Computing Journal*, 27(12):1743–1759. 2, 76, 79, 85

[Vinyals and Le, 2015] Vinyals, O. and Le, Q. (2015). A neural conversational model. ICML Deep Learning Workshop 2015. 22, 47, 49, 61, 63, 97

[Voigt et al., 2014] Voigt, R., Podesva, R. J., and Jurafsky, D. (2014). Speaker movement correlates with prosodic indicators of engagement. 37

[Walker et al., 1998] Walker, M. A., Fromer, J. C., and Narayanan, S. (1998). Learning optimal dialogue strategies: A case study of a spoken dialogue agent for email. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 1345–1351. Association for Computational Linguistics. 62

[Walker et al., 2004] Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., Wolf, P., and Woelfel, J. (2004). Sphinx-4: A flexible open source framework for speech recognition. 34

[Weizenbaum, 1966] Weizenbaum, J. (1966). Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45. 10

[Wendler, 2014] Wendler, D. (2014). Improve your social skills. *CreateSpace Independent Publishing Platform*. 7, 11, 49, 91, 101

[Whitehill et al., 2014] Whitehill, J., Serpell, Z., Lin, Y.-C., Foster, A., and Movellan, J. R. (2014). The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *IEEE Transactions on Affective Computing*, 5(1):86–98. 89

[Williams et al., 2015] Williams, J. D., Kamal, E., Mokhtar Ashour, H. A., Miller, J., and Zweig, G. (2015). Fast and easy language understanding for dialog systems with microsoft language understanding intelligent service (luis). In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 159. 98, 110

[Williams and Young, 2007] Williams, J. D. and Young, S. (2007). Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language*, 21(2):393–422. 7, 62, 63, 97, 101

[Yu et al., 2015a] Yu, Z., Bohus, D., and Horvitz, E. (2015a). Incremental coordination: Attention-centric speech production in a physically situated conversational agent. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 402. 76, 79, 86, 96, 111

[Yu et al., 2013a] Yu, Z., Gerritsen, D., Ogan, A., Black, A. W., and Cassell, J. (2013a). Automatic prediction of friendship via multi-model dyadic features. In *Proceedings of SIGDIAL*, pages 51–60. 6, 34, 79, 88

[Yu et al., 2016a] Yu, Z., He, X., Black, A., and Alex, R. (2016a). User engagement modeling in virtual agents under different cultural contexts. In *International Conference on Intelligent Virtual Agents*. 6

[Yu et al., 2016b] Yu, Z., Nicolich-Henkin, L., Black, A., and Rudnicky, A. (2016b). A wizard-of-oz study on a non-task-oriented dialog systems that reacts to user engagement. In *17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 6, 86, 89, 97

[Yu et al., 2015b] Yu, Z., Papangelis, A., and Rudnicky, A. (2015b). Ticktock: A non-goal-oriented multimodal dialog system with engagement awareness. In *Proceedings of the AAAI Spring Symposium*. 100

[Yu et al., 2016c] Yu, Z., Ramanarayanan, V., Lange, P., and Suendermann-Oeft, D. (2016c). An open-source multimodal dialog system with real-time engagement tracking for job interview training applications. In *ICASSP submitted*. 12

[Yu et al., 2016d] Yu, Z., Ramanarayanan, V., Mundkowsky, R., Lange, P., Ivanov, A., Black, A. W., and Suendermann-Oeft, D. (2016d). Multimodal halef: An open-source modular web-based multimodal dialog framework. 13, 86, 111

[Yu et al., 2015c] Yu, Z., Ramanarayanan, V., Suendermann-Oeft, D., Wang, X., Zechner, K., Chen, L., Tao, J., Ivanou, A., and Qian, Y. (2015c). Using bidirectional lstm recurrent neural networks to learn high-level abstractions of sequential features for automated scoring of non-native spontaneous speech. In *ASRU*. IEEE. 6

[Yu et al., 2013b] Yu, Z., Scherer, S., Devault, D., Gratch, J., Stratou, G., Morency, L.-P., and Cassell, J. (2013b). Multimodal prediction of psychological disorders: Learning verbal and nonverbal commonalities in adjacency pairs. In *Proceedings of the 17th Workshop on the Semantics and Pragmatics of Dialogue*, pages 160–169. 6, 111

[Yu et al., 2016e] Yu, Z., Xu, Z., Black, A., and Rudnicky, A. (2016e). Chatbot evaluation and database expansion via crowdsourcing. In *Proceedings of the chatbot workshop of LREC*. 63, 102

[Yu et al., 2016f] Yu, Z., Xu, Z., Black, A., and Rudnicky, A. (2016f). Film promotion chatbot: Interleave social chats in tasks. In *IEEE SLT*. 11, 70

[Yu et al., 2016g] Yu, Z., Xu, Z., Black, A., and Rudnicky, A. (2016g). Strategy and policy learning for non-task-oriented conversational systems. In *17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 6, 7, 11, 97, 101, 102, 103, 105

[Zue et al., 2000] Zue, V., Seneff, S., Glass, J. R., Polifroni, J., Pao, C., Hazen, T. J., and Hetherington, L. (2000). Juplter: a telephone-based conversational interface for weather information. *Speech and Audio Processing, IEEE Transactions on*, 8(1):85–96. 75

[Zue et al., 1994] Zue, V., Seneff, S., Polifroni, J., Phillips, M., Pao, C., Goodine, D., Goddeau, D., and Glass, J. (1994). Pegasus: A spoken dialogue interface for on-line air travel planning. *Speech Communication*, 15(3-4):331–340. 96