Situated Multiparty Interaction between Humans and Agents

Aasish Pappu, Ming Sun, Seshadri Sridharan, and Alex Rudnicky

Carnegie Mellon University, Pittsburgh PA 15213, USA

Abstract. A social agent such as a receptionist or an escort robot encounters challenges when communicating with people in open areas. The agent must know not to react to distracting acoustic and visual events and it needs to appropriately handle situations that include multiple humans, being able to to focus on active interlocutors and appropriately shift attention based on the context. We describe a multiparty interaction agent that helps multiple users arrange a common activity. From the user study we conducted, we found that the agent can discriminate between active and inactive interlocutors well by using the skeletal and azimuth information. Participants found the addressee much clearer when an animated talking head was used.

1 Introduction

When more than two people engage in a human-human conversation, they seamlessly communicate and sense who is speaking, who desires to speak, and who is simply listening. This phenomenon is referred to as conversation management in a multiparty scenario. Both verbal and non-verbal cues enable efficient conversation management. In a typical multiparty conversation, participants share a common floor of speech and take turns one at a time to address the floor or a particular addressee. Humans not only show efficient floor management skills but also conform to norms such as politeness and social-role in such situations. On the other hand, imagine a situation with one of the participants being an artificial agent (robot). Here, the problem of floor/conversation management escalates. [1] presented issues that arise in such a situation. They discussed the issues related to a) Participant roles b) Interaction management and c) Grounding and Obligations. Based on these issues, we address three research questions 1) How does an agent determine the roles of different participants? 2) When is appropriate for the agent to take/release the floor? 3) How does an agent communicate (and ground) its understanding of floor and conversation dynamics?

The dynamics of a multiparty conversation are distinct from a two-way conversation (or dialog). [2] proposed special conversation acts called hearer and speech acts. They argue that the traditional definition of a speech act only explains the act of a speaker addressing (assert, promise or apologize) an addressee. Here the assumption is that all addressees are hearers. But, in a multiparty conversation all hearers i.e., all listening participants are not necessarily addressees.

M. Kurosu (Ed.): Human-Computer Interaction, Part IV, HCII 2013, LNCS 8007, pp. 107–116, 2013.

From a computational perspective, we can define conversation as an act that encompasses multiple dialogs with shared or distinct goals with interlocutors who share the floor by taking turns. In a conversation, an agent either jointly informs or requests while addressing all the participants. Whereas in a dialog, an agent only communicates with a particular addressee while all other participants assume the hearer role.

Detecting active participants in a conversation is challenging because one can join or leave the conversation anytime. An agent should keep track of who might engage in a conversation and who has disengaged from a conversation. In addition, it should maintain topic congruity with the active participants on the floor. Therefore, it should decide whether to include or exclude a non-participant in the middle of an active conversation. [3] found that rule-based addressee detection methods are comparable to that of supervised statistical methods such as bayesian networks on a mulitmodal meeting corpus. Gaze patterns, speech, and gesture [4] [5] were found to have predictive power to build addressee detection models. Combining different multimodal inputs compensate for the drawbacks of individual modalities. In our work, we use skeleton and auditory information generated by a Microsoft Kinect[6] to tackle the attention detection problem.

Once the conversation is active, the agent needs to constantly monitor who has the floor and who might get the floor. If the agent makes a request for the floor, it needs to know and monitor how many participants may take the floor before it get its turn. [7] have proposed a heuristic turn-taking policy in a multiparty scenario in a social setting. Such a policy sets up a default behavior for the system and is helpful until the system acquires sufficient data to learn a decision model through interactions.

Since conversation dynamics are complex, an agent should effectively convey its understanding of the dynamics to everyone else in the conversation. In a dialog, an agent only needs to communicate whether or not it understood the user's utterance. In a multiparty conversation, it should also communicate its own understanding of the floor ownership. This helps the participants to take their turn and open up the floor in a timely fashion. An animated talking head or a face has been a norm for embodied agents [8]. Both robotic heads and projection on 2D flat panels have been used as a solution to non-verbal communication, although [9] argue that in multiparty scenario, projections on 2D panels are insufficient to convey which user is being addressed. Instead, they propose a 3D animated back-projected avatar with a mechanical tilt-able neck. In this work, we use animated line-drawings as a 2D talking head. We conducted user studies to investigate its efficacy in turn-taking and state transparency.

In this work, we describe a new framework for multiparty conversation management for an agent in a social settings. This framework tries to address the three fundamental challenges described above. The agent detects the active participants with the help of skeletal and audio sensory information and engages them in conversation. Then, it uses this sensory information and the current conversation state to actively monitor which participant has the floor. With the help of verbal and non-verbal cues (gaze), it conveys its belief of floor ownership and utterance understanding. In addition to addressing these challenges, we also made this framework extend to existing dialog applications for multiparty applications.

This paper is organized as following: In Section 2, we discuss the architecture of the conversation framework. In Section 3, we present empirical results evaluating a system built on this framework. Finally, in section 4 we present concluding remarks and future directions of this work.

2 System Architecture

2.1 Ravenclaw/Olympus Framework

The agent ("SocBot") is implemented using the Olympus/Ravenclaw dialog framework, augmented with multi-modal capabilities (see gray blocks in Fig. 1). A Kinect device is used to acquire speech and human skeleton data. Skeleton information and sound source azimuth information are used to manage the agents attention strategy and as part of the voice activity detection (VAD) process. Speech is decoded by the Automatic Speech Recognizer (ASR) and then processed by a semantic parser. It is further processed by an Input Confidence Estimator (ICE) that combines language, skeleton and azimuth information to determine a given inputs intentionality. Depending on the user input and the current context, the Dialog Management (DM) component decides the next action; this may involve communicating with the Domain Reasoner (DR). Finally a natural language response is generated by the NLG and systems response is synthesized via text-to-speech engine (TTS). Interaction manager (IM) coordinates between system's listening and speaking states to allow barge-ins from the user.

2.2 Customized Ravenclaw/Olympus Framework

We adapted the Olympus/Ravenclaw framework[10] to handle multiparty interaction. For each user, a set of essential components (ICE and DM, as shown in gray blocks on the right side in Fig.1) are spawned and interconnected. Three new components are implemented to naturally handle multiparty conversation (see black blocks in Fig.1). Awareness Server tracks the users in front of the agent and associates each speech input with its corresponding user by using both the skeleton and azimuth information. Conversation Manager (CM) decides when to speak, what to speak, when to listen and whom to listen to. Talking head extracts the addressee stream from CM and NLG. Through its behavior, it displays the current system state (e.g., understood, confused) and the current focus of the agent.

2.3 Conversation Manager

Conversation Manager can access information of all on-going dialogs. CM gates the message flows from speech signal to each dialog manager and natural language requests from dialog managers to synthesizer. To control when to speak

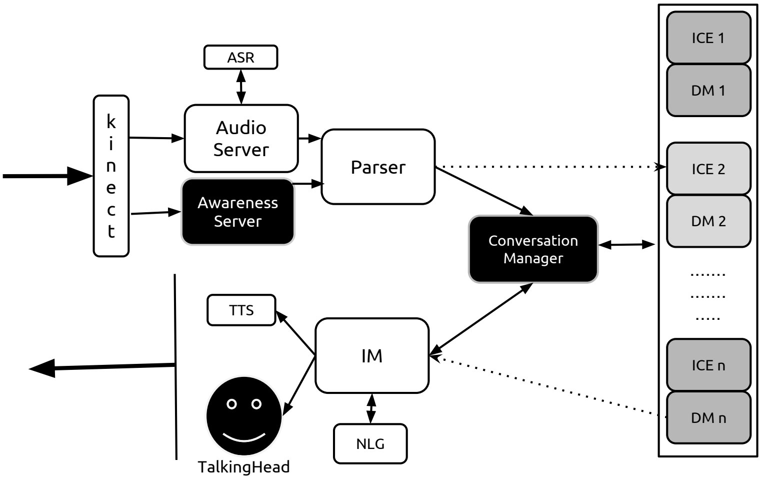

Fig. 1. Multimodal, Multiparty Ravenclaw-Olympus

and what to speak, it also mediates between Dialog Manager (DM) and IM in the output direction. Therefore, CM knows which DM wants to take the floor at any given time. Once the output requests are sent from both Dialog Managers to CM, CM interprets the semantics of the requests and decides which one (user A's, user B's, or a combined one) to forward to IM according to the dialog state. To control when to listen and whom to listen to, CM mediates between Interaction Manager (IM) and Input Confidence Estimator (ICE) in the input direction. Only when CM receives parse(s) from expected user(s), it will send the input to the expected ICE(s) which will then pass the message on to the DM(s) accordingly.

For example, in a dialog state where user hobbies are required, both Dialog Managers send NLG requests with different parameters (DM-A: requests A's hobby. DM-B: requests B's hobby.) to CM. CM first realizes that the two requests are of the same dialog state. Then it finds whether the parameter (user name) is different for these two requests. If so, it generates a plural version of the request (request A and B's hobbies). In the input phase, since CM knows that two answers from two users are expected, it will not send the parses of the input to the rest of the system until two utterances from these two users are provided. Note that a timeout can be triggered if CM does not receive two inputs within a period of time.

In human-human multiparty conversation, we use verbal and non-verbal cues to express our focus. For example, if one is expecting a response from person A, one would either say "*A*, what about you?" or look at person A. instead of someone else. Similarly, these types of cues are provided by CM as well. In the earlier output example, CM interprets the NLG requests and attaches the addressee (user name(s)) to the system prompt. In our system, CM also propagates the addressee information to the talking head, which looks at the expected user(s) accordingly. Details of the talking head are described below.

2.4 Awareness Component

To hold a multiparty conversation, two issues need to be addressed: 1) when to engage and disengage 2) who is the active speaker right now. In our system, the number of skeletons in front of the agent is tracked and updated all the time. When no humans are about, the agent is idle. The agent wakes up upon seeing skeleton(s) in the environment. Based on the number of skeletons available, it decides whether to start a single party dialog or a multiparty conversation. After establishing a conversation with the users, each speech input is associated with the appropriate user. If a skeleton appears or leaves the environment during the conversation, CM is notified to trigger certain conversation-level actions such as acknowledging a newcomer or suspending the dialog channel with the user that left.

To fulfill the capabilities described above, we use the information from the Kinect sensor's microphone array and the visual sensors on-board. This set-up allows multiple users to engage in hands-free fluid interaction with the agent, without wearing close-talking microphones.

As a fundamental requirement to have conversation with multiple people, the system needs to perceive three different types of events. Firstly, it needs to capture audio from mobile sound-sources a few feet away. For this purpose, the far-field microphone array in the Kinect acts as the audio sensor. Before streaming the audio packets to the VAD and the recognizer, we enable on-board noise suppression and echo cancellation to obtain a clean signal. The audio gain level is dynamically adjusted to suit the environmental changes and volume-levels of independent users.

Secondly, the system needs to be aware of user-engagement events. It needs to detect when a user joins the floor, leaves the floor, and should avoid reacting to nonparticipants. We use the skeleton tracking capability of the Kinect to scan the environment for skeletons every 30 frames a second. This allows us to detect skeletal events and assign a unique person-id to each skeleton by tracking/monitoring the skeleton in the environment. To avoid premature firing of events in cases of passerby skeletons entering the environment or the sensor dropping skeletons momentarily, the Awareness component waits for a few hundred milliseconds to observe the environment before making the decision to fire a particular event.

Thirdly, the system needs to be able to discern whom a particular speech input came from. For this purpose, we use the microphone array to track the audio beam, monitoring for changes in the angle of the sound source. When voice is detected at the audio-signal level, we look for a matching skeleton for the current sound-azimuth. When the difference between the orientation of the closest skeleton and the azimuth is within a 15-degree angle, we tag the decoded result for that particular utterance with the user-id of the matched skeleton. The tagged input is sent to the CM for appropriate action.

Since the system knows the exact association between the skeletons and the Dialog Managers, it knows which direction to look at for a particular user's input. This knowledge of the user orientation is used by the graphical user interface (talking head) to direct the prompt towards the target user(s). For example, when DM-A wants an input, the agent looks towards user A's direction, prompt user A, and later expects an utterance from user A.

2.5 Talking Head

As an important non-verbal cue to explicitly indicate the current focus of the agent, we implemented an animated talking head that gazes at whoever it is expecting a response from. Addressee information is sent from CM and the orientation of each user is provided by awareness component. The talking head would also move its lips when the agent is speaking. The following scenarios are considered.

- When addressing the floor and none of the users has replied, its eyes scan the two users.
- When addressing the floor and one of the users has already replied, it looks at the user who has not replied yet.
- When addressing one user, it looks at that user.
- Upon receiving an input not from either of the users, it looks confused.

3 Social Behavior of Multiparty Conversation

The goal of the agent is to engage multiple users in a conversation. To be natural and social, we designed the agent to handle the following situations which are likely to occur in a daily situations.

3.1 Multiparty Conversation Scenario

When talking to two persons, the agent needs to make it clear whom it is addressing and from whom it is expecting a respond. In some dialog states, the agent is addressing the floor by verbally using a plural form of prompt (e.g., "What are your names?") and visually moving its eyes towards both users back and forth. Upon receiving a speech input from one of the users, it gazes at the one other user. In some states, the agent will completely focus on one particular user. Again, the agent will keep eye contact with that specific user during those states. That user's name will appear in the prompts as well if available. One example of this conversation is shown in Fig. 3.3.

3.2 Single User Conversation with Intervention

This scenario describes the situation that one user is talking to the agent while someone else shows an intention to converse with the agent as well. In this case, the agent will shift its focus to the new comer and acknowledge that user. After that, the agent comes back to the first user without losing the context. Once the dialog with the first user is finished, a new interaction can be established if the new comer is still willing to converse.

3.3 Multiparty Scenario with Disengagement

When two users are having a conversation with the agent, if any of them leaves, the conversation should still move on. Awareness component knows exactly which user has left. CM suspends the dialog channel of that user. The agent will focus on the remaining user thereafter. In our future work, we want the agent to reopen the suspended dialog if that particular user comes back in a short period of time. How to help that user recall the earlier conversation context is an interesting research question.

Multiparty Conversation Scenario

Two users walk to the system
System detects the users
S: Are you guys together?
U1: YES
S: What are your names?
U1: DOE
U2: SMITH
System misses U2's utterance
turns towards U2
S: What is your name?
U2: SMITH
S: Hey DOE and SMITH, what activity do you want to schedule?
U1: HIKING
U2: CHESS
System initiates a subdialog with U1
S: Do you want to try "chess" this time?
U1: SURE
S: Thanks for agreeing.
system addresses jointly
S: Your activity has been scheduled.

Single-User Scenario with Intervention

System in the middle of a dialog
U: DOE
S: What is the activity do you want to schedule?
Someone else tries to participate in dialog
System detects them and acknowledges their presence
S: Please give me a minute. I will attend you soon.
resumes dialog with DOE

Multiparty with Disengagement

System in the middle of an interaction
U1: DOE
U2 leaves before conversation ends
System suspends channel with U2
S: Hey DOE, what is the activity do you want to schedule?
U1: HIKING
S: Your activity has been scheduled.

Fig. 2. Example Conversations

4 Experiment

A user study is conducted to answer the following questions:

- How well is the agent distinguishing multiple users' input via skeleton and azimuth features?
- Whether the expected user(s) responded to any given question or not?
- Subjectively, is it clear to the users who is the addressee of the agent in any given dialog state?

The user study included 12 multiparty conversations. Each conversation was carried out by two subjects and the agent. Initially, the two subjects stood outside the view of the agent (Red positions in Fig.3 and Fig.4). Then they walked up to the green positions and faced the agent. Two green positions are one meter in front of the Kinect sensor and the talking head, with 0.8 meter between them . Out of the 12 conversations, 6 were with a talking head. In total, 6 subjects are involved. Each subject participated in 4 conversations — 2 with talking head and 2 without.

The goal of each conversation is to let the agent arrange a common activity for the two subjects. The agent would ask their names first and the two subjects were expected to say their names one by one. Then the agent would ask what activity they wanted to pursue. Again the subjects would tell their activities one after another. The order of response did not matter. After knowing names and activities, the agent would convince one of the subjects to try out the activity of the other subject. A final confirmation would be spoken by the agent.

The subjects don't know in advance who is/are expected to speak in any dialog state. They have to interpret the addressee of current dialog state on their own by language cues (e.g., the agent may say "Hey *A*, what do you want to schedule for tonight?"), talking head cues (e.g., the talking head will look at the current addressee) and conversation context. The two subjects decide whether to take the floor or not. After each conversation, they filled out a survey form.

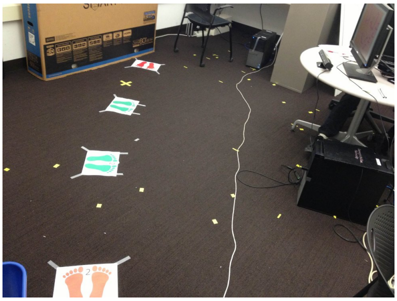

Fig. 3. Real view of the setup

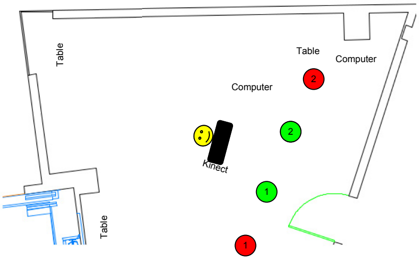

Fig. 4. Schematic view of the setup

4.1 Experiment Results

We found that out of the 140 user utterances, the agent is able to associate 81% of the speech inputs with the correct user. We observed that skeleton angles for both users are very stable. However the azimuth angle was not stable. As a result, sometimes the azimuth cannot be aligned with the correct skeleton, which leads to the errors that the agent either mistook user A's speech as user B's (8% of the total utterances) or none of theirs (11%).

To investigate whether the expected user(s) responded in any given dialog state, we accumulated the number of user utterances which were spoken by the wrong user and were discarded by the system. We found that when there was no talking head, 22% of the user input utterances are wasted. When there was talking head, only 8% were from the wrong user. However, the difference is not statistically significant ($p = 0.24$). Further investigation and experiments are required to verify whether significant difference can be observed when more subjects are involved.

Q1. How would you rate your conversation with the agent?
Q2. Do you think the agent was helpful during the conversation?
Q3. Was it obvious to you what the agent was thinking?
Q4. Is it clear to you whether the agent is expecting a response from both of you?
Q5. Is it clear to you when the agent is talking to both of you?
Q6. Is it clear to you whether the agent is expecting a response from your partner?
Q7. Is it clear to you when the agent is talking to your partner?
Q8. Is it clear to you whether the agent is expecting a response from you?
Q9. Is it clear to you when the agent is talking to you?
Q10. Is it clear to you whom the agent is addressing?

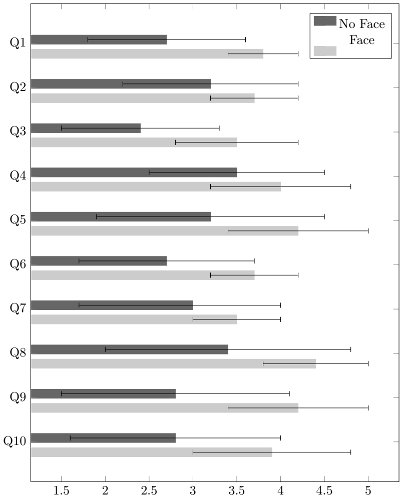

Fig. 5. Subjective results

From the survey results (higher the better), we find out that the overall ratings of the conversation are 2.75 for the agent without a talking head and 3.83 for the talking head one ($p < 0.01$).

The result of the subjective evaluation shows that the addressee is significantly clearer ($p < 0.05$ for each question) when talking head is used. Fig.5 describes the subjective questions and the average score, variance of each question.

5 Conclusion

In this study, we designed and implemented a multiparty spoken dialog system which can simultaneously engage and converse with multiple interlocutors. The addressee of the agent in any given state is indicated via language cues and talking head facial expression cues. From the results of the user study, we found that the agent is able to discriminate multiple users by using skeleton and azimuth information provided by a Kinect. Subjectively, participants found the talking head agent indicates its focus significantly more clearly than the agent without talking head. These results confirm the intuition that multiple conversational cues support more robust and natural interactions.

References

1. Traum, D.: Issues in multiparty dialogues. Advances in agent communication, 1954–1954 (2004)
2. Clark, H.H., Carlson, T.B.: Hearers and speech acts. Language, 332–373 (1982)
3. Akker, R., Traum, D.: A comparison of addressee detection methods for multiparty conversations (2009)
4. Nakano, Y., Ishii, R.: Estimating user's engagement from eye-gaze behaviors in human-agent conversations. In: International Conference on Intelligent user Interfaces, pp. 139–148. ACM (2010)
5. Bohus, D., Horvitz, E.: Learning to predict engagement with a spoken dialog system in open-world settings. In: Proceedings of the 2009 Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 244–252 (2009)
6. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from a single depth image. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (2011)
7. Foster, M., Gaschler, A., Giuliani, M., Isard, A., Pateraki, M., Petrick, R.: "two people walk into a bar": Dynamic multi-party social interaction with a robot agent. In: Proc. of the 14th ACM International Conference on Multimodal Interaction ICMI (2012)
8. Fukuda, T., Taguri, J., Arai, F., Nakashima, M., Tachibana, D., Hasegawa, Y.: Facial expression of robot face for human-robot mutual communication. In: Proceedings of 2002 IEEE International Conference on Robotics and Automation, vol. 1, pp. 46–51. IEEE (2002)
9. Al Moubayed, S., Beskow, J., Skantze, G., Granström, B.: Furhat: A back-projected human-like robot head for multiparty human-machine interaction. In: Esposito, A., Esposito, A.M., Vinciarelli, A., Hoffmann, R., Müller, V.C. (eds.) COST 2102. LNCS, vol. 7403, pp. 114–130. Springer, Heidelberg (2012)
10. Bohus, D., Rudnicky, A.: The ravenclaw dialog management framework: architecture and systems. In: Proceedings of the 2008 Computer Speech and Language, pp. 332–361 (2008)