An Investigation of Interruptions and Resumptions in Multi-Tasking Dialogues

Fan Yang*
Nuance Communications, Inc.

Peter A. Heeman
Oregon Health & Science University

Andrew L. Kun
University of New Hampshire

In this article we focus on human–human multi-tasking dialogues, in which pairs of conversants, using speech, work on an ongoing task while occasionally completing real-time tasks. The ongoing task is a poker game in which conversants need to assemble a poker hand, and the real-time task is a picture game in which conversants need to find out whether they have a certain picture on their displays. We employ empirical corpus studies and machine learning experiments to understand the mechanisms that people use in managing these complex interactions. First, we examine task interruptions: switching from the ongoing task to a real-time task. We find that generally conversants tend to interrupt at a less disruptive context in the ongoing task when possible. We also find that the discourse markers oh and wait occur in initiating a task interruption twice as often as in the conversation of the ongoing task. Pitch is also found to be statistically correlated with task interruptions; in fact, the more disruptive the task interruption, the higher the pitch. Second, we examine task resumptions: returning to the ongoing task after completing an interrupting real-time task. We find that conversants might simply resume the conversation where they left off, but sometimes they repeat the last utterance or summarize the critical information that was exchanged before the interruption. Third, we apply machine learning to determine how well task interruptions can be recognized automatically and to investigate the usefulness of the cues that we find in the corpus studies. We find that discourse context, pitch, and the discourse markers oh and wait are important features to reliably recognize task interruptions; and with non-lexical features one can improve the performance of recognizing task interruptions with more than a 50% *relative error reduction over a baseline. Finally, we discuss the implication of our findings for building a speech interface that supports multi-tasking dialogue.*

1. Introduction

Existing speech interfaces have mostly been used to perform a single task, where the user finishes with one task before moving on to the next. We envision that

* Nuance Communications, Inc., 505 First Ave. South, Suite 700, Seattle, WA 98104.
E-mail: fan.yang@nuance.com.

Submission received: 26 July 2009; revised submission received: 22 July 2010; accepted for publication: 13 October 2010.

next-generation speech interfaces will be able to work with the user on multiple tasks at the same time, which is especially useful for real-time tasks. For instance, a driver in a car might use a speech interface to catch up on e-mails, while occasionally checking upcoming traffic conditions, and receiving navigation instructions; or a police officer might need to be alerted to a nearby accident while accessing a database during a routine traffic stop.

Several speech interfaces that allow multi-tasking dialogues have been built (e.g., Traum and Rickel 2002; Kun, Miller, and Lenharth 2004; Lemon and Gruenstein 2004; Larsson 2003). However, it is unclear that the mechanisms of managing multiple verbal tasks in these systems resemble human conventions or do the best to help users with task switching. For complex domains, the user might be confused about which task the interface is talking about, or might be confused about where they left off in a task.

In order to build a speech interface that supports multi-tasking dialogue, we need to determine a set of conventions that the user and interface can follow in task switching. We propose to start with conventions that are actually used in human–human speech conversations, which are natural for users to follow and probably efficient in problem-solving. Once we understand the human conventions, we can try to implement them in a dialogue manager and run user studies to verify the effectiveness of such conventions in human–computer dialogue.

In this article we focus on understanding the human conventions of managing multiple tasks. Multi-tasking dialogues, where multiple independent topics overlap with each other in time, regularly arise in human–human conversation: For example, a driver and a passenger in a car might be talking about their summer plans, while occasionally interjecting road directions or conversation about what music to listen to. However, little is known about how people manage multi-tasking dialogues. Given the scenario where a real-time task with a time constraint arises during the course of an ongoing task, we are specially interested in two switching behaviors: **task interruption**, which is to suspend the ongoing task and switch to a waiting real-time task, and **task resumption**, which is to return to the interrupted ongoing task after completing a real-time task.

The first question we ask is how quickly conversants respond to a real-time task. Intuitively if the real-time task is very urgent (e.g., the driver is about to miss a turn), the passenger might want to immediately cut off the ongoing conversation, and notify the driver of the turn. However, if the real-time task is less urgent, for example, the driver does not like the music and wants the passenger to load another CD, do conversants still immediately interrupt the ongoing conversation? If conversants do vary how quickly they interrupt, are there any regularities of where conversants switch from the ongoing task to the real-time task? We hypothesize that, given the choice, conversants interrupt the ongoing task where the interruption is less disruptive to the ongoing task.

The second question we ask is how conversants signal task interruptions. Previous research showed that conversants signal the start of a new topic in single-tasking speech (monologue and dialogue) with discourse markers and prosodic cues. We thus hypothesize that conversants also use these cues to signal task interruptions. We also investigate whether conversants vary the intensity of the cues, and under what circumstances.

The third question we ask is what conversants do immediately upon resuming the ongoing task. Switching to a real-time task causes the ongoing task to be temporarily suspended. On completing the real-time task and returning to the ongoing task, do conversants simply continue on from where they were interrupted? We hypothesize that

conversants might sometimes perform certain actions to recover from the interruption. For example, it is imaginable that conversants might ask *where were we at for summer plans*, and then review what was discussed before the interruption.

To answer these questions, we collect the MTD corpus, which consists of a set of human–human dialogues where pairs of conversants have multiple overlapping verbal tasks to perform. In our research, we keep things relatively simple by having conversants talk to each other to play two games on computers. The first game, the ongoing task, is a poker game in which conversants need to assemble a poker hand, which usually takes a relatively long time to complete. The second game, the real-time task, is a picture game in which conversants need to find out whether they have a certain picture on their displays, which can be done in a couple of turns but has a time constraint. In Section 3, we describe the task setup and corpus collection. In Section 4, we examine when and where conversants suspend the ongoing task and switch to the real-time task. In Section 5, we examine how conversants signal task interruptions. In Section 6, we examine the behavior of context restoration in task resumptions.

In addition to the three questions we have asked, in Section 7, we use machine learning to automatically recognize task interruptions. Recognizing task interruptions is an important component in building speech interfaces that support multi-tasking dialogue. For example, the speech interface can accordingly switch the language model when it detects that the user has switched to another task, which should improve speech recognition performance (Iyer and Ostendorf 1999) and utterance understanding, leading to higher user satisfaction (Walker, Passonneau, and Boland 2001). We run machine learning experiments to determine how well we can automatically recognize task interruptions and to understand the utility of the features that we found in our corpus studies. Finally, we conclude the paper in Section 8. This paper includes and extends Heeman et al. (2005), Yang, Heeman, and Kun (2008), and Yang and Heeman (2009) with more corpus data, more robust statistical analysis, more machine learning experiments, and more comprehensive discussions.

2. Related Research

2.1 Existing Systems for Multi-Tasking Dialogues

There is some initial research effort in building speech interfaces to support multi-tasking dialogue. Kun, Miller, and Lenharth (2004) developed a system called Project54, which allowed a user to interact with multiple devices in a police cruiser using speech. The architecture of Project54 allowed for handling multiple tasks overlapped in time. For example, when pulling over a vehicle, an officer could first issue a spoken command to turn on the lights and siren, then issue spoken commands to initiate a data query, go back to interacting with the lights and siren (perhaps to change the pattern after the vehicle has been pulled over), and finally receive the spoken results of the data query. This example shows that system responses related to different tasks could be interleaved: The system responded to the data query after the user had already switched back to interacting with the lights and siren.

Lemon and Gruenstein (2004) also explored multi-tasking in a speech interface. They built a speech interface for a human operator to direct a robotic helicopter on executing multiple tasks, such as searching for a car and flying to a tower. The interface kept an ordered set of active dialogue tasks, and interpreted the user utterance in terms of the most active task for which the utterance made sense. Conversely, during the

interface's turn of speaking, it could produce an utterance for any of the dialogue tasks and thus intermixed utterances from different tasks.

In Kun, Miller, and Lenharth (2004) or Lemon and Gruenstein (2004), the systems did not explicitly signal tasks switching, either for task interruptions or for task resumptions, but instead relied on semantic interpretation to determine which task an utterance belonged to. Larsson (2003) built the GoDis system which hard-coded two types of signals when resuming an interrupted conversation. The first type of signal was to use the discourse marker *so* to implicitly signal a topic resumption. The second type of signal was to use the phrase *returning to the issue of* to explicitly resume an interrupted topic. For example, when searching for the price of an air ticket with GoDis, the user could suspend the system's question *when do you want to travel* by interjecting a question *do I need a visa*. The system, after a short dialogue answering the user's question about a visa, would resume the ticket booking by *returning to the issue of price*.

Traum and his colleagues (Rickel et al. 2002; Traum and Rickel 2002) developed the Mission Rehearsal Exercise system in which the user and virtual humans collaborated on multiple tasks that could interrupt each other. They created a scenario in which a lieutenant (the user) was sent to a village for an Army peacekeeping task. However, on his way, he encountered an auto accident in which his platoon's vehicle crashed into a civilian vehicle, injuring a local boy. The boy's mother and an Army medic were hunched over him, and a sergeant approached the lieutenant to brief him on the situation. These multiple virtual humans could interrupt or be involved in conversations with the lieutenant. The authors proposed and partially implemented a multi-level dialogue manager, with levels for turn-taking, initiative, grounding, topic management, negotiation, and rhetorical structure. In their view, topic management included where one topic is started before an old one is completed. They described how topic shifts in general can be signaled with cue phrases, such as *now* and *anyways*, and with non-verbal cues.

These research works show the usefulness of a spoken dialogue system being able to handle multiple tasks, and promote a thorough examination of multi-tasking dialogue. In this article we examine the conventions of task switching in human–human dialogue as the first step towards understanding the practice of managing tasking switching in a computer dialogue system.

2.2 Insights from Non-Verbal Task Switching

Research in cognitive science suggests that task interruptions and resumptions are complicated behavior and warrant investigation. There is extensive research on the disruptiveness of interruptions, in which individuals switch between multiple manual-visual tasks. For example, Gillie and Broadbent (1989) found that the length (in time) of an interruption is not an important factor, but that the real-time task's complexity and similarity to the ongoing task contribute to the disruptiveness. On the other hand, in their study of checklists, Linde and Goguen (1987) found that it is not the number of interruptions but the length of interruptions that affects the disruptiveness. Cutrell, Czerwinski, and Hovitz (2001) examined the influence of instant messaging on users performing ongoing computing tasks, and found that interruptions unrelated to the ongoing task resulted in longer task resumptions. Although these results do not appear to always converge on the same conclusions, they suggest that task switching can be disruptive to users.

Researchers have been trying to minimize the disruptive effect of task switching in human–computer interaction. McFarlane (1999) explored four alternatives for when

to suspend the ongoing task and switch to the interruption, namely, immediate, negotiated, mediated, and scheduled, and found mixed results. Renaud (2000) argued for, and built, a prototype of a visualization tool to help users restore the context of the ongoing task when returning from an interruption. Hess and Detweiler (1994) and Gopher, Greenshpan, and Armony (1996) found that the disruptive effects are reduced as people gain more experience with interruptions. These studies suggest that it is worthwhile to investigate how a computer dialogue system should manage task switching.

2.3 Insights from Discourse Structure Research

Research in discourse structure also sheds light on task switching. It is important to understand the conventions that people use to manage discourse structure as these might also be used for managing multiple tasks. According to Grosz and Sidner (1986), the structure of a discourse is a combination of linguistic structure, intentional structure, and attentional state. The linguistic structure is a hierarchical segmentation of the dialogue. Each segment has a purpose, which is established by the conversant who initiates the segment. The purposes come together to form the intentional structure. The attentional state contains the objects, properties, and relations that are most salient at any point in the dialogue. The attentional state is claimed to work like a stack. When a new segment is started, a new focus space is created on top of the attentional stack. When the segment completes, the focus space is popped off.[1]

Signaling discourse structure in single-tasking speech is about signaling the boundary of related discourse segments that contribute to the achievement of a discourse purpose. Two types of cues have been identified. The first type is discourse markers (Grosz and Sidner 1986; Schiffrin 1987; Moser and Moore 1995; Passonneau and Litman 1997; Bangerter and Clark 2003). Discourse markers can be used to signal the start of a new discourse segment and its relation to other discourse segments. For example, *now* might signal moving on to the next topic, and *well* might signal a negative or unexpected response.

The second type of cue is prosody. In read speech, Grosz and Hirschberg (1992) studied broadcast news and found that pause length is the most important factor that indicates a new discourse segment. Ayers (1992) found that pitch range appears to correlate more closely with hierarchical topic structure in read speech than in spontaneous speech. In spontaneous monologue, Butterworth (1972) found that the beginning of a discourse segment exhibits slower speaking rate; Swerts (1995) and Passonneau and Litman (1997) found that pause length correlates with discourse segment boundaries; Hirschberg and Nakatani (1996) found that the beginning of a discourse segment correlates with higher pitch. In human–human dialogue, similar behavior has been observed: The pitch value tends to be higher for starting a new discourse segment (Nakajima and Allen 1993). In human–computer dialogue, Swerts and Ostendorf (1995) found that the first utterance of a discourse segment correlates with slower speaking rate and longer preceding pause. Thus, we are interested in whether discourse markers and prosodic cues are also used in signaling task interruptions in multi-tasking dialogue.

1 Grosz and Sidner (1986) also briefly talked about interruptions. In their discourse structure theory, interruptions are modeled as special discourse segments. When a task interruption happens, an attentional state is created for the real-time task and pushed on top of the discourse stack. There is an impenetrable separation between the attentional state of the real-time task and the interrupted ongoing task, so that the real-time task cannot access the ongoing task. When the real-time task is completed, its attentional state is popped off and the ongoing task becomes salient.

3. The MTD Corpus

In order to better understand multi-tasking human–human dialogue, we collected the MTD corpus, in which pairs of players perform overlapping verbal tasks.

3.1 Design of Tasks

For the MTD corpus, we decided to have players complete two types of tasks via conversation: an ongoing task and real-time tasks. The ongoing task needs to build up significant context that players have to keep in mind. On task resumption, this context is needed to finish the task, and so might need to be re-established. The task should also encourage both players to equally participate as we believe that mixed-initiative will be the conversational mode in future speech interfaces. The real-time task can be kept simple: It does not build up much context and can be finished in a couple turns. However, we vary the urgency of this task.

For the ongoing task, a pair of players collaborate to assemble as many poker hands as possible, where a poker hand consists of a full house, flush, straight, or four of a kind. Each player initially has three cards in hand, which the other cannot see. Players take turns drawing an extra card and then discarding one, until they find a valid poker hand, for which they earn 50 points; they then start over to form another poker hand. To discourage players from rifling through the cards to look for a specific one without talking, one point is deducted for each picked-up card, and ten points for a missed or incorrect poker hand. To complete this game, players converse to share card information, and explore and establish strategies based on the combined cards in their hands (Toh, Yang, and Heeman 2006). The poker game is played on computers. The game display, which each player sees, is shown in Figure 1. The player with four cards can click on a card to discard it. The card disappears from the screen, and a new card is automatically dealt to the other player. Once they find a poker hand the player with four cards clicks the *Done Poker Hand* button to start a new game.

The real-time task is a picture game. From time to time, the computer prompts one of the players to determine whether the other has a certain picture on the bottom of the display. The picture task has a time constraint of 10, 25, or 40 seconds, which is (pseudo) randomly determined. Two solid bars above and below the player's cards flash when there is a pending picture game. This should alert the player to a pending picture game without taking the attention away from the poker game. The color of the flashing bars depends on how much time remains: green for 26–40 seconds, yellow for 11–25 seconds, and red for 0–10 seconds. The player can see the exact amount of time left in the heading of the picture game. In Figure 1, the player needs to find out whether the other player has a blue circle, with 6 seconds left. The players get 5 points if the correct answer is given in time. The overall goal of the players is to earn as many points as possible from the two tasks.

3.2 Corpus Collection

We recruited six pairs of players, who each received US $10 for completing the data collection. All players were native American English speakers, and had a bachelor's degree or higher in computer science or electrical engineering. None of the players were in our research lab, and there was no evidence that any player knew about our research program before they participated.

Figure 1
The game display for players.

The data collection for each pair of players lasted about one hour. Players were separated so that they could not see each other and they talked to each other through headsets. After a short orientation, the players played the poker game for about 5 minutes to become familiar with the rules. They then had a practice conversation with both the poker game and the picture game for about 15 minutes, so that they got used to managing both tasks. Finally, they had two more conversations, each lasting for about 15 minutes. In each conversation, nine picture games, three for each urgency level, were prompted for each player. In this research, we analyze the last two conversations, but not the practice one. Thus we have a total of about 180 minutes of conversation from the six pairs of players.

For each dialogue, we recorded both channels of speech (each in an audio file) and created a log file. The log file contains all the events of the computer dealer and the GUI actions of the two players for each task with time-stamps. For the poker game, it contains information of when a card is dealt or discarded, and information of when a poker hand is achieved or missed; for the real-time task, it contains each question, the time it is generated, the answer, and the time it is answered.

A post-experiment survey was conducted in which players were given the following questions: (1) Did you ever play poker before you participated in this experiment? (2) Did you always immediately notice the flashing that signaled a new picture task?

Table 1
Summary statistics of game, card, and picture segments for each pair of players.

	R1	R2	R3	R4	R5	R6	Total
Game segments	7	13	39	35	11	15	120
Card segments	40	118	227	225	82	89	781
Picture segments	30	36	36	35	35	36	208

(3) Did you ever purposefully ignore a picture task? (4) How did you make use of the different urgency levels (40, 25, or 10 seconds)? (5) How did the picture task affect the poker game? (6) Do you have any other comments? All players had at least some poker experience. All players reported that they always noticed the bars immediately when they started to flash, and that they never ignored a real-time task on purpose. Some players also mentioned that they enjoyed the games.

3.3 Dialogue Segmentation

We segmented each dialogue into utterances using consensus annotations (see Yang and Heeman [2010] for more details), following the guidelines of the Trains corpus (Heeman and Allen 1995). We also annotated each utterance as to whether or not it is a trivial utterance. We define **trivial utterances** as those that are just a stall (such as *uh* and *um*) or a simple acknowledgement (such as *okay*, *uh-huh*, and *alright*). According to Strayer, Heeman, and Yang (2003), annotators reached high inter-coder agreement on a similar annotation scheme.[2] There are in total about 4,300 non-trivial utterances in playing the poker game.

The ongoing task can be naturally divided into individual poker games, in which the players successfully complete a poker hand. Each poker game can be further divided into a sequence of card segments, in which players discuss which card to discard, or players identify a poker hand. In total, there are 120 game segments and 781 card segments in the corpus. We also group the utterances involved in each picture game into a segment. Of the 216 prompted picture games, 8 were never started, although players reported that they never ignored a picture game. Hence we have 208 picture games. Table 1 shows the statistics for each pair of players (R1, R2, ..., R6).

Figure 2 shows an excerpt from an MTD dialogue with the segmentations. Here b7 is a game segment in which players get a poker hand of a flush; and b8, b10, b11, b12, and b14, inside of b7, are card segments. Also embedded in b7 are b9 and b13, each of which is a segment for a picture game. As can be seen, players switch from the ongoing poker-playing to a picture game. After the picture game is completed, the conversation on the poker-playing resumes.

Most of the segments can be automatically derived from the log file. For example, the time a new hand is dealt is usually the start of a new game segment; the time a new card is dealt is usually the start of a new card segment. We then manually fixed any mistakes. For example, a mis-generated segment is removed where a player simply discarded a card without any discussion; and a segment boundary is moved if an utterance about the card being discarded, typically an acknowledgment, is said after the new card is dealt.

2 They reported an inter-annotator agreement of 92%, which corresponded to $\kappa = 0.83$.

Figure 2
An excerpt of an MTD dialogue.

The game, card, and picture segments are cohesive units of discourse in which the conversants attempt to complete a domain task, that of winning the card game, deciding what card to discard, or identifying a picture. Thus they follow Grosz and Sidner's (1986) definition of discourse segments.

3.4 Discourse Context

We define discourse context on the task level. We distinguish three types of discourse context where a player suspends the poker playing and switches to a pending picture game: (G) immediately after completing a poker game (at the end of a game), (C) immediately after discarding a card (at the end of a card discussion), and (E) embedded in a card discussion, where players are deciding which card to discard. Corresponding to our dialogue segmentations, an interruption at the end of a game is thus a picture game segment between two poker game segments; an interruption at the end of a card is a picture segment between two card segments; and an interruption embedded in a

card discussion is a picture segment embedded in a card segment. As shown in Figure 2, both b9 and b13 are interruptions at the end of a card discussion.

4. Where to Interrupt

In this section, we examine whether players wait for certain discourse contexts in the poker playing to interrupt with a picture game.

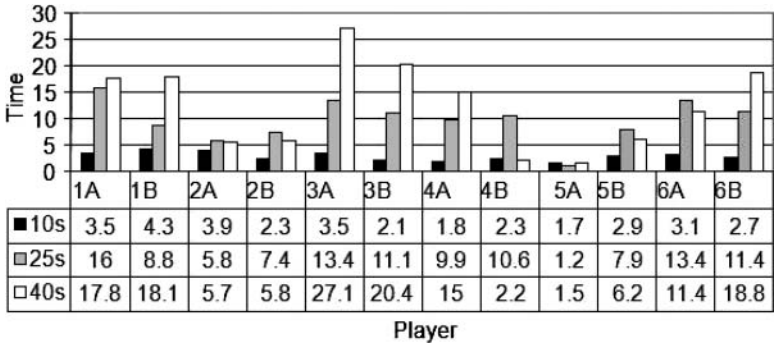
4.1 Response Delay

During poker playing, if a picture game is prompted, the bars around the cards flash in different colors depending on the amount of time left. It is up to the player to decide when to start the picture game (by asking the other player whether there is a certain picture at the bottom of the display). The players can start the picture game as soon as they notice it, for example, within one second; or they can delay the picture game, for example, for 35 seconds, if the time constraint allows. We thus examine the **response delay**, defined as the time interval between when a picture game is prompted and when the player starts it, to understand how soon a player responds to a picture game. We are particularly interested in how players respond to different urgency levels, i.e., whether players wait longer when they are given more time.

Figure 3 shows the average response delay for each player for the urgency levels of 10 sec (black), 25 sec (gray), and 40 sec (white), with the actual values displayed in the columns below. There are certainly individual differences. Player 5A seems to respond to a real-time task as soon as the bars start flashing, regardless of the urgency levels. In fact, in 17 out of the 18 picture games, 5A has less than three seconds of response delay; and the longest response delay is only 3.22 seconds. Player 4B also has interesting behavior: He waits a significant amount of time under the urgency level of 25 sec, but promptly responds under the urgency level of 40 sec. However, overall the response delay under the urgency levels of 40 sec (M = 12.5 sec) or 25 sec (M = 9.7 sec) is much higher than under the urgency level of 10 sec (M = 2.8 sec). The response delay for 40 sec is significantly higher than for 10 sec, $t(11) = 4.2$, $p < 0.001$; as is for 25 sec versus 10 sec, $t(11) = 6.36$, $p < 0.001$. In fact, for question (4) *how did you make use of the different urgency levels (40, 25, or 10 seconds)* in the post-experiment survey, all players but 5A answered that they waited to initiate the picture game when they were given 25 sec or 40 sec (5A answered "not really.") The 10 sec urgency level requires players to start a picture game very quickly in order to complete it in time. On the other hand, when given 25 sec or 40 sec, players are in less of a hurry to switch.

4.2 Urgency Level and Discourse Context

The results on response delay show that players do not always start the real-time picture game as soon as the bars start flashing, especially when players are given 25 sec or 40 sec. Of course there are individual differences: Some players wait longer, some players wait less time, and one does not even wait. The more interesting question, however, is if players do not immediately start the picture game, what is the purpose of delaying the switch to this real-time task? Are players delaying the switch just because they feel that they have time and thus do not need to rush, or because they want to interrupt at a certain point in the ongoing task?



	1A	1B	2A	2B	3A	3B	4A	4B	5A	5B	6A	6B
■10s	3.5	4.3	3.9	2.3	3.5	2.1	1.8	2.3	1.7	2.9	3.1	2.7
▩25s	16	8.8	5.8	7.4	13.4	11.1	9.9	10.6	1.2	7.9	13.4	11.4
□40s	17.8	18.1	5.7	5.8	27.1	20.4	15	2.2	1.5	6.2	11.4	18.8

Figure 3
Response delay for different urgency levels.

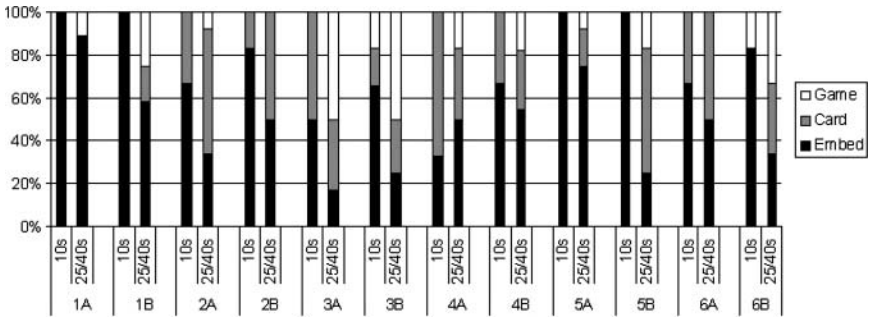


Figure 4
Distribution of discourse contexts for task interruptions under different urgency levels.

We now examine how the urgency level affects where in the discourse context players interrupt the ongoing task and switch to the real-time task. Because we do not find a statistically significant difference of response delay under the urgency levels of 25 sec and 40 sec, $t(11) = 1.51$, $p = 0.16$, we combine these two urgency levels in this analysis.[3]

Figure 4 shows the distribution of the discourse contexts of task interruptions for the urgency levels. Overall the percentage of embedded interruptions for the 10 sec urgency level ($M = 76\%$) is significantly higher than for 25/40 sec ($M = 47\%$), $t(11) = 4.46$, $p < 0.001$. In fact, all players except 4A have a higher percentage of embedded interruptions for 10 sec than for 25/40 sec. The percentage of interruptions at the end of a game for 10 sec ($M = 3\%$) is significantly lower than for 25/40 sec ($M = 20\%$), $t(11) = 4.16$, $p < 0.001$. In fact, all players have a higher or equal percentage of interruptions at the end of a game for 25/40 sec than for 10 sec. These results suggest an answer to our question about why players delay switching to the ongoing task. When players are given more time, that is, when the picture game is less urgent, players often utilize the additional

3 In fact, we find that under the urgency levels of 25 sec and 40 sec, players behave similarly in terms of the discourse context of task interruptions. The reason for the lack of difference might be that it takes on average 90 seconds to complete a poker hand and 14 seconds to complete a card segments. Hence, there is little to be gained from separately reasoning about the 25 sec versus 40 sec urgency levels. In hindsight, we should have used a longer time for the lowest urgency level.

time to delay the switch to the real-time task such that this switch would happen at the end of a game or a card rather than in the middle of a card discussion.

4.3 Response Delay and Discourse Context

In Section 4.2, we find that players tend to interrupt more often at the end of a card or a poker game when they are given more time. However, players do not necessarily wait for more time when the picture game is less urgent. For example, player 5A seems to always start a picture game as soon as the bars start flashing, regardless of how urgent the picture game is. To better understand the rationale of delaying a prompted picture game, we next examine the correlation between response delay and the discourse context where the switch to the real-time task occurs.

We assume that if the response delay is shorter than some amount of time, say t_1, players intend to start the picture game as soon as possible; we also assume that if the response delay is longer than some other time, say t_2, players intend to delay the picture game. For the window between t_1 and t_2, it is unclear as to what players are doing due to individual differences. In this article, we set t_1 to 3 seconds and t_2 to 6 seconds. From listening to the dialogues, it seems to us that when players interrupt within 3 seconds, they intend to do so right away, and when players wait at least 6 seconds, they do not. These two time points are also consistent with human performance in task switching (Meiran, Chorev, and Sapir 2000). This gives us 77 cases of interruptions with a response delay of less than 3 seconds, 88 cases greater than 6 seconds, and 43 cases in between. We have also examined other time thresholds, and find similar results.

Figure 5 shows the distribution of the discourse contexts of task interruptions regarding the response delay. Because 5A always starts a picture game as soon as the bars start flashing, we do not have data for when he waits for more than 6 seconds. We thus exclude 5A from this analysis. The percentage of embedded interruptions for less than 3 sec response delay ($M = 71\%$) is significantly higher than for more than 6 sec response delay ($M = 41\%$), t(10) = 3.54, p = 0.002. The percentage of interruptions at the end of a game for less than 3 sec response delay ($M = 5\%$) is significantly lower than for more than 6 sec response delay ($M = 23\%$), t(11) = 3.49, p = 0.003. Compared with immediately starting a picture game, if players wait for a certain amount of time, they are more likely to suspend the ongoing task at the end of a poker game or a card than to suspend the ongoing task in the middle of a card discussion.

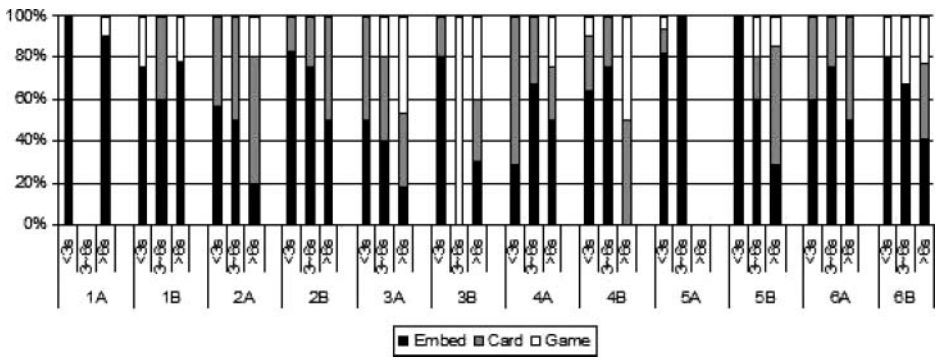


Figure 5
Distribution of discourse contexts for task interruptions under different response delays.

4.4 Discussion

In our research, we define task-level discourse contexts, and investigate the discourse contexts where task interruptions of different urgency occur. We first examine the response delay, and find that players do not always interrupt the poker playing as soon as a picture game starts flashing, but instead they tend to wait longer for less urgent picture games. We then examine the correlation between discourse context and urgency level, and find that when given more time players tend to switch more often to a picture game at the end of a (poker) game or a card. We finally examine the correlation between discourse context and response delay, and find that if players wait for at least a certain amount of time, they tend to switch more often to a picture game at the end of a (poker) game or a card. These results suggest that players prefer to interrupt at the end of a game or a card rather than interrupt in the middle of a card discussion. In fact, after the practice session, player pair R3 explicitly decided that they should try to delay a picture game until the end of a poker game. In other work, Shyrokov, Kun, and Heeman (2007) examined the correlation between task interruption and conversational-level discourse context. Similarly, they found that conversants try to avoid interrupting adjacency pairs.

Discourse context is probably not the only factor that determines when players switch tasks. We observed that sometimes players had time but still chose to interrupt inside a card discussion; or that sometimes players waited past a card segment and then interrupted inside the new card discussion. One guess is that at certain points in a card discussion, players have less cognitive load and so switch tasks. Another guess is that at certain points during poker playing, players get frustrated and decide to switch to a pending picture game. However, these analyses are beyond the scope of this article.

5. Signaling Task Interruption

In this section, we examine how players signal that they are switching from the ongoing task to a real-time task. In Section 2.3, we discussed how people use certain cues, such as discourse markers and prosody, to signal discourse structure in single-tasking speech. This suggests that people might also signal task interruptions in multi-tasking dialogues and might even use similar cues.

5.1 Discourse Markers

First, we examine whether discourse markers co-occur with task interruptions. For this exploratory study, we treat any word that can serve as a discourse maker and that precedes a task interruption as a discourse marker, even though their roles in dialogue are sometimes ambiguous, such as *and*, *now*, and *okay* (Gravano et al. 2007). We also include the fillers *uh* and *um*, which were shown to sometimes have a discourse function (Swerts 1998).

Of the total 208 task interruptions, 76 are initiated with a discourse marker, which accounts for 36.5%. We list these discourse markers in Table 2 grouped by their discourse function. For their use in task interruptions, column 2 shows the number of occurrences of each group and column 3 shows the number of players who use them. The first group consists of *oh* and *wait*, which are usually used to signal a sudden or urgent event (Heritage 1984; Schiffrin 1987; Byron and Heeman 1997). This group has the most frequently uttered discourse markers in task interruption with 27 occurrences, and seven players utter them at least once. The second group consists of the fillers *uh* and

Table 2
List of discourse markers used in task interruptions.

Discourse Markers	Total Occurrences	Number of Players
oh wait	27	7
um uh	23	10
now okay alright	13	8
and	10	5
OTHERS (so but hey)	3	3

um. This group is uttered by the most players with 23 occurrences. The third group consists of *now*, *okay*, and *alright*, which can signal the end of the current topic and moving on to the next (Hirschberg and Litman 1987; Gravano et al. 2007). This group has 13 occurrences by eight players. The word *and* is uttered 10 times by five players. Finally there is one occurrence of *so*, one of *but*, and one of *hey*. Interestingly, there are also two cases of calling the name of the other player, such as *Gary do you have a blue triangle?*

We next examine the discourse markers *oh* and *wait* in more depth. We choose them because this group co-occurs most frequently with task interruptions, and because task interruptions involve starting a new and urgent task, which fits their discourse function. Verifying whether *oh* and *wait* are being used as discourse markers is straightforward. We manually verified that all 27 instances of *oh* and *wait* that initiated a picture game are discourse markers, and we also identified all usages of *oh* and *wait* in poker playing that are discourse markers. For each player, we calculated the rate of task interruptions initiated with an *oh* or *wait*, and compared it with two baselines: (1) the rate of non-trivial utterances in poker playing that are initiated with an *oh* or *wait*, and (2) the rate of card segments that are initiated with an *oh* or *wait*. The rate of task interruptions initiated with an *oh* or *wait* ($M = 12.7\%$) is significantly higher than the rate of utterances initiated with an *oh* or *wait* ($M = 5.7\%$), $t(11) = 1.80$, $p = 0.05$. It is also higher than the rate of card segments initiated with an *oh* or *wait* ($M = 7.1\%$), which is marginally significant $t(11) = 1.66$, $p = 0.06$. These results suggest that the discourse markers *oh* and *wait* are sometimes used in signaling task interruptions.

5.2 Prosody

To understand the prosodic cues in initiating a topic, traditionally researchers compared the prosody of the first utterance in each topic with other utterances (e.g., Nakajima and Allen 1993; Hirschberg and Nakatani 1996). For example, they calculated the average pitch in the utterance or the first part of the utterance that initiates a topic and found that it is higher than the other utterances in the topic. This approach encounters two problems here. First, the words in an utterance might affect the prosody. For example, the duration and energy of *bat* are usually larger than *bit*. Thus a large amount of data are required to balance out these differences. Second, in the MTD corpus, players typically switch to a picture game by using a yes–no question, such as *do you have a blue circle*, whereas most non-trivial utterances in the ongoing task are statements or proposals. As questions have very different prosody than statements or proposals, a direct comparison is further biased.

Examination of the MTD corpus finds that 82% (170/208) of the picture games are initiated by *do you have ...* with optional discourse markers at the beginning. While in

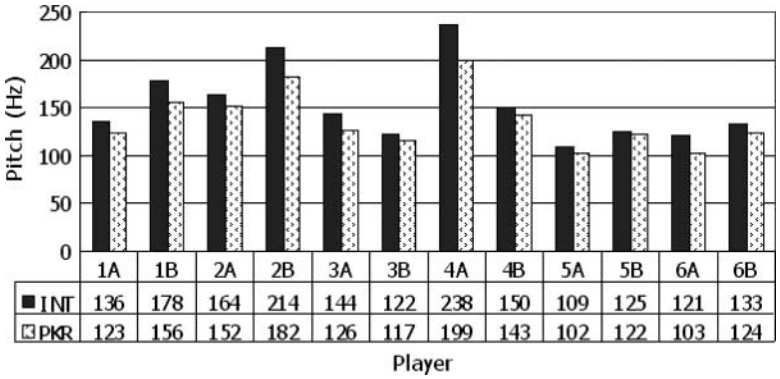


Figure 6
Average pitch of *do you have* for task interruptions and poker playing.

the poker game, players use *do you have ...* 115 times to ask whether the other has certain cards, such as *do you have a queen?* This abundance of utterances with identical initial-wording and speech-act-type inspired us to compare the prosody of the phrase *do you have* in switching to a picture game and during poker-playing.[4] This avoids comparing prosody of different words or of different types of utterances.

We measure pitch, energy (local root mean squared measurement), and duration of each case of *do you have*. We aggregate on each individual player and calculate the average values. Figure 6 shows the average pitch of the phrase *do you have* in task interruption (INT) and poker-playing (PKR) of each player, with the actual values displayed in the columns below. For task interruption, players' average pitch is significantly higher than poker-playing, $t(11) = 4.82$, $p < 0.001$. In fact, for each of the 12 players, the average pitch of *do you have* in task interruption is higher than in poker-playing. These results show a strong correlation between task interruption and higher pitch.

We also examine energy and duration (speaking rate) for the phrase *do you have* in task interruption and poker-playing. However, we do not find a statistically significant difference in energy, $t(11) = 1.53$, $p = 0.16$, or in duration $t(11) = 1.67$, $p = 0.12$.

5.3 Intensity of Cues

To better understand how pitch is used in signaling task interruptions, we next examine whether it correlates with the discourse context of interruptions, namely, interrupting at the end of a game, at the end of a card discussion, or embedded in a card discussion. Because there are relatively fewer data for interrupting at the end of a game, we combine interruptions at the end of a game and at the end of a card discussion (G/C).

Figure 7 shows the average pitch of *do you have* when switching to a picture game embedded in a card discussion, at the end of a game or card discussion, and during poker-playing (i.e., no task switching involved), with the actual values displayed in the columns below. The difference between these three conditions is statistically significant, $F(2, 11) = 21.60$, $p < 0.001$. Interruptions embedded in a card discussion has a significantly higher pitch than at the end of a game or card discussion, $t(11) = 5.74$, $p < 0.001$,

4 It would have been interesting to compare the prosody of utterances that initiate a picture game and those that initiate a card segment. However, we do not have enough utterances that initiate a card segment that begin with *do you have*.

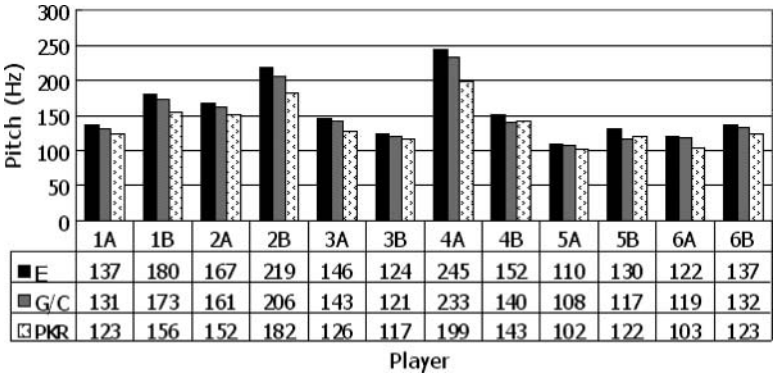


Figure 7
Average pitch of *do you have* for different discourse contexts.

which in turn has a significantly higher pitch than during poker-playing, $t(11) = 3.56$, $p = 0.002$. These results suggest a statistical correlation between discourse context of task interruption and intensity of cues.[5]

5.4 Discussion

We find that discourse markers are sometimes used to mark task interruptions, but for less than 40%. For the discourse markers *oh* and *wait*, we find a statistical correlation between their use and task interruptions. This result should not be surprising as task interruptions involve a sudden change of the conversation topic, and previous research found that conversants use *oh* to mark a change of state in orientation or awareness. *wait* is used to mark a discontinuity in the ongoing topic, which is also required by task switching. Thus, it seems natural for people to use these discourse markers to signal switching to a real-time task. The use of the other discourse markers is less clear, but we have some speculations about their use with task interruptions. The discourse markers *now*, *okay*, and *alright* tend to start a new topic in single-tasking speech, which is consistent with initiating a task interruption. The fillers *um* and *uh* might be used to hold the floor giving the player who initiates the picture game extra time to mentally switch tasks; or they might be used to help mark the switch itself, similar to how they sometimes mark topic shifts (Swerts 1998). Calling the name of the other player or saying *hey* might be used to alert the other player of the task switching.

We also find that players signal task interruptions with prosodic cues. Pitch turns out to be the most prominent feature. Not only do we find a strong correlation between higher pitch and task interruption, but we also find a correlation between pitch and the discourse context of the interruption. Switching embedded in a card segment has a higher pitch than switching at the end of a card segment or a game, which in turn has a higher pitch than non-switching (poker-playing). We speculate that pitch, as well as discourse markers and calling the name of the other player, is used to disengage the hearer from the ongoing task, signaling an unexpected event (see Section 8.1 for more discussion).

5 We are also interested in whether players alter their use of discourse markers depending on the place of interruption. However, perhaps due to a lack of data, we do not find a statistical difference.

On the other hand, we do not find a statistically significant correlation between energy and task interruption, or between speaking rate and task interruption. It would be interesting to understand why pitch is used yet not other prosodic cues. In another study, in which we examined initiative conflicts, where both conversants speak at the same time trying to steer the conversation in different directions, we found that energy is the dominant device for resolving who wins the conflict (Yang and Heeman 2010). Probably conversants use different prosodic devices, such as pitch, energy, and speaking rate, for different conversational functions. Further research is needed to explore this hypothesis.

Finally, it is also interesting to investigate whether players signal the urgency of the real-time task. In our task setup, besides the urgency level, which is the time (10 sec, 25 sec, or 40 sec) initially given to the players to complete a picture game, a more important factor that defines urgency is the remaining time, which is the time left to complete a picture game when players switch to it. Intuitively, when players start a picture game, the more time that is remaining to finish the task, the less hurried they need to be. However, we were not able to find a statistical correlation between urgency level and pitch, or between remaining time and pitch. Nor was there a statistical correlation with volume or speaking rate. Our explanation is that our task setup might not be complicated enough. It only takes a couple utterances to finish a picture game, and players were able to start the picture task far enough ahead that remaining time was rarely a factor.

6. Context Restoration

On completing an interrupting picture game, players resume poker playing. Due to our task setup, players tend to mutually know when a picture game ends. Thus we do not examine how players signal task resumption, but instead we focus on how players restore the context of the ongoing task, that is, how players re-establish the conversation on poker playing after being interrupted by a picture game. We use the same distinction of discourse contexts as we use in examining task interruptions: (1) restoration in the middle of a card discussion, which corresponds to the players interrupting embedded in a card discussion; (2) restoration at the beginning of a card, which corresponds to the players interrupting after a card discussion, and then resuming to poker playing with one of the players having a new card; and (3) restoration at the beginning of a game, which corresponds to the players interrupting at the end of a poker game, and then resuming to poker playing with the beginning of another poker game.

6.1 Restoration in the Middle of a Card Discussion

We start by investigating context restoration in the middle of a card discussion, because these have the most context. We explored the corpus to look for signs of context restoration behavior after an embedded interruption, by examining informational redundancy (Walker 1996) of the first non-trivial utterance after completing a picture game.

Probably due to the simplicity of the picture game, especially that it can be completed in a couple turns, we find that after completing an embedded picture game players usually continue poker playing without a clear indication of context restoration. As shown in Example (1), B suspends his own question in poker playing and interrupts with a picture game. After the completion of the picture game, A gives the answer to B's original question right away: The dialogue on poker playing continues as if the interruption never happened.

Example 1 (Continuation)

B: what do you have to make a high straight with?
B: you got a red circle?
A: no
A: I have a ten of diamonds and an ace of clubs

We do, however, find two types of utterances at the beginning of a resumption that are informationally redundant (Walker 1996), as listed here.

Utterance Restatement: The first non-trivial utterance after the interruption is a restatement of the last non-trivial utterance before the interruption. This can be further divided into three sub-categories: A) self-repetition: the player repeats (part of) his or her own utterance, as shown in Example (2); B) other-repetition: the player repeats (part of) the other's utterance, as shown in Example (3); and C) clarification: the player asks for a repetition with a clarification question, as shown in Example (4).

Example 2 (Self-Repetition)

B: I have three clubs right now
B: do you have a yellow square?
A: yes
B: I have three clubs
B: do you have any clubs?

Example 3 (Other-Repetition)

B: I have jack and two queens
B: um do you have a yellow plus sign?
A: yes
A: a jack and two queens
A: I have a ten

Example 4 (Clarification)

A: I have a six of clubs a nine of spades and a four of diamonds
B: okay
B: okay how about a uh red cross
A: no
B: okay
B: four diamonds six something?
A: clubs

Card Review: The player re-communicates what cards are in hand, as shown in Example (5). We define card review as utterances that inform of all of the cards in the player's hand, and where this information has already been communicated.

Example 5 (Card Review)

A: so I got a ten of spades
B: alright
B: and do you have a red circle?
A: um yes
B: I mean no no a blue circle
A: oh yes
A: and okay I have a queen of spades a ten of I mean a queen of diamonds a ten of spades a king of clubs and a two of clubs

For the 115 embedded interruptions, we find 34 cases of utterance restatement (20 self-repetitions, 4 other-repetitions, and 10 clarifications) and 9 cases of card review. Figure 8 shows the rate of each category, aggregated on each pair of players (R1-R6).

The rate of utterance restatements, calculated as the number of embedded interruptions that are followed by an utterance restatement divided by the total number of embedded interruptions, ranges from 22% to 37% among the player pairs. To make sense of these numbers, we annotate each non-trivial utterance in the poker games to mark whether it is a restatement of the immediate previous one within a card segment, and calculate the baseline as the rate of performing utterance restatement without being interrupted by a picture game. From Figure 8, we see that for all six pairs of players, the rate of utterance restatement after an embedded interruption is higher than the baseline, and it is statistically significant, $t(5) = 13.52$, $p < 0.001$. This suggests that utterance restatement after an embedded interruption is not a random behavior, but it is part of the resumption to the ongoing task.

We next examine card review, which does not seem to be a common behavior in all player pairs. The pairs R1, R4, and R6 never performed it in resuming poker playing, and R2 only performed it once. The pairs with the highest rates are R5 and R3, with 26% (6/23) and 17% (2/12), respectively. Interestingly, these two pairs have the lowest rates of performing utterance restatement, with 22% and 27%, respectively. This might suggest that card review might be complementary to utterance restatement for context restoration, although more data are needed to validate this hypothesis.

6.2 Restoration at the Beginning of a Card Segment

For restoration at the beginning of a card segment, we find that players mostly just continue poker playing without a clear indication of being affected by the interruption, as illustrated by card segments b10 and b14 in Figure 2. Some players might perform an act similar to card review, in that they communicate all of the cards in his or her hand. However, the version here differs as it includes the new card just picked up, which has not been communicated before. Thus we refer to this act as **card review + new card**.

Table 3 shows the rate of performing *card review + new card* after an interruption for each pair, respectively. The baseline is the rate of performing this action at the beginning

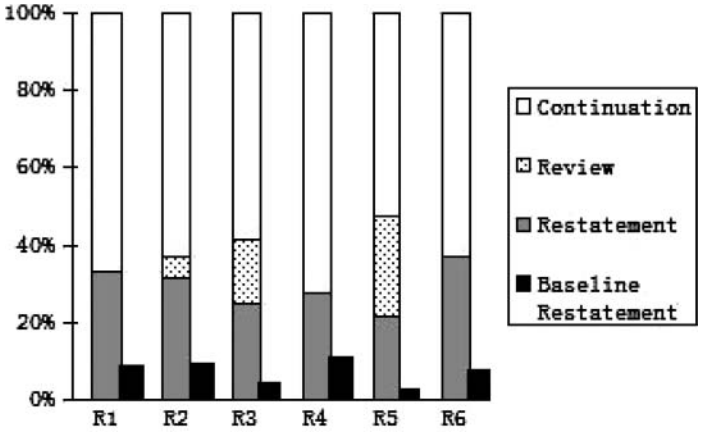


Figure 8
Restoration in the middle of a card discussion.

Table 3
Restoration at the beginning of a card segment.

	R1	R2	R3	R4	R5	R6
Card review + new card	0% (0/2)	19% (3/16)	9% (1/11)	8% (1/13)	0% (0/9)	42% (5/12)
Baseline	0% (0/31)	6% (5/89)	1% (2/177)	3% (5/177)	0% (0/62)	5% (3/62)

of a card segment (excluding the first card segment of a poker game) not following an interruption of a picture game. Player pairs R1 and R5 never performed this action at all. R3 and R4 performed this action only once after interruptions at the end of a card discussion and have a very low overall rate of performing this action during poker playing. However, for player pairs that have a high overall rate of using this action (R2 and R6), they have an even higher rate of using this action after an interruption. For R6 in particular, the rate of performing this action after an interruption at the end of a card segment is significantly higher than the baseline, $\chi^2(1) = 6.81$, $p = 0.01$. This suggests that if players use *card review + new card* in conversation, they tend to use it more often after an interruption at the end of a card segment probably for context restoration.

6.3 Restoration at the Beginning of a Game

For interruptions at the beginning of a poker game, we do not find any behavior associated with context restoration. This is not surprising because there is really no context that needs to be carried over to the next game.

6.4 Discussion

In this section, we examine the behavior of context restoration when players complete an interrupting picture game and resume poker playing. Probably due to the simplicity of the picture game, we find that players mostly just have a smooth continuation as if the interruption did not happen. However, we do find that players sometimes make two types of context restorations—utterance restatements and card reviews—and we find that players have a higher rate of performing these when returning to the ongoing task.

Card review seems to be refreshing the critical information needed to complete a task, while utterance restatement is refreshing the last utterance. Both types of restoration behavior are similar to the informationally redundant units that Walker (1996) studied. In Walker's work, she posited a limited memory model in which information will eventually fade away. This might be the explanation here as well. On resuming to a task that was discussed several utterances ago, the conversant might feel that some of the critical information might have been forgotten, and so might use card review to refresh the information. Conversely, the conversant might feel that just the last utterance needs to be refreshed. Depending on whether it is the same conversant who resumes the ongoing task and who says the last utterance before the interruption, it takes the form of a self-repetition, other-repetition, or request-repetition if clarification is needed. This explanation for card review and utterance restatement is consistent with the results of our post-experiment survey, in which some players reported that they had difficulties

remembering the context of poker playing when they were interrupted by a picture game.

In a more complex domain, conversants will probably perform context restoration more frequently when returning to an interrupted task (Gillie and Broadbent 1989; Villing 2010), and might use even higher-level summarization beyond utterance restatement and information review, such as reviewing the agreements or decisions that have been made so far in the conversation.

7. Recognizing Task Interruption: A Machine Learning Approach

Recognizing task switching is important for a speech interface; for example, the speech interface can accordingly switch the language model when it detects that the user has switched to another task. In this section, we describe two machine learning experiments of recognizing task interruptions using prosody, discourse context, and discourse markers. The purpose of the first experiment is to understand how these features contribute to the automatic identification of task interruptions; here, we only include utterances that start with *do you have* for better extracting prosodic features. The purpose of the second experiment is to investigate how well interruptions can be identified without using lexical features, as could be used in an actual system.

7.1 Recognizing Task Interruptions on *Do You Have* Utterances

In the previous sections, we examined players' behavior of task switching in the MTD corpus. We found that players favor certain discourse contexts in the ongoing task for task interruptions, and that they signal task interruptions with prosodic cues and sometimes with certain discourse markers (*oh* and *wait*). We thus conduct a machine learning experiment to understand how these features contribute to the automatic identification of task interruptions. In this experiment, we focus on the 285 cases of *do you have*, 170 for task interruption and 115 for poker playing. As we argued in Section 5.2, this allows us to better extract and understand prosodic features of task interruptions.

We extract the following features: 1) discourse context: whether the utterance before *do you have* is the end of a poker game, the end of a card segment, or in the middle of a card segment; 2) oh/wait: whether the discourse marker *oh/wait* precedes *do you have*; 3) normalized pitch: the pitch of *do you have* divided by the average pitch of the speaker during the dialogue. We refer to these features as the core feature set, which we found to be correlated with task interruptions (Section 4 and 5). We also include the following additional features: 4) discourse markers: whether a discourse marker precedes *do you have*; 5) normalized energy: the energy of *do you have* divided by the average energy of the speaker during the dialogue; and 6) duration: the duration of *do you have*.

We use a decision tree classifier (C4.5) to discriminate task interruption from poker playing (Quinlan 1986). C4.5 builds a decision tree by using a top–down, greedy procedure to (locally) optimize mutual information, and prunes the tree with a confidence level (of 25%). We use C4.5 because its output is interpretable and we have found its performance comparable to other discriminative classifiers for this task.

We use three re-sampling methods in training and testing the decision tree learning, which we refer to as general-leave-one-out, speaker-leave-one-out, and leave-one-speaker-out. In the general leave-one-out method, each data point is tested with the decision tree trained on all other data points. This approach allows decision trees to be built with as much training data as possible, which in our case is 284 data points.

Table 4
Performance for general-leave-one-out.

	Accuracy	Recall	Precision	F
Baseline	59.6%	100.0%	59.6%	74.7%
Core features	81.4%	89.4%	81.3%	85.2%
Core + discourse markers	80.7%	88.2%	81.1%	84.5%
Core + energy + duration	80.7%	85.3%	82.9%	84.6%
All features	80.4%	84.7%	82.8%	83.7%

In the speaker-leave-one-out method, each data point is tested with the decision tree trained on the other data points of the same player. This approach is a speaker-specific model that evaluates the performance of training a decision tree and testing on the same speaker. In the leave-one-speaker-out method, each player's data are tested with the decision tree trained on the other 11 players. This approach is a speaker-independent model that evaluates the performance of a learned decision tree on a new speaker.

Table 4 shows the results with the general-leave-one-out method. The decision tree learning with the core feature set obtains an accuracy of 81.4% in recognizing whether a *do you have* initiates a task interruption or belongs to poker playing; and the recall, precision, and F-score for task interruption are 89.4%, 81.3%, and 85.2%, respectively. For comparison, we use a naive baseline that assumes that all cases of *do you have* are task interruptions, which has an accuracy of 59.6%. Thus we achieve 54.0% relative error reduction in comparison to the baseline. These results show that our machine learning approach substantially improves the recognition of task interruptions.

Also from Table 4 we see that there is no improvement by adding more features, namely, discourse markers, energy and duration, or all of them. This suggests that these features are not adding more information to this discrimination task, which is not surprising as we did not find them strongly correlated with task interruption in our corpus study.

Table 5 shows the results for each player with the general-leave-one-out, the speaker-leave-one-out, and the leave-one-speaker-out, using the core feature set.

Table 5
Accuracy for the three re-sampling methods.

Player	General-leave-one-out	Speaker-leave-one-out	Leave-one-speaker-out
1A	75.0%	66.7%	75.0%
1B	84.6%	69.2%	73.1%
2A	77.8%	74.1%	77.8%
2B	100.0%	90.0%	95.0%
3A	88.0%	88.0%	88.0%
3B	76.7%	76.7%	72.6%
4A	94.4%	94.4%	94.4%
4B	64.7%	64.7%	64.7%
5A	73.9%	69.6%	73.9%
5B	92.9%	71.4%	92.9%
6A	89.5%	84.2%	78.9%
6B	63.6%	90.9%	63.6%
Mean	81.8%	78.3%	79.2%

Overall, all the three reach an accuracy of about 80%, which is much higher than the baseline performance. The performance with the leave-one-speaker-out ($M = 79.2\%$), which is a speaker-independent model, is particulary encouraging, because in building a speech interface, it is not always possible to collect speaker-specific data. On the other hand, we see that the performance with the speaker-leave-one-out ($M = 78.3\%$) is slightly lower than the leave-one-speaker-out ($M = 79.2\%$). Although this could be interpreted as that interruption recognition is a speaker-independent task, we think that a more viable explanation is that for some players, we do not have enough data to build speaker-specific decision trees. The general-leave-one-out ($M = 81.8\%$), which uses the most data for training, out-performs the leave-one-speaker-out and the speaker-leave-one-out. In fact, the general-leave-one-out can also be viewed as a naive speaker-adaptive model by simply combining speaker-independent data and speaker-specific data together for training. We speculate that more improvement can be achieved by interpolating a speaker-independent model with a speaker-specific model, which we leave for future work.

Finally, we examine the structure of the decision trees learned. Here, we build a single tree from all 285 cases of *do you have* with the core feature set, shown in Figure 9. In the decision tree, the first query is about pitch. If pitch is low it is for poker playing, otherwise it queries about *oh*/*wait*. If the utterance starts with a *oh* or *wait*, it is for task interruptions, otherwise it queries about discourse context. If the discourse context is at the end of a game or a card discussion, it is for task interruption, otherwise it queries pitch again. If pitch is lower than a threshold it is for poker playing, otherwise it is for task interruptions. The structure of the learned tree and its performance confirm that discourse context, the discourse markers *oh* and *wait*, and normalized pitch are useful features for recognizing task interruptions.

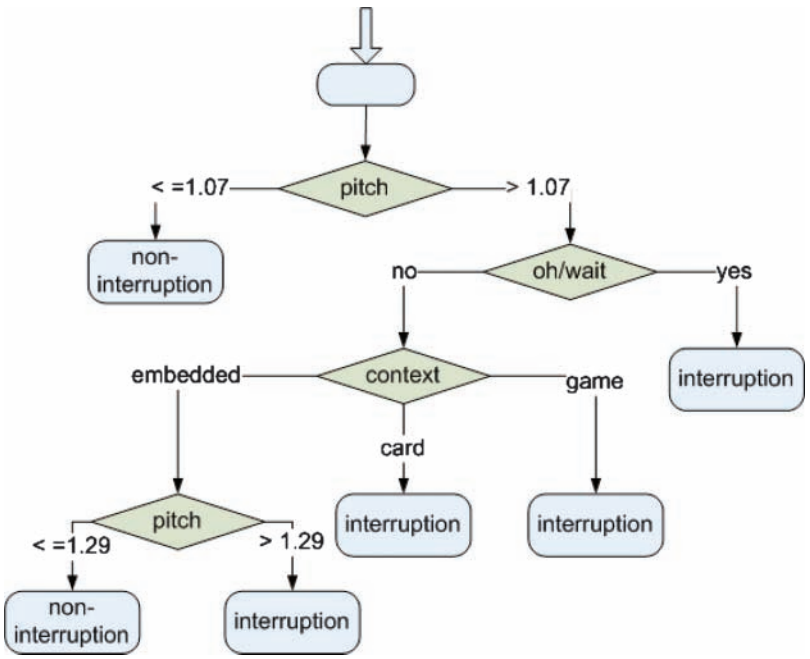


Figure 9
The learned decision tree.

7.2 Recognizing Task Interruptions on All Utterances

The previous experiment helped us determine which features are useful for recognizing task interruptions. However, the experiment was based only on utterances that start with *do you have*, yet not all task interruptions are initiated with *do you have*. We thus conduct a further machine learning experiment on recognizing task interruptions involving any utterances. We extend our feature set to help make up for not limiting ourselves to *do you have* utterances. We purposely do not use any lexical features of the current utterance so that our approach can be applied before speech recognition is performed.

We extract the following features for all non-trivial utterances: 1) discourse context: whether the previous utterance is the end of a poker game, the end of a card segment, or in the middle of a card segment;[6] 2) overlap: whether the utterance overlaps with the previous non-trivial utterances; 3) duration: the length in time of the utterance; 4) normalized pitch: the average normalized pitch of the first 100 msec/200 msec/500 msec and the whole utterance (four features); 5) normalized energy: the average normalized energy of the first 100 msec/200 msec/500 msec and the whole utterance (four features); and 6) pitch range: the pitch range of the first 100 msec/200 msec/500 msec and the whole utterance (four features). In total we have 15 features.

The data that we have are highly skewed. We have 208 cases of task interruptions but more than 4,000 non-interrupting utterances. We thus perform down-sampling so that both classes have the same number of data points. In the first down-sampling, which we refer to as **general down-sampling**, we use all 208 cases of task interruptions, and we randomly select 208 non-interrupting utterances. A concern with the general down-sampling is that 82% of the task interruptions are *do you have* questions, and *do you have* questions are only about 2.5% of the non-interrupting utterances. It is unclear whether a classifier trained from such a data set discriminates task interruptions or discriminates *do you have* utterances. Thus in the second down-sampling, which we refer to as **DYH down-sampling**, we use all 208 cases of task interruptions, and we also use all 105 cases of non-interrupting *do you have* utterances, then finally we randomly select 103 other non-interrupting utterances. The DYH down-sampling, however, still has imbalanced *do you have* utterances in the two classes. Thus we further introduce the **Balanced-DYH down-sampling**, in which we use all 105 *do you have* utterances in the poker playing and 38 other (i.e., non *do you have*) utterances in task interruptions, and randomly select 105 *do you have* utterances from task interruptions and 38 other utterances from poker playing. We run the experiments with decision tree learning (C4.5) (Quinlan 1986) and support vector machine (SVM) (Chang and Lin 2001).

We evaluate the performance using general-leave-one-out. The procedure of down-sampling and general-leave-one-out is repeated 10 times, and then we calculate the average performance. Note that in our evaluation, the distribution of task interruption (which is 50%) is different from the true distribution in the corpus (which is less than 5%). We adopt some metrics from medical diagnostic tests that do not involve prior distributions. **Sensitivity** is defined as $TruePositive/(TruePositive + FalseNegative)$, which, in our case, is the recall of task interruptions. It measures the percentage of task interruptions that the classifier correctly identifies as such. **Specificity** is

6 Card and game segments could be determined fairly accurately from the mouse clicks even without the speech.

defined as $TrueNegative/(TrueNegative + FalsePositive)$, which, in our case, is the recall of non-interruptions. It measures the percentage of non-interruptions that the classifier correctly identifies as such. These two metrics can then be combined using the **likelihood ratio**, which provides a direct estimate of how much a prediction will change the odds. The likelihood ratio for a positive result (LR+) is defined as $\mathbf{LR+} = sensitivity/(1 - specificity)$. It tells us how much the odds of a task interruption increase when the classifier predicts positive (task interruption). The likelihood ratio for a negative result (LR−) is defined as $\mathbf{LR-} = specificity/(1 - sensitivity)$. It tells us how much the odds of a task interruption decrease when the classifier predicts negative (non-interruption).

Table 6 shows the results. If we assume a naive baseline with no knowledge, its sensitivity and specificity are both 50%, and LR+ and LR− are both 1.0. For all three down-sampling settings, SVM performs slightly better than C4.5, and both are much better than the baseline. The result for SVM with general down-sampling shows how well we can recognize task interruptions for our MTD domain, for which we achieve a sensitivity of 78.6% and a specificity of 76.9%. For the Balanced-DYH down-sampling, in which we have the same number of *do you have* utterances in both the classes, SVM cannot make use of the features that distinguish *do you have* from other utterances. Hence, its result might be more indicative of performance in other domains, where task interruptions might not be marked by the same introductory words. Even here, we obtain a sensitivity of 75.3%, a specificity of 75.8%, and 3.11 in LR+ and 3.07 in LR−, which is more than a 50% relative error reduction over the baseline.

Overall, our results show that non-lexical features are useful for the recognition of task interruptions. Because the features used in our machine learning experiments do not require the lexical information of the current utterance, we can make use of the identification of task interruptions to benefit automatic speech recognition (ASR). For example, we can build two language models, one for the ongoing task, and one for the real-time task. For each utterance, we can calculate the likelihood of the utterance being a task interruption, using the decision tree classifier or the SVM classifier. We can then use this likelihood to dynamically interpolate the two language models in the speech decoding. This should be able to improve the accuracy of ASR, which we leave for future work.

8. Conclusion

In this article we describe a series of empirical studies of human–human multi-tasking dialogues, where people perform multiple verbal tasks overlapped in time. We first

Table 6
Performance for non-lexical features.

	Sensitivity	Specificity	LR+	LR−
Baseline	50.0%	50.0%	1.0	1.0
C4.5 + general down-sampling	77.5%	75.3%	3.14	3.35
C4.5 + DYH down-sampling	72.9%	73.2%	2.72	2.70
C4.5 + B-DYH down-sampling	69.4%	71.8%	2.46	2.35
SVM + general down-sampling	78.6%	76.9%	3.40	3.59
SVM + DYH down-sampling	78.6%	78.4%	3.64	3.66
SVM + B-DYH down-sampling	75.3%	75.8%	3.11	3.07

examined the discourse context of task interruptions, that is, where conversants suspend the ongoing task and switch to a real-time task. Our analysis shows that people are more likely to wait until the end of a card or game segment for task switching. We then examined the cues that people use to signal task interruptions. We find that task interruptions correlate with certain discourse markers and prosodic variations. More interestingly, the intensity of pitch depends on the discourse context of the task interruption. We next conducted an exploratory study on context restoration in task resumption. We find that when returning to an interrupted task, conversants sometimes re-synchronize the interrupted ongoing conversation by either restating a previous utterance or summarizing the critical information. Finally, our machine learning experiments show that discourse context, pitch, and the discourse markers *oh* and *wait* are useful features to reliably recognize task interruptions; and, more importantly, with non-lexical features one can improve the performance of recognizing task interruptions with more than a 50% relative error reduction over the baseline.

8.1 Disruptiveness of Task Interruption

In our study on multi-tasking dialogues, we distinguish three types of discourse contexts where players suspend the poker player and switch to a picture game. We claim that these discourse contexts differ in terms of players' engagement and memory load in the ongoing task. First, we feel that players are more engaged in the ongoing task during card discussion. In the middle of a card discussion, players actively share information, explore different (potential) poker hands, and decide what to discard if no poker hand is found. Second, we feel that players also have a higher memory load in the middle of a card discussion. Across poker games, players do not have to remember anything; across card segments, players need to remember what cards each other has; while inside of a card discussion, players need to also remember what card is being discussed, and how far they are into deciding which card to discard.

Engagement can be used to explain the intensity of cues in task interruptions. As we found in Section 5, when players interrupt in the middle of a card discussion, they use a higher pitch than in the case when they interrupt at the end of a game or a card, which is also marked with a higher pitch than non-task-switching (during poker playing). According to Miyata and Norman (1986), a more intrusive signal is needed to attract the attention of people heavily engaged in an ongoing task. Sussman, Winkler, and Schröger (2003) found that higher pitch can serve as a more intrusive signal. Thus when interrupting in the middle of a card discussion, the speaker uses higher pitch probably because the hearer is more engaged in the ongoing task.

Memory load can explain the context restoration behavior in task resumptions. As we found in Section 6, after a picture game that is at the end of a game, players smoothly start a new poker game as if nothing happened; after a picture game that is at the end of a card segment, players might sometimes use information summary to remind each other of what cards they have in hand; and after a picture game that is in the middle of a card segment, players might even repeat or clarify the previous utterance that has been said before the interruption. These observations are consistent with the memory load of discourse contexts. If players are interrupted in a discourse context where the memory load is high, because of the limited working memory, players would need to spend extra effort to recover the memory after completing the interruptions.

Engagement and memory can also explain our finding on the discourse context of task interruptions. According to Miyata and Norman (1986), interruptions where

people are deeply engaged in the ongoing task, or where people have a high memory load, should be disruptive. Thus interruptions at the end of a card game are the least disruptive, with those at the end of a card discussion being more disruptive, and those embedded inside of a card discussion being the most disruptive. A more disruptive interruption tends to have a higher cost to the ongoing task. The disruptiveness of interruptions thus explains players' behavior of delaying the picture game. For task interruptions, players do not always switch to a real-time task when it is prompted, but instead they take into account the discourse context of the ongoing task. They strive to switch to a picture game at the end of a (poker) game or a card when possible. According to Clark and Wilkes-Gibbs (1986), players would try to minimize their collaborative effort in dialogue. The reason that players try to avoid interrupting in the middle of a card discussion probably is because such interruptions have a higher cost to the ongoing task, i.e. these interruptions are more disruptive. Delaying the switch to the real-time task is thus used as a tool to reduce the disruptiveness of the switch.

Our studies thus suggest that conversants strive to interrupt at a discourse context where the cost of interruption is low, but if they interrupt in a more intensive context they use stronger cues to mark the more disruptive interruption.

8.2 Implication for Speech Interface Design

By understanding people's conventions in task interruptions and context restoration, we can implement these conventions into a speech interface to allow natural and smooth task switching in human-computer dialogue. Based on our findings, we propose the following principles for building a speech interface that supports multi-tasking dialogue:

- Minimize the disruptiveness of task switching. Delay task switching till the user's engagement and memory load in the ongoing task are low so that the interruption is less disruptive, while still accomplishing the interruption task in a timely matter. Minimizing the disruptiveness reduces the cost of interruptions to the ongoing task.
- Signal task switching. Discourse markers, such as *oh* and *wait*, and prosodic variations, especially high pitch, can be used to signal task switching. These devices help to disengage the user's attention from the ongoing task so that the user is aware of the task switching. Use stronger cues (e.g., higher pitch) when the task switching is more disruptive (i.e., when the user is more engaged in the ongoing task).
- Recognize task switching. The speech interface can make use of non-lexical features, such as contextual information and the user's pitch, together with discourse markers if available, to help recognize the user's initiation of task interruptions. Recognizing task switching helps the speech interface to interpret the user's utterance in the correct context, which should lead to higher speech recognition accuracy and better language understanding.
- Restore context after an interruption. Utterance restatement and information summary are two effective devices. Context restoration is needed, especially after a disruptive interruption where the memory load in the ongoing task is high, in order to help resolve or prevent misunderstandings and forgetting.

8.3 Future Work

There are obviously a lot of open questions regarding multi-tasking dialogue that are not solved in this article. In this research, we only examined a domain where an ongoing task, rich in context, is interrupted by real-time tasks, which are short and simple in nature. Psychological research showed that the complexity of the real-time task and its similarity to the ongoing task play an important role in the disruptiveness of interruptions (Gillie and Broadbent 1989); thus we can vary these factors in future research. First, we can vary the complexity of the real-time tasks; for example, for some interruptions, the player needs to find out whether the other player has a combination of pictures, such as a black square but not a white triangle (■ ∧ ¬△). This will allow us to examine the correlation between the length of interruptions and context restoration. Second, we can use real-time tasks that are less structured, so that people do not mutually know when it ends. This will allow us to examine whether and how people signal task resumptions. Third, we can introduce ambiguity between the ongoing task and the real-time task: for example, to put the card suits (♡♣♢♠) into the picture game, where an utterance such as *do you have a heart?* can belong to either task. This will allow us to see a wider range of task switching behavior.

Furthermore, in this research we do not investigate how multi-tasking dialogue would be affected by a manual-visual task, such as driving. This is an important question, because for hands-busy, eyes-busy situations such as driving, speech interfaces may provide a human–computer interaction modality that interferes the least with the execution of the manual–visual task (Weng et al. 2006; Villing et al. 2008). We expect that the presence of the manual–visual task will even further necessitate a good understanding of the natural and efficient human conventions for managing multi-tasking so as not to adversely affect the manual–visual task.

Finally, it was pointed out that human–computer dialogue is not exactly the same as human–human dialogue—that is, people might change their behavior when talking to a computer (Doran et al. 2001). It will thus be useful to build an actual speech interface for multi-tasking dialogue, or perhaps first simulate such a system with Wizard of Oz experiments, and to examine whether following the principles that we derived from human–human dialogue does lead to improvements.

Acknowledgments

This work was funded by the National Science Foundation under grant IIS-0326496. The authors thank Alex Shyrokov, David Traum, Elizabeth Shriberg, and members of CSLU for helpful discussions. The authors also wish to thank the reviewers for their constructive comments.

References

Ayers, Gayle M. 1992. Discourse functions of pitch range in spontaneous and read speech. Presented at the Linguistic Society of America Annual Meeting. 9–12 January, Philadelphia, PA.

Bangerter, Adrian and Herbert H. Clark. 2003. Navigating joint projects with dialogue. *Cognitive Science*, 27:195–229.

Butterworth, Brian. 1972. Hesitation and semantic planning in speech. *Journal of Psycholinguistic Research*, 4:75–87.

Byron, Donna K. and P. Heeman. 1997. Discourse marker use in task-oriented spoken dialog. In *Proceedings of the 5th EUROSPEECH*, pages 2223–2226, Rhodes.

Chang, Chih-Chung and Chih-Jen Lin. 2001. *LIBSVM: a library for support vector machines*. Software available at `www.csie.ntu.edu.tw/~cjlin/libsvm`.

Clark, Herbert H. and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognitive Science*, 22:1–39.

Cutrell, Edward, Mary Czerwinski, Eric Horvitz. 2001. Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In *Proceedings of INTERACT*, pages 263–269, Tokyo.

Doran, Christine, John Aberdeen, Laurie Damianos, and Lynette Hirschman. 2001. Comparing several aspects of human–computer and human–human dialogues. In *2nd SigDial Workshop on Discourse and Dialogue*, pages 1–10, Aalborg, Denmark.

Gillie, Tony and Donald Broadbent. 1989. What makes interruptions disruptive? A study of length, similarity, and complexity. *Psychological Research*, 50(4):243–250.

Gopher, Daniel, Yaakov Greenshpan, and Lilach Armony. 1996. Switching attention between tasks: Exploration of the components of executive control and their development with training. In *Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting*, pages 1060–1064, Santa Monica, CA.

Gravano, Agustin, Stefan Benus, Julia Hirschberg, Shira Mitchell, and Illa Vovsha. 2007. Classification of discourse functions of affirmative words in spoken dialogue. In *Proceedings of INTERSPEECH*, pages 1613–1616, Antwerp, Belgium.

Grosz, Barbara J. and Julia Hirschberg. 1992. Some intonational characteristics of discourse structure. In *Proceedings of 2nd International Conference on Spoken Language Processing*, pages 429–432, Banff.

Grosz, Barbara J. and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Heeman, Peter A. and James F. Allen. 1995. The Trains 93 dialogues. Trains Technical Note 94-2, Department of Computer Science, University of Rochester.

Heeman, Peter A., Fan Yang, Andrew L. Kun, and Alexander Shyrokov. 2005. Conventions in human–human multithreaded dialogues: A preliminary study. In *Proceedings of Intelligent User Interface*, pages 293–295, San Diego, CA.

Heritage, John. 1984. A change-of-state token and aspects of its sequential placement. In J. Maxwell Atkinson and John Heritage, editors, *Structures of Social Action: Studies in Conversation Analysis*. Cambridge University Press, chapter 13, pages 299–345.

Hess, Stephen M. and Mark C. Detweiler. 1994. Training to reduce the disruptive effects of interruptions. In *Proceedings of the Human Factors and Ergonomics Society 38th Annual Meeting*, pages 1173–1177, Nashville, TN.

Hirschberg, Julia and Diane Litman. 1987. Now let's talk about now: Identifying cue phrases intonationally. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 163–171, Stanford, California.

Hirschberg, Julia and Christine H. Nakatani. 1996. A prosodic analysis of discourse segments in direction-giving monologues. In *Proceedings of 34th Annual Meeting of the Association for Computational Linguistics*, pages 286–293, Santa Cruz, CA.

Iyer, Rukmini M. and Mari Ostendorf. 1999. Modeling long distance dependence in language: Topic mixtures versus dynamic cache models. *IEEE Transactions on Speech and Audio Process*, 7(1):30–39.

Kun, Andrew L., W. Thomas Miller, and William H. Lenharth. 2004. Computers in police cruisers. *IEEE Pervasive Computing*, 3(4):34–41.

Larsson, Staffan. 2003. Interactive communication management in an issue-based dialogue system. In *Proceedings 7th Workshop on the Semantics and Pragmatics of Dialogue*, pages 75–83, Saarbrücken.

Lemon, Oliver and Alexander Gruenstein. 2004. Multithreaded context for robust conversational interfaces: Context-sensitive speech recognition and interpretation of corrective fragments. *ACM Transactions on Computer-Human Interaction*, 11(3):241–267.

Linde, Charlotte and Joseph Goguen. 1987. Checklist interruption and resumption: A linguistic study. Technical Report CR-177460, National Aeronautics and Space Administration.

McFarlane, Daniel C. 1999. Coordinating the interruption of people in human–computer interaction. In *Proceedings of INTERACT*, pages 295–303, Edinburgh, Scotland.

Meiran, Nacshon, Ziv Chorev, and Ayelet Sapir. 2000. Component processes in task switching. *Cognitive Psychology*, 41:211–253.

Miyata, Yoshiro and Donald A. Norman. 1986. Psychological issues in support of multiple activities. In D. A. Norman and S. W. Draper, editors, *Participant Centered Design: New Perspectives on Human Computer Interaction*. Lawrence Erlbaum, Hillsdale, NJ, chapter 13, pages 265–284.

Moser, Megan and Johanna D. Moore. 1995. Investigating cue selection and placement in tutorial discourse. In *Proceedings of 33rd Annual Meeting of the Association for Computational Linguistics*, pages 130–135, Cambridge, MA.

Nakajima, Shin'ya and James F. Allen. 1993. A study on prosody and discourse structure in cooperative dialogues. TRAINS Technical Note 93-2, University of Rochester, Rochester, NY.

Passonneau, Rebecca J. and Diane J. Litman. 1997. Discourse segmentation by human and automated means. *Computational Linguistics*, 23(1):103–139.

Quinlan, J. R. 1986. Induction of decision trees. *Machine Learning*, 1(1):81–106.

Renaud, Karen. 2000. Expediting rapid recovery from interruptions by providing a visualisation of application activity. In *Proceedings of OzCHI*, pages 348–355, Sydney.

Rickel, Jeff, Stacy Marsella, Jonathan Gratch, Randall Hill, David Traum, and William Swartout. 2002. Towards a new generation of virtual humans for interactive experiences. *IEEE Intelligent Systems*, 17(4):32–38.

Schiffrin, Deborah. 1987. *Discourse Markers*. Cambridge University Press.

Shyrokov, Alexander, Andrew Kun, and Peter Heeman. 2007. Experiments modeling of human–human multi-threaded dialogues in the presence of a manual–visual task. In *Proceedings of 8th SIGdial Workshop on Discourse and Dialogue*, pages 190–193, Antwerp.

Strayer, Susan E., Peter A. Heeman, and Fan Yang. 2003. Reconciling control and discourse structure. In J. van Kuppevelt and R. W. Smith, editors, *Current and New Directions in Discourse and Dialogue*. Kluwer Academic Publishers, Dordrecht, chapter 14, pages 305–323.

Sussman, E., I. Winkler, and E. Schröger. 2003. Top–down control over involuntary attention switching in the auditory modality. *Psychonomic Bulletin & Review*, 10(3):630–637.

Swerts, Marc. 1995. Combining statistical and phonetic analyses of spontaneous discourse segmentation. In *Proceedings of the 12th ICPhS*, volume 4, pages 208–211, Stockholm.

Swerts, Marc. 1998. Filled pauses as markers of discourse structure. *Journal of Pragmatics*, 30:485–496.

Swerts, Marc and Mari Ostendorf. 1995. Discourse prosody in human–machine interactions. In *Proceedings of ESCA Workshop on Spoken Dialogue Systems: Theories and Applications*, pages 205–208, Visgo.

Toh, Siew Leng, Fan Yang, and Peter A. Heeman. 2006. An annotation scheme for agreement analysis. In *Proceedings of 9th International Conference on Spoken Language Processing*, pages 201–204, Pittsburgh, PA.

Traum, David and Jeff Rickel. 2002. Embodied agents for multi-party dialogue in immersive virtual world. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-agent Systems*, pages 766–773, Bologna.

Villing, Jessica. 2010. Now, where was I? Resumption strategies for an in-vehicle dialogue system. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 798–805, Uppsala.

Villing, Jessica, Cecilia Holtelius, Staffan Larsson, Anders Lindström, Alexander Seward, and Nina Åberg. 2008. Interruption, resumption and domain switching in in-vehicle dialogue. In *Proceedings of the 6th International Conference on Advances in Natural Language Processing*, pages 488–499, Berlin.

Walker, Marilyn A. 1996. The effect of resource limits and task complexity on collaborative planning in dialogue. *Artificial Intelligence Journal*, 85:181–243.

Walker, Marilyn A., Rebecca Passonneau, and Julie E. Boland. 2001. Quantitative and qualitative evaluation of DARPA communicator spoken dialogue systems. In *Proceedings of the Association of Computational Linguistics*, pages 515–522, Toulouse, France.

Weng, Fuliang, Sebastian Varges, Badri Raghunathan, Florin Ratiu, Heather Pon-barry, Brian Lathrop, Qi Zhang, Harry Bratt, Tobias Scheideck, Kui Xu, Matthew Purver, and Rohit Mishra. 2006. CHAT: A conversational helper for automotive tasks. In *Proceedings of 9th International Conference on Spoken Language Processing*, pages 1061–1064, Pittsburgh, PA.

Yang, Fan and Peter A. Heeman. 2009. Context restoration in multi-tasking dialogue. In *Proceedings of 13th International Conference on Intelligent User Interfaces*, pages 373–377, Sanibel, FL.

Yang, Fan and Peter A. Heeman. 2010. Initiative conflicts in task-oriented dialogue. *Computer Speech and Language*, 24:175–189.

Yang, Fan, Peter A. Heeman, and Andrew Kun. 2008. Switching to real-time tasks in multi-tasking dialogue. In *Proceedings of International Conference on Computational Linguistics*, pages 1025–1032, Manchester.