Control in Task-Oriented Dialogues

Peter A. Heeman, Fan Yang, Susan E. Strayer

Computer Science and Engineering
OGI School of Science and Engineering
Oregon Health & Science University
20000 NW Walker Rd., Beaverton OR 97006

heeman@cse.ogi.edu yangf@cse.ogi.edu susan_strayer@yahoo.com

Abstract

In this paper, we explore the mechanisms by which conversants control the direction of a dialogue. We find further evidence that control in task-oriented dialogues is subordinate to discourse structure. The initiator of a discourse segment has control; the non-initiator can contribute to the purpose of the segment, but this does not result in that person taking over control. The proposal has important implications for dialogue management, as it will pave the way for building dialogue systems that can engage in mixed initiative dialogues.

1. Introduction

The dialogue manager of a spoken language system is responsible for determining what contributions a system can make and when it can make them. For instance, in a collaborative problem solving domain, the system has to decide if it should take the lead and propose possible solutions, or if it should let the user take the lead. There are conventions of human dialogue that govern this behavior. These conventions help ensure that dialogue runs smoothly, without both participants continually trying to take the conversation in opposite directions.

Two separate theories in computational linguistics have emerged to specify the conventions that speakers follow in determining the direction of a dialogue. The first is discourse structure theory, as proposed by Grosz and Sidner [4]. In their theory, dialogue is composed of a hierarchical linguistic structure consisting of discourse segments. Discourse segment boundaries can be identified by pause lengths, speech rate, intonation, and discourse markers, such as "anyway", and "by the way". Each discourse segment has a discourse segment purpose that is determined by the person who initiates the segment. The two speakers then work together to achieve the discourse segment purpose. The theory, however, does not comment on the types of utterances that the initiator of the segment or the non-initiator can make. Hence, it underspecifies what speakers can do in dialogue.

Control theory, on the other hand (or initiative), aims to account for which speaker is driving the conversation at any given point [13, 12, 2]. For example, in a question-answer pair, the speaker asking the question is said to have control during the exchange. Statements, questions and commands demonstrate control by the speaker, while acknowledgments, answers and agreements do not. Whittaker and Stenton [13] segmented dialogues at points where control shifts from one speaker to the other. They found that control "did not alternate from speaker to speaker on a turn by turn basis, but that there were long sequences of turns in which control remained with one speaker," lasting on average eight speaker turns. As such, control tends to have momentum: the person in control tends to continue showing control. However, control theory has yet to indicate when control can shift between the speakers.

In this paper, we build on our previous work in determining the conventions that govern how speakers determine the direction of a dialogue [11]. In that work, we gave preliminary evidence that control theory and discourse structure theory are not unrelated; rather, control is subordinate to discourse structure theory. The person who initiates the discourse segment tends to stay in control. When the non-initiator shows control, control almost always reverts immediately back to the initiator of the discourse segment.

This paper extends our previous results. We not only look at what happens after utterances by the non-initiator that show control, but also what happens after embedded segments. If the non-initiator of the segment initiates an embedded segment, once the embedded segment ends, control returns to the initiator of the parent segment. Hence, just like utterances by the non-initiator that show control, the control is fleeting, and tends to revert immediately back to the initiator of the segment. These results refine our understanding of control in dialogue: control is subordinate to discourse structure theory.

In the rest of this paper, we describe our schemes for coding discourse structure and control. We then describe the annotation process and our intercoder reliability. We

The authors acknowledge funding from the Intel Research Council, and thank members of CHCC and CSLU for helpful discussions.

then present our evidence that control is subordinate to discourse structure.

2. Coding Schemes

In this section, we introduce our coding scheme. The coding scheme was used for coding dialogues from the Trains corpus of human-human task-oriented dialogues [6]. In these dialogues, two participants work together to formulate a plan involving the manufacture and transportation of goods. One speaker, the user (u), has a goal to solve, and the other speaker, the system (s), knows the detailed information involved in how long it takes to ship and manufacture goods.

2.1. Discourse Structure

In coding discourse structure segments, we followed Carletta et. al [1] by distinguishing between two types of segments. The first are dialogue games. Dialogue games are related to adjacency pairs from conversational analysis [9], and include greeting-greeting, question-answer, check-agreement, and command-response pairs. Dialogue games typically have a first-part and a second-part. Each part can have one or more utterances. Hence, we explicitly mark the second part as a special type of segment. Second-part segments are unique though in that control is explicitly given to the speaker of the second-part by the person who initiated the dialogue game.

The second type of discourse segments are *transaction blocks*, which are related to the task of the dialogues. Here, we required that the proposed initiator of the block made the purpose of the block clear in the first utterance. Cases where a speaker has a purpose in mind but doesn't communicate that there is a purpose were not counted as a transaction block.

2.2. Control and Utterance Tags

We did not directly code which utterances show control. Instead, we tagged each utterance with a variation of the DAMSL coding scheme [3]. Utterances were tagged as forward or backward looking, or other. Forward functions included statements, questions, checks and suggestions. Backward functions included agreements, answers, acknowledgments, repetitions and completions. Other included stalls, incomplete and abandoned.

We used the utterance tags to derive whether an utterance showed control. We found that in general forward functions mapped to utterances demonstrating control in the scheme of Whittaker and Stenton [13] and backward functions and others mapped to utterances not demonstrating control. For utterances that do not demonstrate control, control is said to belong to the last speaker that demonstrated it.

Of special concern is how we dealt with utterances in the second parts of dialogue games. The second-part block already indicates the backward function of the utterances in the second-part. Hence, answers and agreements of in the second-part by the responder are treated as showing control. If the asker acknowledges them or shows agreement, those utterances are treated as having a backward looking function, and so not showing control.

3. Annotating the Corpus

We annotated twelve dialogues totaling 50 minutes from the TRAINS corpus using DialogueView [7]. We started with the distributed word annotations [6]. We then segmented the speech into utterance units, annotated the disfluencies, and marked abandoned and incomplete utterances. We compared and resolved the discrepancies using ACT [14]. For the utterance segmentation, we found that we would sometimes arbitrarily choose one segmentation over another. Two of the authors annotated each dialogue for discourse structure. They then compared their annotations to reach a consensus annotation. Most of the differences between the coders were resolved successfully, though there continued to be differences in the higher level segments.

Halfway through the coding effort, we compared our segment annotations before doing consensus annotation. Intercoder reliability for the two dialogues coded independently showed 43 hits, where discourse segments for both coders were identical in extent and segment initiator. There were 26 misses, where the first coder coded blocks not included by the second coder, and 39 false positives, where the second coder coded blocks not included by the first coder. These figures yield a recall rate of 62% and a precision rate of 52%. In examining the missing and false positive blocks, we found that 13 of the misses corresponded very closely to 13 of the false positives, differing by only one utterance in their beginning or end, and always having the same initiator. If we include these as hits, our recall and precision rates increase to 81% and 68% respectively. We further examined the blocks that were scored as misses and false positives. Similar to parser evaluations [5], we looked at crossing segments, where a segment from one coder overlaps with a segment from the other coder, but neither is properly contained in the other. We will refer to such segments as being inconsistent with the other coder's segmentation. Obviously, inconsistent segments indicate major disagreement about the structure of the discourse. Only 3 of the 26 misses and 2 of the 39 false positives were inconsistent with the other annotator's segmentation.

For annotating utterance tags, we followed the same methodology as for discourse segmentation. For intercoder reliability, our primary focus was whether we agreed as to whether an utterance demonstrated control or not. Using our mapping for the DAMSL scheme, we found that our agreement on the two dialogues as to which utterances show control was 92%.

4. Control in Discourse Segments

Walker and Whittaker [12] suggested that changes in control correspond to changes in discourse structure, but they did not determine the exact relationship between them. At the beginning of a discourse segment, the initiator makes an utterance that shows control. But control does not stay with the initiator throughout the segment, with the non-initiator just answering questions, agreeing, and acknowledging. The non-initiator does occasionally make an utterance that shows control. Figure 1 gives an excerpt of a discourse segment initiated by the user. One

Init	Ctrl	Spkr		Utterance
u	u	u	u1	and then go to Dansville
	s	s	u2	and that's one more hour
		u	u3	yeah
	u	u	u4	we can . . .
	s	s	u5	. . . drop off at the . . .
	u	u	u6	drop off that boxcar drop off the boxcar of . . .
	s	s	u7	and then take two empty ones
		u	u8	right
	u	u	u9	two empty ones down to to Avon
	u	u	u10	and pick up the the bananas
		s	u11	right

Figure 1: Example Discourse Segment (d93-19.5)

special case of how the non-initiator might show control is through a collaborative completion [8], where a speaker finishes (or helps to finish) an utterance of the other speaker. Utterance u5 is an example of the non-initiator making a collaborative completion. Whether such utterances show control is debatable. The DAMSL annotation scheme included these in the backward-looking functions. In previous work [11], we argued that they show control, but at the utterance level. In this paper, we do not include them as showing control. The more general case is shown in u2, where the non-initiator says how long long the operation will take, and in u7, where he suggests a step in the plan. Here, the non-initiator added content that contributed to the discourse segment purpose that is not predicted from the initiator's speech. We refer to these as *co-contributions*.

Co-contributions should be expected, as these are collaborative dialogues where the two speakers work together to achieve a common goal. There are times where the two speakers are so closely synchronized that they build on each other's utterances. In fact, we think this phenomenon of co-contributions is related to the phenomena that Schiffrin [10] referred to as *shared turns*.

4.1. Frequency of Co-contributions

Given that co-contributions can occur, we examined how often they occur with respect to control utterances by the initiator. In total, the initiator made 783 utterances that show control (including utterances inside of second-parts). Of these, 327 were the first utterance in the block (including second-parts). The first utterance of every block by definition will be by the initiator, so we exclude those. This gives us 456 times where the initiator shows control after the first utterance. In comparison, the non-initiator made 55 co-contributions. Hence, the initiator is 8.3 times more likely to make an utterance that shows control inside of a block.

4.2. Control After Co-contributions

We next looked at what happens after a co-contribution. Does the non-initiator keep control, or does control revert back to the initiator? To answer this, we looked at what happens after the initiator makes an utterance that shows control versus what happens after the non-initiator does.

For every utterance by the initiator or non-initiator that shows control (collaborative completions were not included), we examined who shows control next, either by making an utterance that shows control, or by initiating an embedded discourse segment. The results are given in Table 1. We restrict our analysis to what happens in the current discourse segment. If there is no utterance that shows control after the utterance in question and there are no discourse segments embedded in the block, we count it as *end block*, and report it in the first row of Table 1. We excluded 118 control utterances that were followed by a second-part block. The person who initiates the dialogue game explicitly gives control to the non-initiator for the second-part block.

	After control utterance by	
	Initiator	Non-initiator
End Block	190	12
Initiator		
- shows control	360	22
- starts subblock	54	5
Non-initiator		
- shows control	33	15
- starts subblock	28	1

Table 1: After a speaker shows control

What happens in the 55 cases where the non-initiator shows control? In twelve cases, the discourse segment ended before either speaker showed control. In 22 cases, the initiator makes a control utterance, and in five cases, the initiator starts an embedded block. In 15 cases, the non-initiator continues by making another co-contribution and in one case, the non-initiator starts an embedded block. Hence, the non-initiator is the next person to show control 37% of the time (16 of 43 times). On the other hand, when the initiator shows control, the next person who shows control is the initiator in 87% of the cases (414 of 475), by either making an utterance that shows control

or by starting an embedded segment.

4.3. Control after Embedded Segments

In the above, we looked at what happens after an utterance that shows control. Here, we look at what happens after an embedded segment ends. As with the above, we compare what happens after an embedded segment made by the initiator of the parent segment, versus what happens after the one by the non-initiator. We excluded embedded segments that are second parts. The results are given in Table 2. We see that the initiator of the parent block starts an embedded segment 154 times while the non-initiator starts one 48 times. More importantly, we see if the embedded segment belonged to the non-initiator, he would keep control after the embedded segment ended 23% of the time (9 of 39 cases), whereas if the embedded segment belonged to the non-initiator, he/she would keep control after the embedded segment ended 84% of the time (117 of 140 cases).

	After embedded segment by	
	Initiator	Non-initiator
End block	14	9
Initiator		
- shows control	54	14
- makes subblock	63	16
Non-Initiator		
- shows control	6	2
- makes subblock	17	7

Table 2: After embedded segments end

5. Conclusion

We proposed that control is subordinate to intentional structure in dialogue. We have backed up this claim by examining utterances that demonstrate control made by the non-initiator of the discourse segment. We found that after these utterances, control returns to the segment initiator in most cases. The same happens after the non-initiator starts an embedded block. These results are contrary to previous theories of control [12, 2] that predict that when a person shows control, that person tends to keep control. Chu-Carroll and Brown [2] even added an extra level of control. Their two levels of control can be accounted for by the embedding of discourse segments inside of each other.

The reconciliation of control and discourse segments means that we now understand how control and dialogue segment purposes are related and have a clearer picture of how both participants can contribute to discourse intentions. This result implies that control in itself does not need to be tracked in a dialogue. Control belongs to the speaker who started the current discourse segment.

6. References

[1] J. Carletta, A. Isard, S. Isard, J. C. Kowtko, G. Doherty-Sneddon, and A. H. Anderson, "The reliability of a dialogue structure coding scheme," *Computational Linguistics*, vol. 23, 1997.

[2] J. Chu-Carroll and M. Brown, "Tracking initiative in collaborative dialogue interaction," in *35th Annual Meeting of the Association for Computational Linguistics*, Madrid, 1997, pp. 262–270.

[3] M. Core and J. Allen, "Coding dialogs with the DAMSL annotation scheme," in *Working notes of the AAAI Fall Symposium on Communicative Action in Humans and Machines*, 1997.

[4] B. Grosz and C. Sidner, "Attention, intentions, and the structure of discourse," *Computational Linguistics*, vol. 12, no. 3, pp. 175–204, 1986.

[5] P. Harrison, S. Abney, E. Black, D. Flickinger, C. Gdaniec, R. Grishman, D. Hindle, B. Ingria, M. Marcus, B. Santorini, and T. Strzalkowski, "Evaluating syntax performance of parser/grammars of English," in *Workshop on Evaluating Natural Language Processing Systems*, Berkeley, CA, 1991, pp. 71–77.

[6] P. Heeman and J. Allen, "The Trains spoken dialog corpus," Linguistics Data Consortium," CD-ROM, April 1995.

[7] P. Heeman, F. Yang, and S. Strayer, "DialogueView: An annotation tool for dialogue," in *Third SIGdial Workshop on Discourse and Dialogue*, 2002, pp. 50–59.

[8] P. Linell, *Approaching Dialogue: Talk, Interaction and Contexts in Dialogical Perspectives*. John Benjamins Publishing, 1998.

[9] E. Schegloff and H. Sacks, "Opening up closings," *Semiotica*, vol. 7, pp. 289–327, 1973.

[10] D. Schiffrin, *Discourse Markers*. New York: Cambridge University Press, 1987.

[11] S. Strayer, P. Heeman, and F. Yang, "Reconciling control and discourse structure," in *Current and New Directions in Discourse and Dialogue*, J. van Kuppevelt and R. Smith, Eds. Kluwer, 2003.

[12] M. Walker and S. Whittaker, "Mixed initiative in dialogue: An investigation into discourse segmentation," in *28th Annual Meeting of the Association for Computational Linguistics*, 1990, pp. 70–78.

[13] S. Whittaker and P. Stenton, "Cues and control in expert client dialogues," in *26th Annual Meeting of the Association for Computational Linguistics*, 1988, pp. 123–130.

[14] F. Yang, S. Strayer, and P. Heeman, "ACT: a graphical dialogue annotation comparison tool," in *ICSLP*, 2002, pp. 1553–1556.