Toward Measurement of Conversational Interactivity in COMPS Computer Mediated Problem-Solving Dialogues

Michael Glass
Valparaiso U.
michael.glass@valpo.edu

Jung Hee Kim
North Carolina A&T State U.
jungkim@ncat.edu

Kelvin Bryant
North Carolina A&T State U.
ksbryant@ncat.edu

Melissa Desjarlais
Valparaiso U.
melissa.desjarlais@valpo.edu

Micayla Goodrum
North Carolina A&T
mjgoodru@aggies.ncat.edu

Thomas Martin
North Carolina A&T
tmartin2@aggies.ncat.edu

Abstract

This paper reports on experiments in measuring the general level of conversational interactivity in COMPS problem-solving dialogues. COMPS is a web-delivered computer-mediated problem solving environment for student collaborative exploratory learning. The primary mode of interaction is typed dialogue. We anticipate that the computer will provide a status display to aid the instructor, who is effectively looking over the shoulders of the students as they work. Toward the goal of computer monitoring of conversation quality, we have analyzed dialogue turns for Initiate and Respond dialogue moves as prescribed by Conversation Analysis theory. Many of our dialogues are quite interactive by this measure. However computer tagging of individual turns as Initiate or Respond has proved difficult. Here we show what makes such tagging difficult in our problem-solving environment. We also propose that there are shallow measures of overall interactivity that may correlate with how much the students are responding, without the need to correctly tag individual dialogue turns.

Introduction

The goal of the COMPS project is to provide a computer-aided instrument for collaborative learning of concepts through problem-solving dialogue [Desjarlais, Kim, and Glass, 2012; Kim et al., 2013]. The students mainly engage in typed-chat, though for some problems COMPS has specific problem-related affordances for the students to manipulate. COMPS shows the instructor the conversations in real time, permitting the instructor to intervene.

An unusual feature of the COMPS online chat environment is that students type simultaneously. They can see each other's comments as they are typed in real time. This adds an interactive dimension that even spoken language does not support, since students chat simultaneously without interrupting each other.

We anticipate that the computer will provide a status display to aid the instructor, who is effectively looking over the shoulders of the students as they work. The instructor will be informed of each group's degree of cooperative behavior and progress toward solving their problem.

This paper illustrates annotated dialogues collected using COMPS while students were solving problems related to a Java Swing program. We annotated these dialogues using our own scheme that describes the social style of the contribution: confident assertion, suggestion, asking a question, etc. One aspect of our coding scheme corresponds to Exchange Structure from Conversation Analysis, where each turn of dialogue is analyzed as initiating a new dialogue segment (I) or responding to the initiate turn (R). Although some exchange structure analyses also recognize a follow-up category (F), our own annotations recognize only the I and R categories. From these R- and I-annotated dialogues we have documented different styles of group interaction [Kim et al., 2013; Glass et al., 2013].

Theories of student collaborative learning such as group cognition [Stahl, 2006] and knowledge co-construction [Zhou, 2009] presuppose that students are responding to each others' utterances. From our annotated transcripts, we see that COMPS conversations have a high percentage of respond moves. Being able to machine-tag individual turns as respond or initiate would be a step toward judging whether collaboration was happening in our sessions.

In this paper we discuss the issue of building machine classifiers to recognize whether each turn represents an I or R dialogue move. Our efforts have been informed by efforts to classify transactivity [Rosé et al., 2008; Ai et al., 2010]. Transactivity is a classification of R dialogue turns specialized for collaborative learning. However our efforts at building classifiers have not been successful to date. We discuss here possible reasons, and suggest directions for future work.

We also discuss another path toward recognizing whether student conversations contain a high level of interaction: counting the easily recognizable phenomena that occur in conversation when people are responding to each other. These dialogue phenomena are independent of the particular domain under discussion. We propose that measuring the general level of interaction, without tagging individual I and R dialogue turns, might be sufficient to give a rough measure of quality. In this paper we identify and computer-tabulate several of these phenomena occurring

within COMPS dialogues, suggesting that a general measure of interactivity might be possible.

Background

The learning task. The data for this study come from a second semester Java programming class at North Carolina A&T. The protocol was as follows. During lab period, students logged into the COMPS web page in groups of 3. They solved problems in understanding a Java Swing graphical user interface. The problems were presented to the students on paper, accompanied by a picture of the GUI with its components numbered. The nature of the task was to understand and articulate the Java software structure that necessarily lay behind the interface they were seeing. For example, they needed to decide which of the visible components could be anonymous in the code, which event listeners must be present in order to support the desired behaviors, and what is the visibility of instance variables in certain Java classes. The questions exercise their ability to understand Java Swing.

The students were instructed to come to an agreement on answers. One student would take the answer to the professor for feedback in person, then return to the group to finish the discussion. This process continued for each problem until all problems were understood by all members of the group.

Theoretical justification for using COMPS for this kind of learning task. The student skills that are the focus of this project are oriented toward understanding and manipulating concepts. This is what Skemp [1987] calls "relational understanding," a complement to the instrumental skills of programming that are the bread and butter of elementary programming classes. COMPS exercises, such as this one, are focused on learning things that are hard to measure. This orientation guides the construction of our exercises, in particular having students come to shared agreement, telling them the answers, and having them reconcile their understanding with the given answers.

There is also research showing that collaborative activity is a desirable pedagogical approach specifically for creating conceptual understanding [Tchounikine et al., 2010]. Key to engendering learning is dialogue that engages in domain reasoning, such as explaining, negotiating, or inferring [Stahl, 2006]. Justifying, arguing, and similar knowledge-engendering dialogue moves were notable in the Virtual Math Team dialogues [Zhou, 2009].

Collaborative discourse is also, in theory, a fertile application for applying computers to analyze student knowledge and behavior. When student thinking is naturally expressed in the conversation it is made available for the computer to find it. Working in groups forces student thinking out in the open, for example as observed by Koschmann [2011]. In addition to reasoning together, conversational participants also communicate their level of understanding to achieve grounding and to satisfy discourse obligations [Clark and Brennan, 1991]. There is no need for the computer to ask special assessment questions, for example, because student thinking is visible.

The construct representing interactivity. To determine whether a student conversation is interactive, we propose to look for transactivity. In educational dialogue analysis, a dialogue move is transactive if a) it responds to another dialogue move, and b) contributes to knowledge building. If we can identify by machine in two separate procedures that a student's utterance a) responds to another student and b) is on task, we will have approximately identified a transactive contribution. In this paper we are largely concerned with tagging the first aspect, whether a turn responds to another.

Transactivity appears in Weinberger and Fischer's [2006] four dimensional framework for group cognition. Transactivity is the "social mode" dimension: it categorizes in what ways interpersonal processes are at work in the construction of the answer without addressing the knowledge or reasoning. The categories of transactive contribution are: externalization (simply contributing), eliciting, quick consensus building, integration-oriented consensus building, and conflict-oriented consensus building. These categories seem to be on a scale of less transactive to more transactive [Teasley, 1997; Weinberger and Fischer, 2006]. We hypothesize that for the purposes of assessing a conversation, simply measuring the degree of transactivity could be useful. It may not be necessary to specifically identify the above different varieties.

As a way toward annotating transactivity we turn to the linguistic discipline of Conversation Analysis (CA). CA analyzes the exchange structure of a dialogue, dividing up the turns into three types: initiate (I), respond (R), and sometimes followup (F). These basic structural units of dialogue are the workhorse for analyzing phenomena such as turn-taking (how people arbitrate who will speak next), social loafing (who is not participating, or being lazy), and power relationships [Wells, 1999]. Followup is sometimes omitted; these turns can be thought of as additional responses.

Conversation Analysis belongs to the structuralist branch of linguistics; it is concerned with observables first (whether somebody is responding), not what function is being accomplished or what the speaker's intention is. In this aspect annotating initiate and respond is similar to analyzing transactivity.

There is a caveat: I/R/F can be hard to analyze in conversations where there are more than two participants. When there is only one other person deciding which statement a person might be responding to is easier. Also, in a many-party conversation a single statement might elicit several responses from different participants.

Data and Manual Analysis

We conducted 17 COMPS problem-solving dialogues over two semesters with the Java Swing problem. Students were in the General Engineering 165 class at North Carolina A&T, the second semester of elementary programming. Statistics on the dialogues are:

- Sessions: 17
- Dialogue turns: 1827
- Turns per session: 107
- Mean / median duration: 50min / 52min
- Min / max duration: 26min / 67min

Three of these dialogues have been extensively annotated by hand. The annotation categories have been revised since our earlier work [Kim et al., 2013] to a) more accurately match the judgments of the annotators, b) include conversation analysis I or R variants of most categories. The annotation categories are in Tables 1 and 2. Figure 1 (at end) shows an extract of annotated dialogue.

In Figure 1, dialogue turns marked '<<' are not categorized as I or R. The annotations with a hyphen '-' suffix are I, the other annotations are R.

The following annotation categories were devised by our student annotators after they and previous students had some experience with the categories of transactivity outlined above. Essentially the difference between the commonly used transactivity codes and our codes is that in our codes the perceived affect of the speaker substitutes for the social construction of reasoning. For example, the coders felt they could more reliably distinguish whether a speaker was being hesitant or confident, as opposed to distinguishing whether a contribution was more integration-oriented or conflict-oriented.

Table 1: Mode of participation: response categories.

Response	R	A statement that refers to one made earlier
Acknowledgement	A	Cosigning on a message/definitive//suggestion
Contradictory	C	Response that disagrees with a message
Definitive	D	Response that confidently gives a solution
Suggestion	S	A less confident possible solution
Group Work	G	Group working together
Question	Q	Someone asking for clarification or stating confusion

Table 2: Mode of participation: Initiate categories.

Initiate	I-	General start of a new thread
Definitive	D-	A sure answer to a question or problem
Suggestion	S-	A less-sure answer to a question or problem
Question	Q-	A request for feedback/statement of confusion
Group Work	G-	Group working together

If successive turns respond to each other or build on each other serially, we annotate them as a string of responses. In other words, turn *i*+2 can respond to turn *i*+1 which responds to turn *i.* This differs from conventional Conversation Analysis practice which would divide these into a number of Initiate/Respond/Followup exchange segments. One motivation for this departure is the nature of multi-party conversation. In two-party conversation, it is possible to (somewhat arbitrarily) declare that a new segment has started. In multi-party conversations students B and C may both respond to A, or C may respond to B who responded to A. It becomes impossible to isolate initiate-respond pairs without assigning two roles to one turn. For example: B responds to A, while B's same turn simultaneously initiates to C. Motivated in part by that kind of case, we changed the protocol to admit serial response turns. This is also more in line with how transactivity is usually annotated.

Overlapped typing presents another difficulty in annotating I and R. Figure 2 (at end) illustrates overlapped dialogue, specifically turns 5 and 6 from the Figure 1 transcript.

1. Time 2:21: A starts to type "Labels 1, 2, 3, 4, 5, and 14 can be instantiated ...
2. Time 3:16: B types: "what about 6 and 7?"
3. Time 3:48: A finishes typing: "...these do not have to be changed."

Notice that B started asking a question *after* A started. Inspection reveals that B was probably responding to A. But B also *finished* first. B's response to A thus occurs as the earlier dialogue turn in the transcript.

Observations from Manual Analysis

Annotation of the dialogues reveals patterns of group interaction and group cognitive functioning. One phenomenon that is illustrated in the Figure 1 segment is that student C is the primary initiator and serves to set the goal structure of the conversation. Other students largely respond to C's

agenda. This is an example of a pattern we often see [Kim al., 2013; Glass et al., 2013] where one student takes the role of metacognitive regulator for the group cognitive process. This regulator student is not necessarily the one who contributes the most to the solution. Two of the three intensively annotated transcripts illustrate this pattern, visible in Table 3. Student B in both sessions 1 and 2 had the largest fraction of turns in these three-party conversations. Most tellingly, in both discussions large percentages of student B's turns were I (initiate). Student B (marked with *) was driving the conversational agenda, initiating statements into the conversation that the other two students were responding to. Session 3 did not follow this pattern. Session 3 was also unusual in that participant C joined late in the conversation; it was a two-party dialogue for much of its duration. We do not have enough two-party dialogues to say with confidence, but anecdotally it seems that two-party dialogues do not usually follow the same pattern of one person setting the goals.

Table 4 shows the numbers of I, R and off-task turns in each of the three coded sessions and in total. The ratio of R/(I+R) is a responsiveness index: higher numbers mean the students are responding more and initiating less. The lesson to note is that our students are indeed mostly on task and mostly responding to each other.

The number of question turns may also be indicative of group interactive behavior. In our coding scheme, question turns can be either responding or initiating. But any question (except possibly a rhetorical one) is a sign of students engaging with each other. Table 5 shows the numbers of questions, with I and R questions aggregated together.

Table 3: Pattern of one student controlling agenda. Counts of I and R turns only, off-task turns omitted.

Sess no.	Stu	Turns	Stu's pct. of all I+R turns	Pct. of Stu's turns that are I
1	A	23	25%	17%
	B*	48	52%	63%
	C	21	23%	10%
2	A	28	24%	21%
	B*	58	49%	52%
	C	32	27%	34%
3	A	27	40%	52%
	B	31	46%	32%
	C	9	13%	11%

Table 4: Fraction of Responsiveness and On-task Turns.

Sess No.	I	R	Off task	Off task pct.	R/(I+R) pct.
1	36	56	8	8%	61%
2	47	71	17	13%	60%
3	25	42	28	29%	63%
Total	108	169	53	16%	61%

Table 5: Fraction of Question Turns.

Sess No.	Q	Q/(I+R) pct.
1	19	21%
2	27	23%
3	8	12%
Total	54	19%

I/R Classifier

In order to measure whether a dialogue turn is transactive we need to identify whether the turn is a) responding to another person and b) on-task addressing epistemic knowledge-building. We are building classifiers to identify initiate and responding categories first.

Using the hand-annotated transcripts we tried to train Weka J48 decision tree classifiers to recognize I vs. R dialogue turns. In these experiments each training case represented one dialogue turn. Each case contained the following feature set:

- The length of the dialogue turn.
- Presence or absence of each of about 90 common words, chosen for occurring with high frequency in the transcripts.
- Presence of a discourse marker word within the first five words of the turn, chosen from a small set of discourse markers, e.g. "so."
- Presence of one of a dozen vocabulary words specific to the problem domain, e.g. "JPanel."
- Presence of a question mark.
- Predicted class variable: either a code from Tables 1 and 2, or Initiate / Respond / neither.

Decision trees were trained and tested on the approximately 300 annotated turns. The decision trees often overtrained or picked spurious features such as incidental vocabulary words. Thus they did not hold up when applied to held-out test data.

Another set of experiments incorporated timing information as features, using the same classifier methods. When-

ever participant A completed a chat message (by pushing the enter key), we compared A's message to the most recent messages of participant B. This generated four time differences for each record:

A-start-typing – B-start-typing
A-end-typing – B-end-typing
A-end – B-start
A-start – B-end.

In a three-participant conversation, computing time differences A vs. B and A vs. C doubles the number of cases. One set of cases contains the delta-times for A vs. B, the second set is identical except for delta-times A vs. C. Most of the features in the duplicated records, e.g. sentence length, discourse markers, and class variable, remain the same.

The delta-time feature also sometimes revealed cases of simultaneous typing. For example consider turn 6 vs. turn 5 in Figure 2. Turn 5 is the "earlier" turn because it ended earlier, therefore the "later" turn 6 is evaluated as a potential response to 5. However 6 started before 5. The delta time A-start – B-start is thus negative, indicating overlap.

When care is taken to remove duplicate records and identify which turn is responding to which other participant, J48 pruned decision trees utilizing the delta-time features are more robust than the earlier classifier experiments. Applying the trained trees to held-out data works reasonably well.

Results and Discussion

Results. None of the experiments were notably successful.

Using the non-timing features, typical good results using 10-fold cross-validation were kappa agreement of about 0.45 with human raters, and F scores of 0.7 on identifying the I and R labels.

When delta-times were available as features for classifier training the accuracy was about the same. Kappa agreement with the human raters remained in the low end of the 0.4 – 0.5 range, and F scores remained at about 0.7.

The best decision trees using timing features were not startling. If A started 61 or more seconds after B ended, A was most likely not a response to B. But given that A started late, if A's statement was long it was a little more likely a response to B.

Introducing delta-times is an improvement in classification. Even though classification accuracy was not improved, the timing features are potentially domain-independent. The classifier trained on timing features might work for all our COMPS dialogues in three different classes. Whereas a classifier that uses vocabulary might work only for the particular problem or student population it was trained on. The fact that cross-fold validation tended to degrade accuracy in classifiers using the word features more than it degraded classifiers using timing features is another indication that timing features will hold up better with larger and more diverse data sets.

Generating time difference records against every other participant in the conversation proved to be a methodological problem. It biases the class labels. A single I turn is represented by two records in the data set with nearly identical features predicting the same class variable. The result is a strong tendency for the classifier to predict the doubled cases.

Comparison to other results. Other researchers achieve moderately better Kappa between 0.5 and 0.6, e.g. [Rosé et al., 2008] working with online chat discussions and [Ai et al., 2010] working with transcribed classroom discussion. In both cases the class variable was transactivity. They were able to boost Kappa agreement to 0.7 using several stages of classification. It is instructive to analyze some of the differences between their classifiers and ours. In addition to the features we mentioned above, e.g. vocabulary words and lengths, these researchers derived features so that one case (one annotated dialogue turn) would include features contrasting that dialogue turn against previous turns. These derived features were:

1. LSA (latent semantic analysis) comparisons of the words in the current turn to a) the words in the previous turn (usually another speaker), b) two turns back, and c) three turns back.
2. Type of speaker (student or teacher), type of speaker for previous turn, whether the speaker is the same person as for the previous utterance.
3. Change of topic: whether the topic has shifted in the previous utterance.

It also appeared that their chat data did not include overlapped simultaneous typing.

Discussion. Examining Figure 1 shows why, we believe, our classifiers have not been successful to date. A main issue is that much of the dialogue does not contain concepts, students instead refer to multiple-choice answers by letters and to numbered items on the Swing GUI. The concepts and objectives being reasoned about are not situated within the conversation. The letter and number references are more common than sentences containing recognizable reasoning in the domain. As evidence that this complicates the task, we note that it is not possible for a human annotator to tell whether two people are discussing the same concept without a picture of the GUI and the multiple-choice answers handy for reference.

A secondary source of complexity is the typing overlap problem. In addition to the kinds of timing anomaly illustrated in Figure 2, we see students sometimes neglect to press enter. Everybody can see what they typed without it. We see students pause in the middle of typing, wait for other student responses, then pick up again, effectively putting two dialogue turns in one chat message.

The Way Forward 1: More and Better Features

A priority task is to find shallow features that should correlate with either 1) students responding to each other or 2) students reasoning on topic. These are the two components of a transactive contribution. We will use these features to see if the classification task can be improved.

We will also try to use simple detection and counting of these features to derive a transactivity index that correlates with human judgment. This is discussed below.

Features we have extracted from the text but not yet applied to machine learning experiments.

Discourse markers. Using an expanded catalog of discourse markers [Alemany et al., 2005], we see discourse markers start 10% of the 1800 turns from 17 sessions. Discourse markers might indicate that reasoning or argumentation is happening. In addition to the fixed lexicon, we added some discourse markers recognized by regular expressions (e.g. "soooo...").

Problem domain vocabulary. These words are an indication that students are discussing the topic at hand. We expanded the vocabulary of problem domain words for the Java Swing GUI problem. The number of turns that are now recognized as including domain-specific words is 20%, compared to 7% in the machine learning experiments described above.

Task-related vocabulary. These are words related to completing the task but not part of the domain under discussion. For example, the multiple-choice answer letters and the labels on the different components of the Java Swing screenshot are task-related words which we can recognize. If a student says "I think we can rule out b and c", that student is discussing the task at hand. 30% of turns contain a reference to a multiple-choice answer or a numbered component.

Overlapped typing. Among the 1800 turns, 47% exhibit overlapped typing where several people are entering a new message simultaneously.

Emoticons. People put emoticons into their chat dialogue precisely because they are interacting with other people. Emoticons express affective state. Emoticons occur in only 1% of our corpus for this problem but they are much more prevalent in other COMPS exercises using a different student population.

Pronouns. In a similar vein, the presence of 2nd person pronouns and 1st person plural pronouns could be indicative of interactive discourse. 16% of turns contain such a pronoun within the first 10 words.

Other features to explore:

Other expressions of affect. Theories of affect generally hold that people express affect in order for other people to sense it. Expressions of affect, therefore, may be indications of social processes at work. We propose that the presence of such words might be a useful feature.

More use of timing overlap. We may split a turn into two in the event of a lengthy pause, treating the two parts as different dialogue turns. Especially if other people were typing during the pause, it is likely that the two parts serve as distinct dialogue turns. We may use timing differences to try to identify candidate turns as targets of response in a way that does not duplicate records.

Inter-turn comparisons. We can try the LSA comparisons and other measurements on successive turns that other researchers have found fruitful.

The Way Forward 2: A different style of measurement.

We will explore measuring the interactivity of a discussion without labeling each individual turn as transactive or not. Some of the features may by themselves be indicative of students interacting with each other, e.g. emoticons, close timing and overlaps, and pronouns. Other features may be indicative of students engaging in reasoning (discourse markers), of engaging the problem (domain vocabulary) and of engaging the task (task vocabulary). Simply measuring the density of these features might prove sufficient to evaluate the quality of a student problem-solving discussion. This measurement could be applied in real time to the entire discussion starting from the beginning, or to a sliding window of most recent dialogue turns.

For purpose of training a computerized formula for this measurement, we will use the manually coded corpus to independently assess overall transactivity. We will start by counting the fraction of interactive turns in our annotations.

Acknowledgments

Thank you to our hard-working students at North Carolina A&T and Valparaiso Universities. This work is supported by the Lockheed Martin Corporation under the program of Computer Science Undergraduate Researchers to North Carolina A&T State University and by the National Science Foundation under awards 0633953 to North Carolina A&T State University and 0851721 to Valparaiso University.

References

Ai, Hua, Marietta Sionti, Yi-Chia Wang, and Carolyn Penstein Rosé. 2010. Finding transactive contributions in whole group classroom discussions. In Kimberly Gomez, Leilah Lyons, and Joshua Radinsky (eds.), *Proceedings of the 9th International Conference of the Learning Sciences* (ICLS '10). International Society of the Learning Sciences, vol. 1 pp. 976–983.

Alemany, Laura Alonso, Irene Castellón Masalles, and Lluıs Padró Cirera. 2005. *Representing Discourse for Automatic Text Summarization via Shallow NLP techniques*. Unpublished PhD thesis, Universitat de Barcelona. Lexicon of discourse markers in Catalan, Spanish, and English. Retrieved March, 2014 from `http://russell.famaf.unc.edu.ar/~laura/shallowdisc4summ/`

Clark, Herbert H., and Susan E. Brennan. 1991. Grounding in Communication. In Lauren B. Resnick, John M. Levine, and Stephanie.D. Teasley (eds.), *Perspectives on Socially Shared Cognition*. American Psychological Association, pp. 127–149.

Desjarlais, Melissa, Jung Hee Kim, and Michael Glass. 2012. COMPS Computer Mediated Problem Solving: A First Look. *Proceedings of the Midwest AI and Cognitive Science Conference* (MAICS 2012), Cincinnati.

Glass, Michael, Jung Hee Kim, Melissa Desjarlais, and Kelvin S. Bryant. 2013. COMPS Computer-Mediated Problem Solving Dialogues. (Poster abstract) *Proceedings: Computer-Supported Collaborative Learning* (CSCL 2013), Madison, WI, vol II, pp. 257–258.

Kim, Jung Hee, Melissa Desjarlais, Kelvin Bryant, and Michael Glass. 2013. Observations of Collaborative Behavior in COMPS Computer Mediated Problem Solving. *Proceedings of the Midwest AI and Cognitive Science Conference* (MAICS 2013), New Albany, IN, pp. 71–77.

Koschmann, Tim. 2011. Understanding understanding in action. *Journal of Pragmatics.* vol. 43 no. 2, pp. 435-437.

Rosé, Carolyn, Yi-Chia Wang, Yue Cui, Jaime Arguello, Karsten Stegmann, Armin Weinberger, and Frank Fischer. 2008. Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. *International Journal of Computer-Supported Collaborative Learning.* vol. 3 no. 3, pp. 237–271.

Skemp, Richard. 1987. *The Psychology of Learning Mathematics*. Hillsdale, NJ: Erlbaum. Chapter 12.

Stahl, Gerry. 2006. *Group Cognition*. MIT Press.

Tchounikine, Pierre, Nikol Rummel, and Bruce M. McLaren. 2010. Computer Supported Collaborative Learning and Intelligent Tutoring Systems. In R. Nkambo, J.Bourdeau, & R. Mizoguchi (eds.), *Advances in Intelligent Tutoring Systems*. Springer. Chapter 22, pp. 447–463.

Teasley, Stephanie. 1997. Talking about reasoning: How important is the peer in peer collaboration? In L. B. Resnick, R Saljo, C. Pontecorvo, and B. Burge (eds.), *Discourse, Tools, Reasoning: Essays on Situated Cognition*. Springer.

Wells, Gordon. 1999. *Dialogic Inquiry*. Cambridge University Press.

Weinberger, Armin and Frank Fischer. 2006. A framework to analyze argumentative knowledge construction in computer-supported collaborative learning, *Computers & Education*, vol. 46 no. 1, pp. 71–95.

Zhou, Nan. 2009. Question Co-Construction in VMT Chats. In Gerry Stahl (ed.) *Studying Virtual Math Teams.* Springer. pp. 141–159.

Turn	Stu	Text	Annotation	Start time	End time
1	C	hey people	<<	00:00	00:02
2	C	okay question one??	Q-	00:43	00:47
3	B	I'm reading it	<<	01:08	01:15
4	C	do either of you know what the question is even asking? i don't	Q-	01:44	02:10
5	B	what about 6 and 7?	S	03:16	03:26
6	A	"Labels 1, 2, 3, 4, 5, and 14 can be instantiated anonymously. Because these do not have to be changed."	D	02:21	03:48
7	B	that makes sense	A	04:12	04:17
8	A	6 and 7 can not be instantiated anonymously because these values have to change.	C	03:59	04:18
9	C	okay. Im lost where are you guys gettting this from	Q-	04:19	04:43
10	C	the back ground information?	Q-	04:46	04:53
11	A	It's on the second page.	G	04:52	04:56
12	B	the top discription	G	04:58	05:02
13	C	ohhh mow i see thanks	G	05:08	05:16
14	B	for problem two I know the actionlistener interface needs to be implemented	D-	05:40	05:58
15	B	is there any others?	Q-	06:09	06:13
16	C	and actionListener	D	06:27	06:39
17	C	i think those are the only two	D	06:42	06:51
18	B	wha was the first one?	Q	06:55	07:01
19	C	i ment mouse	R	07:02	07:10

Figure 1: Annotated Extract from Java Swing Dialogue

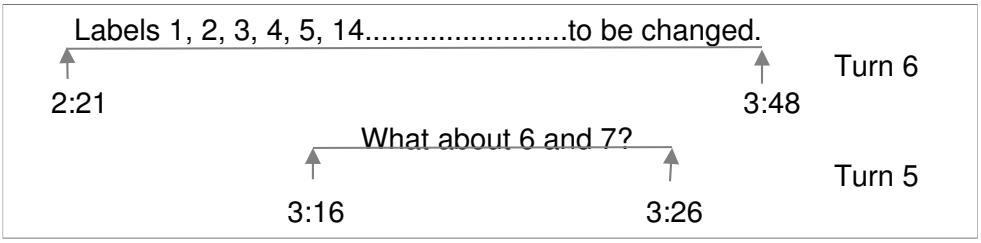

Figure 2: Overlapped Typing of Response.