COME LET US CHAT TOGETHER: SIMULTANEOUS TYPED-CHAT IN COMPUTER-SUPPORTED COLLABORATIVE DIALOGUE [*]

Michael Glass
Dept. of Computer and Information Sciences
Valparaiso University
Valparaiso, IN 46383
219-464-5161
Michael.Glass@valpo.edu

Kelvin Bryant
Dept. of Computer Science
North Carolina A&T State University
1601 Market St
Greensboro, NC 27411
336-285-2446
ksbryant@ncat.edu

Jung Hee Kim
Dept. of Computer Science
North Carolina A&T State University
1601 East Market St
Greensboro, NC 27411
336-285-3695
jungkim@ncat.edu

Melissa Desjarlais
Dept. of Mathematics and Statistics
Valparaiso University
Valparaiso, IN 46383
219-464-5116
Melissa.Desjarlais@valpo.edu

ABSTRACT

This paper describes people chatting at the same time in COMPS group collaborative exercises. Simultaneous typed-chat, students seeing others' typing in real time, is a feature of COMPS. In verbal interaction cooperative dialogue requires turn-taking. People cannot talk simultaneously except for fractions of a second and still conduct dialogue. This paper documents qualitatively that simultaneous chat is happening in the keyboard domain, that students are interactively responding to each other, and that collaborative behaviors occur in this context. COMPS also will incorporate computer recognition of the degree of interactive behavior. This paper also shows evidence that conventionally extracted dialogue turns and features that are

---

used for training text classifiers do not work when applied to simultaneous chat.

INTRODUCTION

The goal of the COMPS (Computer-Mediated Problem Solving) project is to provide a computer-aided instrument for collaborative learning of concepts through problem-solving dialogue [1, 3, 6]. 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 can both type and read without interrupting each other. This paper reports on students who are keyboard-chatting with each other in two different tasks in two different classroom contexts.

The theoretical benefits of computer-mediated collaborative chat hinge on interactivity, on students responding to each other. Most forms of human dialogue require that people take turns as they talk. One goal of this paper is to illustrate that when using a keyboard (as opposed to talking verbally) everybody chatting at the same time occurs in real conversations and is compatible with collaborative discourse.

Future goals of COMPS include machine monitoring of the conversations. This includes training computer models to measure a) whether an individual dialogue turn is responding to another turn by a different student, and b) the degree to which a conversation is interactive. In prior work this project annotated the interactivity of the dialogues. It also trained classifiers to measure whether students are addressing each other. These efforts were not very successful. However these efforts did not succeed in taking into account the simultaneity of student typing [2].

This paper reports on evidence that students are using this full-duplex chat ability in productive ways. It also illustrates how overlapping chat can make it difficult to identify turns that respond to other turns.

THEORETICAL BACKGROUND

Conceptual Knowledge and Dialogue Interactivity

COMPS collaborative problem-solving exercises are designed to address student conceptual knowledge via group work. COMPS exercises have been deployed in two different quantitative literacy exercises for pre-service teachers and in two different object-oriented Java concept exercises in CS2 classes.

The student skills are what Skemp [8] calls “relational understanding,” a complement to the instrumental skills that are the bread and butter of elementary programming and mathematics 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 research showing that collaborative activity is a desirable pedagogical approach specifically for creating conceptual understanding [11]. The Virtual Math Team dialogues illustrate the group cognition behaviors [9] and knowledge co-construction activities [12] underlying this success. In group cognition, the individual bits of knowledge construction can be observed in different participants. They work together to build knowledge. COMPS has also observed group metacognition, where one student regulates the task by posing questions and check answers for plausibility while other students perform the knowledge construction. Key to engendering learning is dialogue that engages in domain reasoning, such as explaining, negotiating, or inferring. Justifying, arguing, and similar knowledge-engendering dialogue moves were also notable in the Virtual Math Team dialogues.

These conversational theories presuppose that students are responding to each others' utterances. Whether students can and do respond to each other while all typing simultaneously is a question that has not been well studied. The definitive studies of students engaging in typed communication, e.g. [5], do not consider this case.

Measuring Interactivity

Accordingly, one goal of this project has been to measure how frequently students respond to each other. For this purpose this project has used an interactivity index, defined as the percentage of on-task turns that respond to other turns [4, 2].

The linguistic discipline of Conversation Analysis (CA) divides dialogue turns into three types: initiate (I), respond (R), and sometimes follow-up (F). CA then identifies exchanges, segments of dialogue that start with I turns [10]. However exchange structure analysis proves to be difficult for dialogues with more than two people. In COMPS conversations a single statement might elicit several responses from different participants. Persons C (turn i + 2) and B (i + 1) can both respond to A (turn i). Also B can respond to A, and then C can respond to B, creating a chain of responses. Furthermore, these two cases may not be distinguishable. The follow-up category is similarly hard to distinguish: C's follow-up to B might look like a response to A. Thus when analyzing dialogues, this project's method is simply to mark R for a turn that responds to the most recent turn of any of the other people. Otherwise it is an I turn. Dialogue turns that are not related to the task are excluded, these are largely the greetings and leave-takings at the beginning and end.

CA exchange structure analysis has proven to have another difficulty: it relies on people taking turns. In COMPS overlapping utterances are common.

EXPERIMENT

The transcripts reported on in this paper came from two different contexts. Context A was a Java reasoning problem in a second semester programming class at North Carolina A&T State University, a public HBCU. Context B was a logical and mathematical reasoning problem in a class for pre-service teachers at Valparaiso University, a private sectarian institution. In both cases the exercises are administered during a lab period. Students logged in to the COMPS web page and communicated via typed-chat with the other members of their group. By protocol, the members of each group were seated apart from each other in the classroom, they did not interact verbally.

For the Java exercise, each student received a picture of a GUI with its components numbered. This was accompanied by a list of questions. 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 both the principles of Java classes and the particular ideas behind Java Swing. The exercise had five multiple choice questions. The exercise protocol was as follows. The students were in three person groups. They were instructed to come to an agreement on answers to one or more of the questions. One student would take the answer to the professor for feedback in person, and 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.

For the logical reasoning exercise, the students were required to discover whether there is a winning strategy for the game of Poison. It is a Nim-like game where players alternate removing 1 or 2 stones from a pile. The player with the last stone loses. The web page supported playing the game online during the chat. The students were typically in four person groups. They were instructed to divide into two teams for playing purposes. They were to chat and play until the answer had been worked out and understood by all members of the group, then agree on what would go into a written description of the answer that one member was charged with writing afterward. In this exercise the professor was logged into the conversations, intervening when it appeared that help would be useful.

	Java	**Poison**
# Sessions	17	5
Dialogue Turns	1790	1088
Mean turns/session	107	218
Mean duration	50 min	72 min
Min duration	26	68 min
Max duration	67	77 min
% of turns overlapping	47%	50%

Table 1. Statistics on dialogues

Statistics on the dialogues are in Table 1. Simultaneous chat was prevalent in these dialogues. Roughly half of all dialogue turns occur while other turns were in progress. We count a turn as in progress from the moment of the first keystroke until the enter-key at the end. We make an exception for long pauses in the middle of a turn. The other students can see what has been typed already, so effectively they are not interrupting if they type after the first student has paused. In order to filter out overlapping with paused turns, keystrokes from the two overlapping turns must occur within 10 seconds of each other in order to be counted.

The Java dialogues were hand-annotated by a team of three students for I (initiate) and R (respond). Each dialogue turn that was not marked as I or R was excluded. The result was 1790 annotated turns. A set of Poison dialogues were likewise coded, but have not been used for machine learning.

ILLUSTRATIONS OF SIMULTANEOUS DIALOGUE

This section illustrates what simultaneous chat looks like, and shows that it is possible for students to keyboard-chat and still engage with each other simultaneously.

Figure 1 from the Poison dialogues illustrates a common case, where student B does not wait for student A to finish before challenging a statement. In this example, B can be seen as possibly providing a metacognitive plausibility check on A's proposed partial answer, claiming that the proposed answer is not consistent with the problem statement.

St	Text		Start - End mm:ss
A	So part of the strategy to win could be going second, so that	you can always match what the other person picks, right?	42:54 - 43:42
B		We need to find a way that you always win, whether you go first or second.	43:30 - 43:41

Figure 1. Overlapping with a Correction

Figure 2 illustrates an extreme example of overlapped Java dialogue. Student B's question happens entirely within the middle of A's utterance, and A answers B's question without pausing. In this exchange, A can be seen as extending the turn to accommodate the question that B raised while A was still typing. It is also an example of how students can respond to each other while chatting simultaneously, something that will not likely happen in verbal conversation.

A	Labels 1, 2, 3, 4, 5, and 14	can be instantiated	anonymously. Because these do not have to be changed

B	What about 6 and 7?

Figure 2. Student A Responding to Student B in Mid-Turn

Sometimes students engage in sequences of turn-taking dialogue, but they start typing early without waiting for the previous turn to finish. Figure 3 shows an example from the Poison dialogues.

Figure 4 illustrates three students typing simultaneously during the turns marked 2, 3, and 4. The first turn is the ending sentence of a putative answer to the question. At turn 2, A starts to explain. In the middle of the explanation B provides an elaboration on A's explanation. Then C expresses agreement. During turns 2, 3, and 4 all three engage in aspects of the knowledge-building activity while typing simultaneously.

A	So long as you do that, you'll hit all our magic	numbers and win.		
B		But in the last game i started	with taking 2 and	still won.
A				You won the game when you had 18 because....

Figure 3. Serial Interruption

	St	Text	Start - End mm:ss
1	A	loan amount, interest rate, num years... etc.	19:11 - 19:23
2	B	well; they're the only ones that change the others can stay the same throughout the whole program	19:26 - 21:22
3	B	that makes sense since they are the only two that arent going to be input by the user	20:36 - 21:01
4	C	i think u are right	20:56 - 21:12

Figure 4. Three Students Chatting Simultaneously

The next example, Figure 5, illustrates the difficulty that overlapping dialogue presents when trying to classify turns as I or R. Recall that for a turn to be classified as R, there is an antecedent turn which is the most recent turn of another person. The utterances of this dialogue are schematically like this. The turns have been marked with I or R:

Turn 1 (I) A proposes possible answer letter (option E)
Turn 3 (R) B proposes alternate answer letters
Turn 4 (R) C proposes yet other answer letters
Turn 2 (I) A explains the proposed answer in turn 1
Turn 5 (R) B acknowledges A's explanation

However Figure 5 is fundamentally different than some of the previous illustrations. The turn numbers appear to be out of order. The turns were numbered from a transcript where the turns were printed in order of start time.

How do we know the end-time ordering is correct and start-time ordering is incorrect in this case? We know this from the semantics of their statements. Student A proposed some letter choices (turn 1), B and C proposed alternate choices (turns 3 and 4). Similarly, B's acknowledgment in turn 5 is coherent only as a response to A's explanation, which was in turn 2.

For the annotator, simultaneous typing made this into a mess. Recall that a response turn should address the most recent turn from the other student.

- While A was typing turn 2, B and C started responding to A's earlier turn 1. It was not possible to correctly annotate 3 and 4 as responses to turn 1, because turn 1 is no longer the most recent turn by A.
- Turn 5 by B responds to turn 2 by A. Again, correctly annotating turn 5 is a problem because of the intervening turn by B.

On the other hand, the exchanges in Figures 1, 3, and 4 are correctly interpreted only if ordered by start time, while in Figure 2 each turn contains parts that respond to the other. We have not found a mechanical way to reliably order the transcripts so each R turn follows its antecedent I.

Turn	**I/R**	**St**	**Text**	**Start - End hh:mm**
1	I	A	On number 2 I believe its E	31:55 - 32:05
3	R	B	for number two I say A, B, D, E	32:18 - 32:36
4	R	C	I think its A B and D	32:32 - 32:42
2	I	A	because ActionListener Only operates on a text field when enter button is pressed	32:06 - 32:49
5	R	B	oh true	33:09 - 33:11

Figure 5. End-Time Ordering for Correct I/R Annotation

PRIOR RESULTS

Toward the goal of developing COMPS computer chat monitoring, two goals were to train models to measure overall interactivity and classify individual turns as I or R. Experiments for these purposes were conducted using the approximately 1800 turns of manually-annotated Java task transcripts.

Interactivity was defined as percentage of R turns: R / (I+R). In the coded Java dialogues, the most interactive of the 17 sessions had interactivity of 72%, the mode was 64%, and the minimum was 49%. Overall 65% of dialogue turns are annotated as responding to other turns [2].

Features were machine-extracted from the dialogue turns for the purpose of training an I/R classifier. Examples of features that might pertain especially to interactivity included question marks (eliciting a response from another student), emoticons, and pronouns "you," "we," and "us" (indicating joint activity). Examples of features that pertain to knowledge construction included discourse marker words such as "now" and "so" at the beginnings of turns (which are indicative of the occurrence of reasoning), references pointing to items on the GUI interface (deixis), and words from the problem domain. Other features were typical of text mining tasks, e.g. length of turn and the individual higher-frequency words [2].

- Using several algorithms, successfully training a classifier to identify I and R turns based on these features proved elusive. The best classifier was a J48 decision tree. Kappa agreement between the best classifier and human annotators was 0.27.
- Linear models predicting the interactivity index of a session based on the prevalence of the features were not successful.

Hand examination of some of the classification failures revealed to us that many of the errors seemed to be associated with simultaneous chat [2].

DISCUSSION AND CONCLUSIONS

Group cognition is in evidence in COMPS dialogues. All the examples in this paper illustrate group cognition, where each student is contributing to epistemic knowledge construction. Furthermore, everybody chatting at once does not seem to be incompatible with group interaction. Figure 1 through 3 show two students interacting constructively while typing simultaneously, while Figures 4 and 5 illustrate the same with three students all typing at the same time. Most of our groups have three or four people.

Regarding manual annotation of turns as I or R, re-examination of our transcripts shows that the overlap between turns should not have been ignored. The annotators worked from transcripts that had been linearly ordered by end time. Start time seems to more frequently produce a sensible linear ordering in both the Java and Poison transcripts. However as the example in Figure 5 shows, start-time ordering is also not always correct. Examining this example is instructive because it appears that determining the antecedent of a response is partly semantic. Turns 3 and 4 can be seen to be responses to turn 1 because semantically, all three are proposing multiple choice answer letters. Given the timing of the students typing, with turns 3 and 4 interrupting turn 2, it would have been entirely possible for the antecedent to have been turn 2. How to display simultaneous-chat transcripts so that they may be reliably hand-annotated for I and R is an open question. Rosé's experiments in classifying transactivity [7] (a transactive dialogue turn is responsive to an earlier turn and contributes to knowledge-building) utilized latent semantic analysis comparison between pairs of turns as one of the machine-derived features. It is conceivable that COMPS should do likewise.

It is also not clear how to derive features for machine learning that capture the aspects of simultaneous chat. One feature is the number of keystrokes in a turn that were typed while other turns were in progress. This feature was virtually unused by the decision tree classifiers and did not figure into any of the interactivity regression equations.

Between two utterances of two students A and B, with time stamps Astart, Aend, Bstart, and Bend, there are four timing differences. Bstart - Aend is the usual gap between two successive conversational turns: after A ended, how long B waited to start. It is negative if B interrupted A. When attempting to classify I and R turns, classifier features included all four differences between the two students' utterance. For example Bstart - Astart is how long B waited after A started before jumping in. In the decision tree classifier experiments, the most prominent features were the gaps between two turns, e.g. a long positive gap was indicative of a new I turn. But the negative numbers that indicate interruptions were hardly predictive.

The summary conclusion is that students know how to chat simultaneously and still work productively, but successfully capturing the simultaneity in features for machine learning remains future work.

REFERENCES

[1] Desjarlais, M, Kim, J. H., Glass, M., COMPS Computer Mediated Problem Solving: A First Look, *Proceedings of the Midwest AI and Cognitive Science Conference* (MAICS 2012), 50-57, 2012.

[2] Glass, M., Kim, J. H., Bryant, K. S., Desjarlais, M., Indicators of Conversational Interactivity in COMPS Problem-Solving Dialogues, *Ingelligent Support for Learning in Groups (ISLG-3)*, 2014.

[3] Glass, M., Kim, J. H., Desjarlais, M., Bryant, K. S., COMPS Computer-Mediated Problem Solving Dialogues, *Proceedings: Computer-Supported Collaborative Learning* (CSCL 2013), Madison, WI, vol. II, 257-258, 2013.

[4] Glass, M., Kim, J. H., Desjarlais, M., Bryant, K. S., Goodrum, M., Martin, T., Toward Measurement of Conversational Interactivity in COMPS Computer Mediated Problem-Solving Dialogues, *Proceedings of the Midwest AI and Cognitive Science Conference* (MAICS 2014) 31-38, 2014.

[5] Jonassen, D.H., Kwon, H.I. Communication Patterns in Computer Mediated vs. Face-to-Face Group Problem Solving, *Educational Technology: Research and Development*, 49 (10), 35-52, 2001.

[6] Kim, J. H., Desjarlais, M., Bryant, K., Glass, M., Observations of Collaborative Behavior in COMPS Computer Mediated Problem Solving, *Proceedings of the*

Midwest AI and Cognitive Science Conference (MAICS 2013), New Albany, IN, 71-77, 2013.

[7] Rosé, C., Wang, Y., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., Fischer, F.. Analyzing Collaborative Learning Processes Automatically: Exploiting the Advances of Computational Linguistics in Computer-Supported Collaborative Learning, *International Journal of Computer-Supported Collaborative Learning*, 3 (3), 237-271, 2008.

[8] Skemp, R., *The Psychology of Learning Mathematics*, Hillsdale, NJ: Erlbaum, Chapter 12, 1987.

[9] Stahl, G., *Group Cognition*, MIT Press, 2006.

[10] Stubbs, M., *Discourse Analysis: The Sociolingistic Analysis of Natural Language*, U. of Chicago Press, 1983.

[11] Tchounikine, P., Rummel, N., McLaren, B. M., Computer Supported Collaborative Learning and Intelligent Tutoring Systems, in Zhou, N. Nkambo, R., Bourdeau, J., Mizoguchi, R. (eds.) *Advances in Intelligent Tutoring Systems*, Springer, Chapter 22, 447-463, 2010.

[12] Zhou, N., Question Co-Construction in VMT Chats, in Stahl, G. (ed.) *Studying Virtual Math Teams*, Springer, 141-159,