

The SASK Mentor Project: Using Software and Socratic Methods to Foster Reflective Thought in a Learning Environment

Baba Kofi Weusijana

Learning Sciences

k-weusi@northwestern.edu

Christopher K. Riesbeck

Computer Science & Learning Sciences

riesbeck@cs.northwestern.edu

Joseph T. Walsh, Jr.

Biomedical Engineering

jwalsh@northwestern.edu

Northwestern University / VaNTH ERC

Abstract

We have developed SASK (Socratic ASK), a domain-independent architecture for implementing Socratic dialogs to foster deeper student reflections on well-defined tasks. In SASK we have built the Dialysis Mentor, a program that uses Socratic questioning to improve student performance and learning in an undergraduate biomedical engineering lab. Small usability tests and a pilot run in the actual lab suggests that Dialysis Mentor and SASK systems in general can improve the value of pre-defined learn-by-doing task experiences. We are now working on improving our SASK Mentors and building authoring tools for them.

SASK Mentor Project: <http://www.cs.northwestern.edu/~riesbeck/sask/>

Introduction

Dr. J. Walsh's Biomedical Engineering Laboratory course at Northwestern University includes a dialysis lab session in which students characterize how well an artificial kidney (a dialyzer) transports water, urea, and salt across its membrane. Students work together in groups of two or three for several hours on this lab. This lab typically uncovers a number of important gaps in student understanding, ranging from simple trouble using the lab devices, to misinterpretation of key measurements, to performing actions contradictory to the basic goal of the experiment. Dr. Walsh handles such problems not by telling the student what to do, but by asking a few penetrating questions, such as "What are you trying to do here?" followed by "What variables are you controlling?" These questions cause the students to quickly discover for themselves gaps in their understanding of the task. Without such attention, students tend to rush through their lab. Many fail to produce any useful results. Even those who do, likely miss many opportunities to reflect on their thinking while they are doing their lab work. Such reflection is known to increase students' ability to transfer their learning to new settings and events (Bransford et al. 2000).

This Socratic technique is facilitated by broad knowledge of the domain and how one's students best learn that particular domain. Often people with such expertise are a scarce resource. Most graduate assistants lack broad domain knowledge as well as experience in teaching it Socratically. Hence, groups in Dr. Walsh's lab would often wait 20 minutes for his guidance before they could make any progress. Sometimes groups had wasted time doing something unfruitful and would have to continue the lab during the next class session. Even worse, sometimes a group would get too much information from a graduate assistant and finish the lab without gaining the understanding Dr. Walsh had hoped they would.

Our project goal was to provide an artificial mentor (We use the term "mentor" instead of "tutor" because tutoring implies a teacher-centric, drill and practice activity. Mentoring implies that we are trying to provide a student centered, apprenticeship, inquiry activity.) capable of providing a service similar to that Dr. Walsh provides, at the time students need it. A Socratic system makes students active learners, leading them to debug their own thinking and knowledge, unlike alternatives such as Frequently Asked Questions (FAQ) lists and other systems that encourage passive learning.

The Project's Development

The Dialysis Mentor

The SASK engine and the Dialysis Mentor (DM) application were developed in summer and fall of 2001, using Java for the engine, XML for the domain content, and QuickTime for Java to play videos. When students start DM, they first see the "Overview" interface (Fig. 1), where they can watch a video of Dr. Walsh's lecture on the dialysis lab, and read a transcript of the lecture.

Students then switch to the "Lab Mentor" (Fig. 2) to interact with the program. This interface includes an annotated diagram of the laboratory set-up for reference (upper left), a dialog transcript (lower left), and, most importantly, a dialog interaction panel, upper right. In future, we plan to include video clips of Dr. Walsh speaking to the students in the role of the Dialysis Mentor.

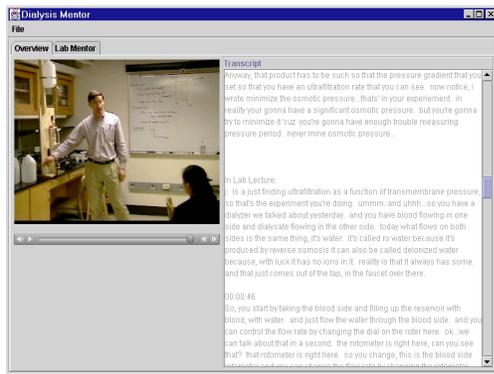


Figure 1: Overview Mode

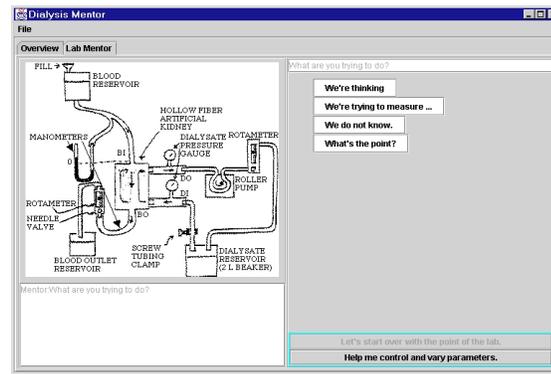


Figure 2: Lab Mentor Screen

The conversation panel was the key interface challenge, specifically the part where students answer questions such as "What are you trying to do?" On one hand, the interface had to avoid overly influencing or constraining what students say. What students say is the key window to their underlying misconceptions. On the other hand, an empty text box where anything can be written is both impossible for current programs to understand and intimidating to students.

Therefore, we use a mixed button and template approach. Simple answers with no variable content, such as "We're still thinking," are buttons. More complex answers with variable content are buttons that bring up structured "fill in the blanks" templates similar to web forms. For example, clicking on "We're trying to measure..." brings up a template of the form "We're trying to measure <empty text field> and control <empty text field>." (Fig. 3). The students can write anything they want in the text fields. This semi-structured approach lets students choose what to say and how to say it, but the program only has to deal with short phrases in well-defined contexts. This interface approach was successfully used in Creanimate, a program that used a question and answer dialog to help children learn and think about zoology (Edelson 1993).

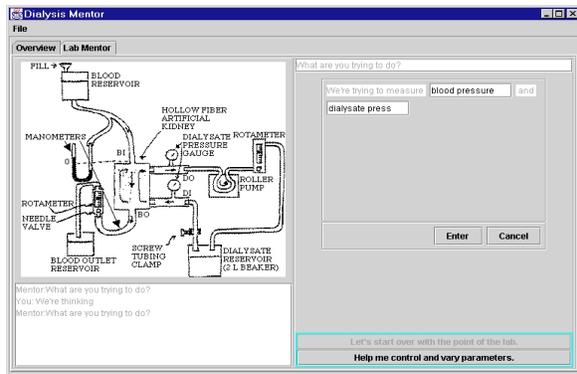


Figure 3: After students choose "We're trying to measure..."

The program's behavior is specified in the Dialog Graph Document, represented in XML, that contains all the Dialysis Mentor's questions, response options, and template patterns for selecting follow-up questions. The current DM graph has 89 utterance nodes and 153 edges (rules that link student responses to DM utterance nodes) in it. Figure 4 shows part of a small part of that graph in outline form. Student responses with underlined elements are template patterns, the other responses are buttons. This fragment Socratically asks students to test their assumption about what controls what, which will lead them to discover a mistake in their thinking.

Mentor's current goal: **Students seem to know the main goal. Lead student to realize that varying the flow does not help the goal**

Mentor: Remember that when you do an experiment you want to vary only one thing at a time. Then you graph Ultrafiltration (U) over Transmembrane Pressure (TMP).

Do you think you will see a relationship between Ultrafiltration (U) and flow rate(s)?

Students: Yes

Mentor: What controls the Ultrafiltration (U), TMP or flow rate(s)?

Students: The flow rate

Mentor: How do you know U is not controlled by the TMP?

Students: I do not know

Mentor: Please set up your experiment so you can tell me what controls U, TMP or flow rate.

Students: TMP directly controls U based on the equation you gave me.

Mentor: That's the theory we presented. You must use the lab to prove that TMP controls U and not flow rate. How do you change only TMP?

Students: ANY OTHER ANSWER directly controls U based on the ANY OTHER ANSWER.

Mentor: Please set up your experiment so you can prove what controls U, TMP or flow rate.

Figure 4: Partial Graph of Dialysis Mentor's Task Dialog Graph Document

Background

Socratic Intelligent Tutoring

A classic early Socratic intelligent tutoring system was WHY (Wenger 1987) created by Dr. Allan Collins. Collins analyzed the dialog between students and Socratic tutors and developed a theory for the implementation of a Socratic-tutoring program, including a set of 24 production rules to improve WHY's pedagogical component (Collins 1977). We used some of these rules as a formal way to understand Socratic techniques. For instance, the partial graph in Figure 4 is an example of Collins's Socratic Rule 15: "Request a test of the hypothesis about a factor".

ASK Systems

ASK systems are a form of multimedia based on the metaphor of having a conversation with an expert, or a group of experts. An ASK system presents a user with a set of initial top-level questions. When the user selects one, the ASK system responds with an answer, either in video or text. In addition, the system displays a set of follow-up questions relevant to the answer given. The user can pursue one of these follow-ups, which will lead to a new answer and a new set of follow-ups, or return to an earlier answer and follow a different path (Cleary 1995). A number of ASK systems in a wide variety of domains were developed by the Institute for the Learning Sciences at Northwestern University, and later commercially by Cognitive Arts Corporation. The success of ASK systems in a number of domains leads us to believe that static dialog graphs are sufficient to provide performance and learning support in well-defined tasks.

Validation

Usability and Pilot Testing

A small usability test was performed with a biomedical engineering undergraduate student. We videotaped the student doing the dialysis lab under Dr. Walsh's tutelage. Her interactions were typical of those in the actual lab, and was used as a seedbed for the initial dialog graph. Several months later, we videotaped her interacting with the first version of the Dialysis Mentor. This test revealed a number of phrases that needed to be added, as well as some new dialog branches. It also showed that it was very easy for certain DM responses to lead the student to treat the system like a multiple choice guessing game. This was particularly true when the DM's immediate follow-up to a student response strongly implied that the response was incorrect or unwise. To avoid this unfortunate phenomenon, some follow-ups were changed to avoid immediate implications of any judgment. This made the dialogs more Socratic.

The Dialysis Mentor was then used in Dr. Walsh's dialysis lab in October 2001. Fifty-two biomedical engineering undergraduates were involved in two sections, a morning section and an afternoon section. Each section had nine groups of two to three students each. The DM logged all interactions for later analysis. Some interactions were video taped. A few weeks after the use of DM in the lab, 47 students completed a survey regarding their experience. Although the test in general is considered a success, preliminary analysis of the data revealed or emphasized issues for future research. In particular, Chart 1 shows the DM's responses categorized by whether the intended follow-ups to student responses were taken. "Missing phrase in task dialog graph document" meant that the DM had a relevant dialog branch internally, but did not recognize that a student response should take that branch. "Wait too long" meant that the DM told students that it would wait for them to do something but waited an improperly long period of time. "No existing branch" meant that the DM's dialog graph didn't include a relevant branch for the student's response. "Appropriate action" meant that the DM selected the dialog branch that we had intended for a given response.

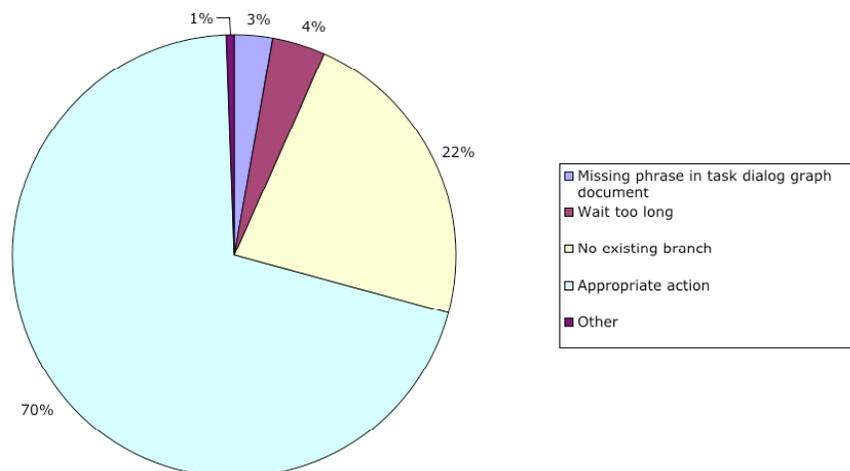


Chart 1: Dialysis Mentor's Categorized Actions with Students in Pilot Test. N= 888 DM Actions Logged.

The following chart from the surveyed opinions of 47 of the 52 students who were in the pilot test shows that the vast majority felt that the DM was at least somewhat helpful to them. Given the rough preliminary nature of the DM, this is quite encouraging.

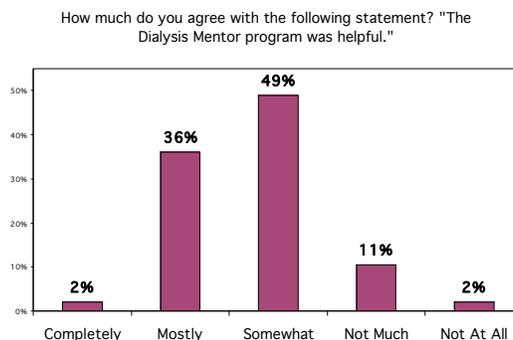


Chart 2: Student Opinions on the Helpfulness of the Dialysis Mentor

The following chart shows the results of the conceptual assessment question in the same survey. Answers to the question “What parameters should one vary when quantifying the hydraulic permeability and what parameters should one hold constant?” was scored by first noting who answered correctly to what parameters should be varied and then who answered correctly to what should be held constant. 9 students (19.15%) were incorrect about what to vary and 12 students (25.5%) were incorrect about what to control. 6 students (12.77%) were found incorrect about both parts of the question and 9 students (19.15%) were partially correct. 32 students (68.09 %) answered the question completely correct.

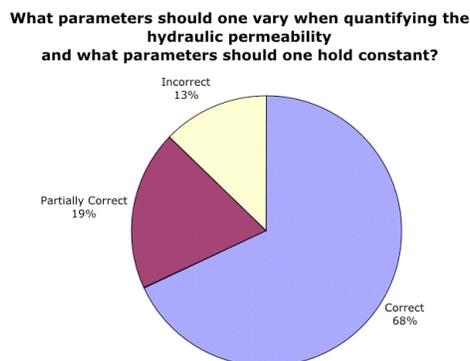


Chart 3: Student Responses to Conceptual Assessment Question

Future Work

Social Interaction and Interpretation Issues

Videotapes of the pilot tests show cases where a student group was at a loss for an answer to a question presented by the DM. When Dr. Walsh asked the students to pretend he had just asked them the same question, the students quickly came up with an answer. When they entered that answer into the DM, it was usually handled properly and the dialog progressed. In other words, students treated questions from the DM differently from the same questions from Dr. Walsh.

There are several possible explanations for this phenomenon. One is the difference between text and speech. Questions that are clear when spoken often become ambiguous in text form. Another factor is the difference in social relationships between a computer program and a faculty member. When a faculty member like Dr. Walsh says “Think about it. I’ll ask you again in 5 minutes,” students stop and think. When a lowly computer program like the DM said the same thing, students often do whatever they can to get around the

delay, including restarting the program. Since getting students to stop and think is the main goal of a SASK system, overcoming this difference in response is a critical research problem.

For these reasons, we are planning to use video clips of Dr. Walsh asking the questions, in conjunction with the text version. The addition of audible speech may improve DM as a pedagogical agent and improve students' retention and problem-solving transfer abilities (Moreno et al. 2001). It may also increase students feeling like they are talking to Dr. Walsh, or someone like him, therefore reducing the differences in their response patterns.

We also need to make sure students do not guess at answers, because this is antithetical to learning. Guessing happens when students believe they can tell if their response choice was correct or not, based on the system's first follow-up. That makes it easy for students to try each response and see what happens. Therefore, we need to minimize two kinds of initial follow-ups: negative "no, because..." follow-ups, and "I don't understand, please rephrase" follow-ups. The latter happened approximately 25% of the time in the DM, either because of missing phrases (3%), or missing branches (22%). Fortunately, avoiding bad follow-ups is mostly a matter of expanding and refining the Dialog Graph Document.

Authoring Tools and Internet Accessibility

Another area of future design research will be the development of the authoring tool for dialog graphs. The Dialysis Mentor's XML files were authored with a text editor. Even with an XML editor, this is a tedious and technically complicated activity. We plan to design an authoring tool so that teachers such as Dr. Walsh can build the dialog graph document themselves. Of particular interest will be the development of "dialog sequence templates," based on Collins' Socratic rules and other useful dialog patterns we have discovered while designing the Dialysis Mentor.

Using Java Servlet technology, we also have a preliminary web interface to the SASK Mentor to ease deployment to many sites.

Conclusions

Overall, our SASK Dialysis Mentor has the potential to be an effective tool for improving students' learning experiences. We look forward to furthering its development and making other SASK Mentors for other learning environments and domains.

References

- Bransford, Brown, Cocking (Eds.) (2000). *How people learn; brain, mind, experience, and school*. Washington, D.C.: National Academy Press
- Cleary, C., & Schank, R. (1995). *Engines for Education*. Hillsdale, NJ: Lawrence Erlbaum.
- Collins, A. (1977). Processes in acquiring knowledge. In Anderson, Spiro, & Montague (Eds.), *Schooling and the acquisition of knowledge*. Hillsdale, NJ: Lawrence Erlbaum.
- Edelson, D. C. (1993). *Learning from stories: Indexing, reminding, and questioning in a case-based teaching system*. Ph.D. dissertation, Northwestern University.
- Moreno, Mayer, Spires, & Lester (2001). The Case for Social Agency in Computer-Based teaching: Do Students Learn More Deeply When They Interact with Animated Pedagogical Agents? *Cognition and Instruction*, 19 (2), 177-213. Hillsdale, NJ: Lawrence Erlbaum.
- Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems*. Los Altos, CA: Morgan Kaufmann.

Acknowledgements

We thank Dr. Brian Reiser, Dr. Daniel Edelson, and Dr. Allan Collins for their very helpful suggestions and groundwork.

This work was supported primarily by the Engineering Research Centers Program of the National Science Foundation under Award Number EEC-9876363.