

Designing a Tutoring Agent for Facilitating Collaborative Learning with Instant Messaging

Sheng-Cheng Hsu¹, Min-Yuh Day^{3,4}, Shih-Hung Wu⁵, Wing-Kwong Wong²,
and Wen-Lian Hsu³

¹ Graduate School of Engineering Science and Technology

² Graduate School of Computer Science and Information Engineering,
National Yunlin University of Science and Technology, Yunlin, Taiwan

³ Institute of Information Science, Academia Sinica, Taiwan

⁴ Department of Information Management, National Taiwan University, Taiwan

⁵ Department of CSIE, Chaoyang University of Technology, Taichung, Taiwan
wongwk@yuntech.edu.tw

Abstract. In this study, we propose a tutoring agent that uses MSN Messenger, a popular synchronous internet media, as the communication platform. Students can invite the agent to a discussion session on MSN Messenger. The agent understands the students' questions in natural language, and provides answers or hints during the group discussion. Unlike a traditional natural language tutoring agent that converses with one student at a time, our agent needs to work with a group of students in MSN chat.

1 Introduction

With the advancement of internet technology, students with distance education can use synchronous internet media to communicate and collaborate with one another. Common synchronous media include chat room, video conference and instant messaging. Among these media, Instant Messaging (IM) is the most popular, e.g., AOL's Instant Messenger, ICQ, MSN Messenger, and Yahoo! Messenger. Some research [1, 2] indicates that IM is a useful tool to carry out distance collaborative learning. Students discuss and exchange ideas with peers simultaneously. During collaborative discussion among a group, a tutor plays the role of information provider when the students have questions. The tutor answers students' questions directly or provides relevant hints. However, it is still impossible for human tutor to stay online for 24 hours a day. Without the presence of a tutor, students often spend too much time searching for answers in order to make progress in a discussion. In this study, we propose a tutoring agent that uses MSN Messenger as the communication platform. It provides a convenient platform for collaborative discussion, recording collaborative process, and sending electronic files. Up to now, this agent is used in an assembly language programming course. Some research has shown the effectiveness of collaborative programming in computer science courses [3]. Therefore, in this study an assembly language programming course is the topic of distance collaborative learning on MSN Message.

2 The Collaborative Programming Process on MSN Message for Assembly Language

In this study, the collaborative learning process for assembly language programming is based on the Learning Together method and problem solving strategies. There are the following seven steps: grouping heterogeneous students, assigning a problem, understanding the problem, making a programming plan, coding a program, testing and debugging the program, and showing the result.

To start collaborative learning, the instructor first divides students with varying ability into groups of four. Then the instructor assigns each group a project, for example, "Checking Password Project: design a program to read and check a 10-bit password from keyboard." After the group-building activity, the students implement the collaborative programming steps on MSN message. When the students within a group have questions during the discussion, they can invite the tutor agent into the discussion on the MSN Messenger. The students can use natural language to consult with the agent that can provide answers or hints. At the step of understanding the problem, the students try to clarify the problem definition by discussing the following questions: (1) What must the program do? (2) What outputs are required and in what form? (3) What inputs are available and in what form? (4) What are the pre-condition and post-condition? If they don't understand the problem description, the students can ask the agent. For example, "Does the 10 bit password contain blank bit?" In the programming plan making step, the students within a group need to break down a task into a number of small subtasks until each subtask is easily doable. If the students cannot design a feasible plan, the agent can provide a general plan to them.

At the step of coding a problem, the students need to code a program in assembly language according to the programming plan. The agent can provide means, format and examples if they have any question about the assembly language instruction, for example, if a student asks the agent, "What is interrupt INT 21H with AH=01?" In the testing and debugging step, the students try to execute the program and find out errors, such as syntax error, run time error, and logic error. Through the file transmission function of MSN Message, each group can combine the program files edited by each group member. Moreover, the agent will provide the explanation of the error message. In the step of showing the result, the students compile the result as a report, which includes programming definition, project plan, requirement specification, acceptance test plan, and user's manual.

3 System Architecture

The collaborative tutoring agent has four components: user interface, comprehension module, tutoring module, and student module. In this study, MSN Messenger is used as the user interface, which includes contact list, collaborative discussion, and management interface. To use the tutoring agent, the student has to add the agent contact ID: ita_yun@hotmail.com to the MSN Messenger contact list. Then the students can invite the tutoring agent into the chat in discussion. The chat of MSN Message is used as the collaborative discussion interface for students, and the tutoring agent provides help during the collaborative learning. The management interface is implemented based on a class library, dotMSN. It provides an interface for managing the tutoring agent such as login information, contact list, conversion records, etc.

The comprehension module is used to understand students' questions in natural language and is implemented with the INFOMAP knowledge engineering tool provided by the Intelligent Agent System Lab, Institute of Information Science, Academia Sinica [4]. INFOMAP has been successfully applied to areas such as question answering and intelligent tutoring [5]. A powerful feature of INFOMAP is its capability to represent and match complicated template structures, such as hierarchical matching, regular expression, semantic template matching, and frame (non-linear relations) matching used to extract important concepts from a natural language text. The tutoring module stores the problems' answers involving the lexicon knowledge retrieved from the textbook, the plan knowledge for designing a program, and the hint knowledge for guiding the students to finish this team-project. The student module is used to record the students' portfolio in the collaborative learning process. The students' portfolio records the discussion process and the questions students ask the agent. On MSN Message, it's easy to store the portfolio in text files.

4 Conclusion

In this study, a tutoring agent for facilitating collaborative learning with instant messaging is proposed. This agent uses MSN Messenger as the communication platform. Students can invite the agent to a discussion on MSN Messenger. The agent understands the students' questions in natural language, and provides answers or hints during the discussion. So far it has been applied to assembly language programming courses. When students get involved in the collaborative programming process, the agent provides help, including understanding problem definition, designing a programming plan, and understanding error messages during debugging.

Acknowledgements

This research is supported by the National Science Council in Taiwan., W-K Wong and S-C Hsu are supported by NSC94-2524-S-224-001 and NSC 94-2520-S-224-001, S-H Wu is supported by NSC94-2218-E-324-003, while M-Y Day and W-L Hsu are supported by NSC94-2752-E001- 001-PAE.

References

1. Nardi, B. A., Whittaker, S., Bradner, E.: Interaction and outeraction: instant messaging in action. The ACM Conference on Computer Supported Cooperative Work (2000) 79-88.
2. Texley S., DeGennaro D. Should We Ban Instant Messaging in School? Learning & Leading with Technology, Vol. 32. (2005) 6-8
3. McDowell, C., Wener, L., Bullock, H., Fernald, J.: The effects of pair- programming on performance in an introductory programming course. The 33rd SIGCSE Technical Symposium on Computer Science Education, 38-42. (2002)
4. Hsu, W. L., Wu, S. H., Chen, Y. S.: Event identification based on the information map – INFOMAP. The 2001 IEEE Systems, Man, and Cybernetics Conference, Tucson, Arizona, USA, (2001) 1661-1672
5. Wong, W. K., Hsu, S. C., Wu, S. H., Lee, C. W., Hsu, W. L.: LIM-G: Learner-initiating instruction model based on cognitive knowledge for geometry word problem comprehension, Computers & Education, Elsevier (In press)