Collaborative Dialogue with a Learning Companion as a Source of Information on Student Reasoning

Eva L. Ragnemalm

Dept. of Computer and Information Science, Linköping University, Sweden.
E-mail: elu@ida.liu.se

Abstract: This report focuses on a problem within the area of Intelligent Tutoring Systems; that of analysing student's reasoning (student diagnosis). A novel approach to collecting information for this analysis, complementary to traditional student modelling techniques, is presented. This technique is based on using a Learning Companion, a computer based agent, as a collaboration partner to the student. In the dialogue between the student and the Learning Companion, information on their problem-solving process is revealed. This information would then be extracted and used for student modelling purposes. Analysis of the proposed solution is commenced in a small experiment and an explorative implementation described here.

Keywords: Learning Companion Systems, Student Modelling, Student Diagnosis, Collaborative Dialogue, Troubleshooting.

1 Introduction

One of the important components in an Intelligent Tutoring System, ITS, is a Student Model. This is a collection of data about the current student which is used at runtime by the other components in the system in a number of different tasks. Such tasks include for instance planning the sequence of instruction, remediating misconceptions, generating feedback and explaining the reason for an error made by the student (see e.g. [23], [18], [12]). The complexity of the student modelling problem has caused researchers to question the necessity of student modelling (see e.g. [16] and for counter-arguments [15]). This question will not be discussed further in this paper, in the following it is assumed that student modelling is worthwhile.

In this paper student modelling is defined as the process of dynamically gathering information about the student and storing that in a student model. Student modelling is a part of Student Diagnosis, which is the process of analysing the student’s actions and converting that to a foundation for educational decisions [20].

Most ITS adhere to some variation of the pedagogic principle of learning-by-doing. That is, most ITS are designed to allow the student to exercise the skill to be learned (e.g. LISP tutor [6], Sherlock [13], Sophie, [1], Scent [11]). It is in this context that student modelling takes place. The student modelling process is usually a covert activity, even though overt and negotiated student modelling have been proposed (e.g. [10], [16]). In covert student modelling student diagnosis is based on the input available as
the student solves the exercises presented by the ITS. The benefit of covert modelling is that the student is not distracted from the problem solving by (to him) irrelevant questions.

The problem of student diagnosis can be subdivided into three phases [20]. These phases are the data acquisition phase, the transformation phase and the evaluation phase. Data acquisition refers to the collection and refinement of the input, including the abstraction of actual input (such as mouse movements or whatever peripherals are used) to concepts in the domain (such as an action or an answer). Transformation refers to the conversion done either by reconstructing the knowledge causing the student action or by applying knowledge of how to solve the exercise and comparing the result to the student’s action. The evaluation refers to the judgement of whether what the student is believed to know is correct or not in relation to the goal of the teaching situation. The student model is constructed either before or after the evaluation phase, thus student modelling covers either the first two or all three phases.

One problem in the data acquisition for covert student modelling is that of gathering sufficient data. If a procedure is taught, procedure oriented evaluation should be used, not goal oriented, but if the procedure is mental it will be inaccessible to the ITS. Techniques for extrapolating the student’s unobservable behaviour from what can be observed include plan recognition and path finding techniques (see e.g. [23]). These techniques have the drawback that they are dependent on the complexity of the exercise the student is doing. When there are many paths or plans that explain observed activities, they quickly become intractable.

There are also techniques that attack the problem of data acquisition by having the student provide additional information to the system, information which is normally not expressed during problem solving. FITS [18] and EPIC [22] require that the student specify what subproblem he wishes to work on. Other systems provide a tool interface through which information is collected while the student uses the tool in the problem solving. An example of this is the program editor in the LISP tutor [6].

This paper proposes another approach to this problem: the use of collaborative dialogue with a Learning Companion, LC. Collaborative dialogue reveals the reasoning of the collaborating partners. The dialogue can thus provide additional information for the data acquisition process. First an example domain is presented. The alternate method is then presented and analysed. A preliminary experiment and explorative implementation are presented. These initial studies indicate that the method will probably suffice but is computationally complex.

2 Example: student modelling of Diagnosis

An example of a situation where the data available is scarce is learning to diagnose faults in a pulp mill by practising with a simulator-based ITS. Diagnosis is here defined as the identification and remediation of faults in a process. The strategy of diagnosis is a cognitive procedure with 10 steps to be executed in specific order [21]. The steps are presented in Table 1 and the observed sequence illustrated in Figure 1.

---

1. The original list [21] contained 8 steps, (1-7, 10), but the author found evidence for another, the ordering of repairs, 8. The application of repair is also shown as an explicit step, 9.