

Beyond the Short Answer Question with Research Methods Tutor

Kalliopi-Irini Malatesta¹, Peter Wiemer-Hastings^{*2}, Judy Robertson¹

¹ Division of Informatics, University of Edinburgh, 2 Buccleuch Place
Edinburgh, EH8 9LW, Scotland
{kalliopm, judyr}@cogsci.ed.ac.uk

² School of Computer Science, Telecommunications, and Information Systems
DePaul University, 243 S. Wabash, Chicago IL 60604
peterwh@cti.depaul.edu

Abstract. Research Methods Tutor is a new intelligent tutoring system created by porting the existing implementation of the AutoTutor system to a new domain, Research Methods in Behavioural Sciences, which allows more interactive dialogues. The procedure of porting allowed for an evaluation of the domain independence of the AutoTutor framework and for the identification of domain related requirements. Specific recommendations for the development of other dialogue-based tutors were derived from our experience.

1 Motivations for a New Tutor

Recent advances in Intelligent Tutoring System technology focus on developing dialogue-based tutors, which act as conversational partners in learning. AutoTutor [5, 7], one of the prevalent systems in this field, claims to simulate naturalistic tutoring sessions in the domain of computer literacy. An innovative characteristic of AutoTutor is the use of a talking head as the primary interface with the user. The system is also claimed to be domain independent and to be capable of supporting deep reasoning in the tutorial dialogue.

One goal of the current project is to test these claims. Another motivation was the fact that the *domain* of AutoTutor, computer literacy, provides limited potential for activating deep reasoning mechanisms. By porting the tutor to a new domain, which requires in-depth qualitative reasoning, we can address issues of domain independence and framework usability in a concrete manner.

The new tutor, based on the AutoTutor framework, is built on the domain of Research Methods in Behavioural Sciences and thus was named Research Methods Tutor (RMT).

* This work was completed in Edinburgh University.

In this paper, we describe the issues that arose during the porting process. In particular, we will focus on the usability and extensibility claims of the AutoTutor system. Based on the results, concrete suggestions are made on feasible modifications of the framework.

1.1 AutoTutor

AutoTutor aims to collaborate with the student as human tutors do: by co-constructing the knowledge taught through a dialogue on a one-to-one basis.

The tutor presents questions and problems from a predefined curriculum script, attempts to comprehend learner contributions that are entered by keyboard, formulates dialogue moves that are sensitive to the learner's contributions (such as short feedback, pumps, prompts, elaborations, corrections, and hints), and delivers the dialogue moves with a talking head [5]. The talking head was intended to provide a more natural modality for the tutor-student dialogue. It also allows the tutor to give graded feedback, supporting the pedagogical and politeness goals of the system.

AutoTutor has seven modules: a curriculum script, language extraction, speech act classification, latent semantic analysis, topic selection, dialogue move generator, and a talking head. We will not describe all of these modules in detail. Nevertheless, sufficient information will be delivered in the relevant sections, in order to comprehensibly describe the development of the new tutor.

1.2 Main Research Goals

Our main goal was to explore more complex types of dialogue by porting the existing framework of AutoTutor to the new domain of Research Methods. Our secondary goal was to evaluate various aspects of the AutoTutor model. The main research questions addressed were:

How portable is the current software implementation? Does the system allow deep reasoning mechanisms to be activated in the tutorial dialogue of the new domain? Are the dialogue management and the knowledge representation adopted in the framework sufficient to cope with the requirements of a teaching domain that is richer in causal relationships? How does the absence of a user model affect the system's performance in the new conditions?

It should be noted that this evaluation is performed on a qualitative level and is only concerned with identifying system weaknesses and putting forward feasible suggestions for improvement. The teaching effectiveness of the tutor is not raised as an issue and thus will not be addressed throughout this article.

2 Research Methods Domain

As pointed out earlier, the depth of AutoTutor's conversations is limited by its subject. Computer Literacy attempts only to familiarise students with the basic concepts of computers, and does not get into any deep issues. Thus, many of AutoTutor's

questions have a short-answer feel. A more complicated domain would set the grounds for testing if indeed the system can support deeper reasoning in the discourse. Instead of merely accessing simple facts, which are associated with the question, the students should have to make inferences, which lead them to discover or construct a conceptual link in their understanding of the topic.

Deep reasoning in RMT, is the process of a multi-turn dialogue between the tutor and the student. It is constituted of several steps. Firstly a principal concept of the domain is introduced through a fleshed out example. Then the role of the concept and its interaction with other secondary concepts is elicited through the dialogue. The tutor guides the discussion so as to help the student realise and articulate correctly the meaning of the concept and its interaction with other secondary concepts. Issues of causality that link different concepts are of particular interest in this process. As opposed to the computer literacy implementation, the RMT domain supplies a set of concepts that can be analysed beyond their definitional attributes, through worked examples.

At the current stage of this attempt only a subdomain of Research methods was chosen as teaching material, that of the fundamental concepts of True Experimental Design in Behavioural Research Methods. A possible future full-scale implementation of the tutor would aim to teach these concepts to first year college students in psychology or cognitive science, through a tutorial dialogue on specific experimental design examples. Prior preliminary knowledge of the domain by the students is assumed.

2.1 Learning from Examples

Examples are regarded as important components of learning and instruction. In the case of one-to-one tutoring it has been reported that most questions asked by tutors were embedded in a particular example [4]. Sweller [8] has suggested that worked examples have cognitive benefits over active problem solving. Active problem solving often leads to dead-ends, or lengthy, error ridden solution paths. Providing students with worked examples reduces the student's cognitive load by eliminating futile problem-solving efforts.

Others claim that examples are most beneficial when they are rich in context and anchored in real-world situations. These anchored examples include challenging material, are motivational in nature, and ultimately facilitate transfer to new problems [6, chapt. 1].

Based on this research it is apparent that grounding tutoring dialogues in examples is particularly important in one-to-one tutoring and thus in the design of Intelligent Tutoring Systems that aim to simulate naturalistic tutorial dialogues.

Motivated by the finding that most examples in naturalistic one-to-one tutoring dialogues originate from textbooks [6] and having already decided on a research methods related domain, a specific topic selection from the Cozby [1] textbook was decided.

The topic selection was influenced by existing studies of human-to-human tutoring in research methods conducted by Person [6]. The Tutoring Research Corpus of the Institute of Intelligent Systems at the University of Memphis was collected from

upper-division college students who were enrolled in a course on research methods in psychology.

After a detailed study of the topics covered in the transcripts, and keeping in mind the remarks made regarding the value of grounding one-to-one instruction to examples, the topic selection for the new domain was derived from the eighth chapter of the Cozby textbook on Experimental Design. Using the transcripts of the related examples, in conjunction with the actual text from the chosen textbook, four example-based topics were selected as teaching material for the new domain.

2.2 Porting Procedure

After choosing the new domain, the porting procedure consisted of three steps, the collection of a sufficient corpus in order to train the language-understanding component, the development of a curriculum script based on the topic selection previously discussed and the creation of a lexicon of the concepts and terms to be introduced in the tutorial dialogue.

LSA and Corpus

Latent semantic analysis (LSA) is a major component of the mechanism that evaluates the quality of student contributions in the tutorial dialogue. In a study by Wiemer-Hastings et al. [9] LSA's evaluations of college students' answers to deep reasoning questions are found to be equivalent to the evaluations provided by intermediate experts of computer literacy, but not as high as more accomplished experts in computer science. LSA is capable of dealing with different classes of student ability (good, vague, erroneous, versus mute students) and in tracking the quality of contributions in the tutorial dialogue.

LSA is a corpus-based mechanism that represents texts as vectors in a high-dimensional space. Two texts can be compared by calculating the cosine between the vectors that represent them. The training of LSA starts with a corpus separated into units, which are called documents or texts. For the AutoTutor corpus, the curriculum script was used, with each item as a separate text for training purposes. The corpus also included a large amount of additional information from textbooks and articles about computer literacy. Each paragraph of this additional information constituted a text. The paragraph is said to be in general, a good level of granularity for LSA analysis because a paragraph tends to hold a well-developed, coherent idea.

In the first stage of processing a student's response, a speech act classifier assigns the student's input into one of five speech act categories: Assertion, WH-question, YES/NO question, Directive, and Short Response. Only the student's Assertions are sent to LSA for evaluation. The other types of speech acts are processed using simpler pattern-matching procedures.

LSA computes the similarity between any two bags of words. In AutoTutor one bag of words is the current Assertion given by a student. The other bag of words is the content of one of the curriculum script items associated with a particular topic, i.e., a model good answer or bad answer [5]. AutoTutor calculates a general goodness and badness rating by comparing the student contribution with the set of good and bad answers in the curriculum script for the current topic. More importantly, it compares

the student response to the particular good answers that cover the aspects of the ideal answer.

In a study of LSA's ability to match the evaluations of human raters, it was concluded that the LSA space of AutoTutor exhibits the performance of an intermediate expert, but not an accomplished expert [9]. This was noted as satisfactory, since AutoTutor aims to simulate a human tutor that does not have specific training in tutoring within that domain.

In the development of the Research Methods Tutor, the same values for training LSA (dimension and threshold) were adopted. The documents originated from seven text books on research methods and from articles and tutorials published on the Internet. As explained earlier, the domain for the new tutor was restricted from General Research Methods in Behavioural Sciences, to the subset of True Experimental Design. Thus only the relevant chapters from each book were scanned. This choice was supported by Wiemer-Hastings et al. [9] finding that more text from the specific tutoring domain is better for LSA. Unfortunately this also made the collection of the corpus a very time-consuming and tedious procedure since only one or two chapters in each textbook were deemed relevant to the desired domain.

Corpus Size

The size of the training corpus for LSA is one important parameter of AutoTutor's language analysis mechanism. The corpus collected for the computer literacy domain was 2.3 MB of documents. A series of tests were performed on the amount of corpus and the balance between specific and general text [9]. As expected, LSA's performance with the entire corpus was best, both in terms of the maximum correlation with the human raters and in terms of the width of the threshold value range in which it performs well. One surprising result was that there was a negligible difference between a corpus consisting of 1/3 of the original items, and one which contained 2/3 of the original corpus. It was observed that there is not a linear relation between the amount of text and the performance of LSA. Another surprising finding was the relatively high performance of the corpus without any of the supplemental items, that is, with the curriculum script items alone.

The fact that there is very little difference in the performance of LSA between the 1/3 and 2/3 of the corpus, is a finding that will be used as a supportive argument for the corpus size collected for the Research Methods Tutor. The size of the corpus that was finally obtained on true experimental design is 750Kb. This renders it close to a third of the AutoTutor corpus. Since the whole procedure was extremely time consuming and the performance of the system was not expected to improve in the case the corpus size was doubled, that size was accepted as optimal for the purpose of this first attempt in implementing RMT.

Development of Curriculum Script

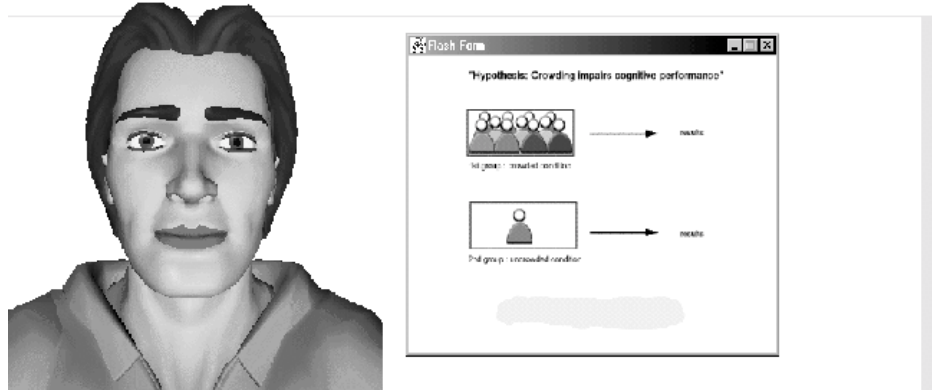
The curriculum script is the module that organises the topics and the content of the tutorial dialogue. In the case of RMT, since the overall goal of this project was to demonstrate the feasibility of this approach instead of creating a full version of the tutor, only four topics in Experimental Design were developed. AutoTutor provides three levels of difficulty (easy, medium, difficult). In RMT the curriculum script

included only the easy level since its short-term goal was to test the overall behaviour of the framework in the new domain.

The topics have a Graphic Display + Question + Answer format. The dialogue is initialised by the presentation of the example of an experimental design which the tutor and student are going to collaboratively work on improving. An image accompanies each subtopic of the topic covered in order to clarify the conceptual relationships that are being discussed. To make this clearer we will take a closer look at the curriculum script developed for one of the examples.

Initially the student is presented with the hypothesis that "Crowding impairs cognitive performance" and a preliminary experimental design on testing it. A multi-turn dialogue will follow, aiming in putting across the fundamental concepts of the topic taught. An image depicting the current experimental design is used to facilitate understanding (Figure 1).

(Hypothesis: Crowding impairs cognitive performance) Which are the experimental variables in this example and what are their characteristics?



Crowding is the independent variable, cognitive performance is the dependent variable.

Fig. 1. RMT

Particular attention was given to constructing dialogue moves that promote deeper reasoning within the limits of the available dialogue moves. As opposed to the fact-based dialogues in the computer literacy domain, the new domain is taught through case-based dialogues. The formulation of the questions was done by adopting the dialogue structure of the tutoring transcripts available, to the form of hints, prompts, elaborations and splices. A major restriction in this approach was the fact that word order is not taken into account from the language understanding component. So, the design of the questions had to be done in such a way that word order was not of significance. This proved quite limiting since deep reasoning is achieved only through the discussion of causal relations were the direction of causality, and thus the word order in the phrase, does matter.

Lexicon

A set of definitions of the concepts introduced in the curriculum script and of other relevant terms that might arise in the tutorial dialogue, was derived from the textbook at hand and from the other sources used for the corpus collected.

3 Lessons Learnt and Problems Identified with AutoTutor

This section describes some problems encountered during the porting of AutoTutor to the Research Methods domain. These problems include: the knowledge representation approach, the ignorance of word order from the language understanding component and the dialogue planning mechanism. Further problems were encountered with the interface and tutoring dialogues during a small-scale evaluation session with AutoTutor. These are discussed in more detail below, as they are important obstacles in the way of developing a full version of RMT (pilot study section).

A primary limitation of the system, which is unavoidably reflected in all its modules, is the knowledge representation approach. The use of LSA shows satisfactory effects in "understanding" student contributions. Nevertheless, the fact that this understanding is local and limited in the sense that no more information can be extracted from LSA apart from a similarity metric to pre-stored information is problematic. While concepts and causal relationships have no means of being represented it is difficult to track the process of the tutorial dialogue in the new domain. The need of distinguishing between user input which was correct by chance and a user contribution that emerged from true understanding of the topic taught cannot be covered in the existing implementation.

During the porting process it became apparent that the ignorance of word order in a domain that supports deeper reasoning is severely restrictive. In the case of computer literacy it is unlikely that a student, with some preliminary knowledge of the subject, could contribute a phrase of incorrect word order i.e. "the computer is in RAM". Thus the fact that the tutor would consider the phrase correct is not as problematic. Unfortunately, in a discussion on experimental design a phrase like "the dependent variable has an effect on the independent variable" is incorrect. Although the system will have the correct phrase pre-stored, i.e. "the independent variable has an effect on the independent variable" in the case that the inverted order phrase above is typed it will fail to recognise the misconception and moreover will positively acknowledge an erroneous input.

The set of curriculum scripts provides a rich set of responses from which AutoTutor can choose, based on the evaluation of the student's contribution and on the dialogue selection rules. In other words the variety of responses is not dynamic since it is a set of canned text utterances. A discussion on experimental design can lead to various dialogue scenarios that are impossible to document in the curriculum script statically and under the restrictions imposed from the AutoTutor framework on the number of hints and prompts each subtopic is allowed to contain. Thus, because the information is pumped from a knowledge base and the utterances and explanations emerge from canned text, its performance is limited by the resources of its library.

AutoTutor's student modelling techniques are deliberately limited. Studies of human-human tutoring have shown that human tutors have relatively shallow understanding of their students' states of mind [6]. Thus, it was assumed that it is sufficient to keep a log of the tutorial dialogue and calculate the learner's ability based on the learner's assertions in all of the previous learner turns in the dialogue.

A major disadvantage of the absence of student modelling is that the tutor has no student profile stored and hence each time the student commences a new tutorial session she is considered as complete novice because there is no record of what he has been taught so far.

During the procedure of porting, unexpected difficulties were encountered due to the authoring of the curriculum script in a cumbersome token format. This is an issue that has already been identified by the Tutoring Research Group and is soon to be dealt with, with a creation of an authoring toolkit.

3.1 Pilot Study

A pilot study was designed in order to collect feedback on the actual characteristics of AutoTutor from the user's perspective. What concerned us was to investigate the claims made regarding the systems performance in terms of:

- quality of interface design
- dialogue management
- student model
- deep reasoning

Eight subjects were recruited in this study, amongst which four were PhD students in various disciplines (psychology, interface design, agent interfaces, cognitive science) and four were MSc students in cognitive science. Their feedback proved valuable both from an expert's and a user's perspective. It should be noted that these are not the normal target users for the AutoTutor system, that is, first and second year undergrads.

The subjects were asked to interact with the tutor for about 15 to 20 minutes, until they successfully covered three topics of the Hardware macro-topic and one topic of the Internet macro-topic. Then they were asked to fill in a questionnaire that inquired their overall opinion about the tutor and specific information regarding the aspects of interest listed above.

The overall impression of the tutor was recorded mostly as "not very good" (one subject found it good, six found it not very good and one found it as bad). This is a finding that justifies our interest in further investigating the tutor's actual performance.

Interface

Although literature on AutoTutor speculates that the agent interface is more effective than a conventional text-to-text application, there is no empirical data to back this intuitive claim. In an evaluation performed on the interface of the system by Link et al. [3], the factors that influence the perception of feedback delivered by an agent are investigated. The evaluation was derived by user ratings on how positive or negative the agent's feedback seems to be to users, in conditions where the speech parameters

and facial expressions of the talking head were manipulated. Their results support the claim that verbal and nonverbal cues are additive. Specifically, participants relied on both linguistic expressions and the mouth curve.

There are several weaknesses in this evaluation approach that are worth mentioning. Firstly, only a small number of features of each of the two modalities was tested. The values of these features were chosen arbitrarily. Moreover the assumption that the agent should be designed to mimic a human tutor in his facial expressions and gestures is not supported by empirical evidence. In an analysis by Dehn [2] on the impact of animated interface agents it is made clear that such an assumption has yet to be validated.

In the same study, a systematic review of empirical evidence on agent interfaces is conducted. The conclusions report that although an agent character is largely perceived to be more entertaining, other dimensions, such as utility, likeability and comfortability are moderated by the kind of animation used and the domain in which the interaction is set. It is clearly stated that present studies do not suffice to enable us to make clear predictions as to what type of animations employed in what type of domain will result in positive attitudes towards the system. Another important point of this study is that all attempts on evaluation so far in this field, concerned limited short-term interactions with the systems. It is unknown if longer exposure to an animated agent might sustain or reduce the positive effect of entertainment ratings. It is interesting to see how these conclusions are verified from the feedback collected in the pilot study.

On the interface related questions, the feedback collected was mostly negative. The prevalent complaint in all users' suggestions was the poor quality of the agent's voice articulation. "Having to listen so hard made me lose parts of the information", reported one of the users. This feature had immediate impact in the focus of attention of the user. Users were distracted from the actual content of the tutorial dialogue, by the extra effort they had to use to understand the tutor's utterances. The log files of the tutoring session reveal that on average ten requests for repetition were made in sessions of approximately fifteen minutes duration. This is a high rate of repetition, considering that native English speakers were preferred as subjects, so that there would be no confounding effects by their level of mastery in English. Users were asked if they would prefer an agent interface over a conventional graphical user interface and the feedback collected shows that it is not obvious to the users that indeed an agent interface is better than a conventional one. This finding is consistent to the arguments put forward earlier on in this section.

The pilot study showed that the users found it difficult to interpret the agent's feedback. Most of them reported that they could not figure out if the agent was pleased with their contribution or if he thought it was incorrect. This increased the levels of their unease in the interaction with the system, since they had no knowledge of the progress of the tutorial. This dissatisfaction can be clearly observed both in the remarks the users put down at designated free comments area of the questionnaire. It was also pointed out that the prosody of the speech was unnatural, making it difficult to distinguish when the tutor was actually posing a question.

A novice user in computer literacy stated "I wasn't sure if answers were sarcastic, does "right" mean "you know nothing"? It made me feel, Oh my god I'm completely

stupid!"). This point is particularly important since we would expect the tutor to have a more positive and motivational impact on novices in the domain taught.

It is interesting to report some observations made of the users interacting with the system. The users took the agents' reactions seriously and were pleased when his feedback was evidently positive. One of the users said "I'd rather do this than read a book". They often made gestures back at him. According to Dehn [2] the extent to which the users perceive the agent as believable has immediate influence on their expectations from him. They often felt frustrated when correct contributions were dismissed with an ambiguous feedback gesture or even a negative one.

Dialogue Management

We aim to investigate the quality of the dialogue in terms of:

- coherence,
- clarity of the tutor's questions,
- interactivity,
- impression that the tutor understood the contributions.

Users rated the coherence of the dialogue as "intermediate". A problem identified was that many questions were too open-ended, resulting to a sense of confusion from the user's side.

Users were puzzled by some of the tutor's questions, especially the prompts, where it was not clear to them what the tutor was expecting.

Moreover the dialogue is structured in such a way that student initiative is very limited.¹ This leads to one-sided dialogues, in the sense that the tutor maintains the control of the tutorial flow and thus the student is not free to demonstrate her knowledge about the topic at hand. The student is restricted to answering questions and has no possibility of posing her own, apart from mere definition inquiries.

Regarding the limited sharing of control over the tutoring process, the users often felt frustrated towards the tutor because they felt there was no actual interaction. They felt limited to the tutor's instruction plan. Additional users' comments indicate that the session was not perceived to be interactive and that more user initiative was expected in order to conduct a truly collaborative dialogue.

Users reacted positively to the tutor's capability for answering to WH-questions. A problem observed in this functionality is that if the student poses a WH-question while the tutor is expecting an answer to an already given question, then the tutor jumps to answering the WH-question forgetting the previous question at hand. This is a weakness that is apparent in the whole of the dialogue management. As the tutor formulates a question, he expects the next student response to answer to that specific question. If the student types in a contribution long before the tutor finishes posing the question the tutor will still believe that the student input was a response to his question, unless it was a WH-question.

¹ As student initiative we mean any student contribution to the dialogue that is not an answer to a question asked by the tutor.

Student Model

It has been stated earlier that AutoTutor's approach to student modelling is deliberately very limited. This choice has serious implications in the flow of the tutorial dialogue. The tutor is often repetitive because he has no idea of the knowledge of the student. In the pilot study it was observed very often that the student knew the correct answer to a question, but LSA did not manage to match the correct answer with its pre-stored data. This caused the users to become frustrated by the tutor's repetitiveness.

The absence of a user model also has serious implications on the usability of the system. The tutor stores no profile of the user. This makes it pointless for someone to attempt to use the system more than once because he will be forced to go through the same topics covered in his initial session with the tutor.

Deep Reasoning

The AutoTutor literature postulates that the tutor supports deep reasoning in the tutorial dialogue. This claim is difficult to investigate since the domain taught and the instruction approach adopted do not provide the ground for deep reasoning dialogue patterns. The domain of computer literacy does not permit reasoning mechanisms to be fully activated since it constrains the dialogue on a descriptive and definition oriented level. Concepts taught are mostly entities such as hardware components and there are no causal relationships between them that could trigger in-depth conversations.

4 Conclusions and Further Research

Our primary goal during this study was to investigate the feasibility of creating a new tutor based on the current AutoTutor system. It has become apparent that the domain of Research Methods increases the requirements from the system and that the current implementation does not suffice. The issues raised regarding the approaches of interface design, knowledge representation and dialogue management lead us to the conclusion that the current AutoTutor framework must be extended in order to support a new tutor such as RMT.

4.1 Feasible suggestions for improvement

The relationship between knowledge representation, language understanding and user modelling is evidently intricate. The current implementation has proved to be rather constraining. The addition of syntactic information in the language understanding component is deemed necessary for overcoming the word order problem and thus supporting deep reasoning in more demanding domains such as Research Methods. This will be a focus of our future research. Moreover, the dialogue planning approach should be reconsidered, aiming to provide functionalities that allow more student initiative and promote better understanding of the tutor's requests. More student initiative would help alleviate the tutor's repetition incidents since the user will be

able to intervene in the tutorial dialogue and skip the delivered information that is already known. This will render the overall dialogue more interactive.

Dialogue planning should also be reconsidered. The current implementation imposes restrictions on the number of hints and prompts that can be associated with each good answer. The development of hint and prompt generators could help surpass the inflexibility of canned text and allow for a dialogue that is more tailored to the students needs.

Regarding the interface design, it is clear that further research is required into the effectiveness of interface agents. A short-term solution for dealing with the agent's poor articulation would be to accompany all utterances with the equivalent text bubbles. The effectiveness of a talking head over a talking head with speech bubbles and a conventional text-to-text interface has to be empirically tested in order to properly support the choices adopted by the system.

Overall the procedure of porting proved to be productive, since it shed light to issues that can only be identified through a hands-on investigation of the AutoTutor software implementation. The conclusions drawn aspire to provide useful guidelines for researchers in the field. Further research on the areas discussed will allow RMT to advance the state of the art in dialogue-based tutoring.

References

1. Cozby, P. C. (2001) *Methods in Behavioural Research*. Mayfield Publishing Company, 7th edition.
2. Dehn, D. M. (2000) The impact of animated interface agents: a review of empirical research. *Human-Computer Studies*, 52, 1-22.
3. Link, K. E. & Kreuz, R. J. & Graesser, A. C. and the Tutoring Research Group (2001) Factors that influence the perception of feedback delivered by a pedagogical agent. *International Journal of Speech Technology*, 4, 145-153.
4. Graesser, A. C. (1993) Questioning mechanisms during tutoring, conversation, and human-computer interaction. Memphis State University, Memphis, TN. ERIC Document Reproduction Service No. TM 020 505.
5. Graesser, A. C. & Wiemer-Hastings, K. & Wiemer-Hastings, P. & Kreuz, R. & the Tutoring Research Group (1999). AutoTutor: A simulation of a human tutor. *Journal of Cognitive Systems Research*, 1, 35-51.
6. Person, N. (1994) An analysis of the examples that tutors generate during naturalistic one-to-one tutoring sessions. *PhD thesis*. University of Memphis.
7. Person, N.K., Graesser, A.C., Kreuz, R.J., Pomeroy, V., & TRG (2001). Simulating human tutor dialog moves in AutoTutor. *International Journal of Artificial Intelligence in Education*, 12, 23-39.
8. Sweller, J. (1988) Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12, 257-285.
9. Wiemer-Hastings, P. & Wiemer-Hastings, K. & Graesser, A. C. (1999) Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis. *In AI in Education*, 545-542. Les Mans, France.