From: AAAI Technical Report FS-00-01. Compilation copyright © 2000, AAAI (www.aaai.org). All rights reserved.

A computational model of Affective Educational Dialogues

Angel de Vicente* and **Helen Pain**
Artificial Intelligence; Division of Informatics
Edinburgh University; Edinburgh (United Kingdom)
{angelv, helen}@dai.ed.ac.uk

Abstract

Issues of affect have been largely neglected in computational accounts of dialogues so far. Given the importance of affect in education, we present in this paper on-going work on a computational model of affective educational dialogues.

Introduction

Emotional and affective issues are recently enjoying an increased amount of attention in the field of Artificial Intelligence, as can be seen by the number of workshops and conferences devoted to them (e.g.: (Emotion in HCI 1999; Emotion-based Agent Architectures 1999; Affect in Interactions 1999)). These issues are also of great importance in the area of Educational Dialogues, both for better understanding human educational interactions and for creating Tutoring Systems that communicate with students in a more natural and effective way.

Language is a powerful communicator of affect, and it can be used in tutoring systems to empathise with students and to detect their emotional state during the interaction. Related work exists (e.g. (Allport 1992; de Rosis & Grasso 1999; Horvitz & Paek 1999; Person *et al.* 1999)), but a detailed computational model of affective dialogue is missing, a gap that we attempt to fill with the on-going work reported in this paper. The main aim of the model presented in this paper is to generate educational dialogues, focusing primarily on their affective characteristics.

The model (discussed in more detail below) bases its decisions on a number of sources: rules drawn from several theories of motivation and education; features of the past interaction of the student and a model of the student himself. In its current version it has been incorporated into a mock instructional system called AFDI (AFfective DIaloguer), described below. We also give and comment on a short example of a dialogue generated with the current model.

*Supported by grant PG 94 30660835 from the "Programa de Becas de Formación de Profesorado y Personal Investigador en el Extranjero", Ministerio de Educación y Cultura, SPAIN.

A computational model of affective educational dialogues

One of the difficulties of creating a computational model of affective educational dialogues is that it cannot be created in isolation from an instructional system. Although the system does not have to be a 'real' tutoring system, the model needs information about an instructional interaction in order to be useful. In this sense, the development of an educational dialogue model amounts to the development of an instructional model in which different parts of it are 'glued' together by the use of language. This can be seen in figure 1, which represents a simplified overall view of the AFfective DIaloguer (AFDI) computational model.

The rectangular nodes represent steps of text generation; the oval nodes represent decision-making points and the skewed rectangular nodes represent student interaction. Thus, the model assumes that an instructional interaction will start with a general introduction, followed by a cycle in which different instructional units are studied, followed by some concluding remarks. During the main instructional cycle, the model allows five different types of interventions: two of them initiated by the system (to provide help or to attempt to solve motivational problems); and the other three initiated by the student (to finish the lesson, to give up or to ask for help).

The oval nodes in the model represent, as mentioned, decision points. It is in these nodes that affective issues are taken into account in order to generate the dialogue. These decisions are based on the history of interaction with the system and the following student characteristics[1]:

- Student's trait characteristics: fantasy, challenge, control, independence. The model uses a measurement of how much the student appreciates the given characteristic during instruction. For example, does the student like challenging situations?
- Student's motivational state characteristics: satisfaction, relevance, confidence, effort, sensory interest, cognitive interest. These characteristics are transient, and the model assumes two different sources for them:

[1]These are based on previous work on diagnosis of student's motivation (de Vicente & Pain 1999)).

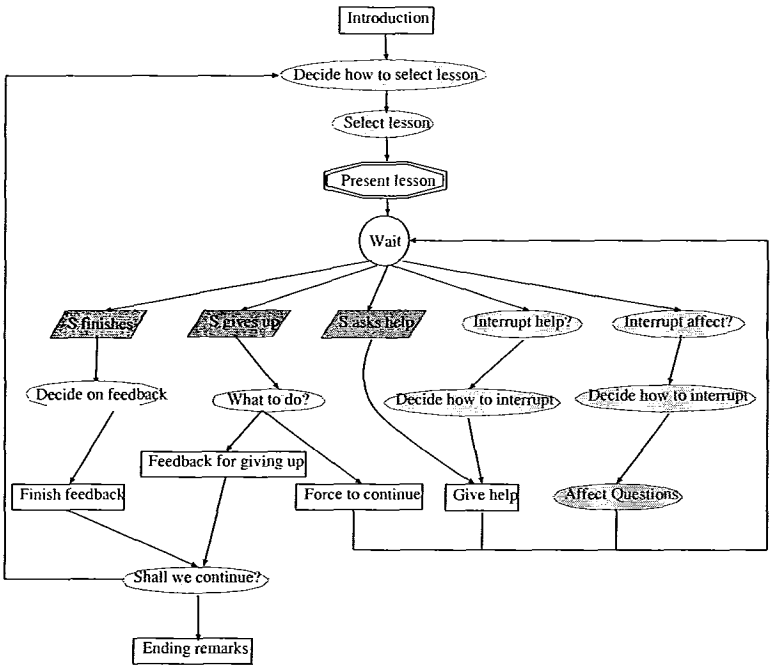

Figure 1: AFfective DIaloguer (AFDI) Computational model (simplified)

- Self-report, in which the student provides his own estimation of each of them.
- Inferred from the behaviour of the student.

Language used

In order to keep complexity low, our generation process is oriented towards generating affective educational dialogues, but producing the actual text from canned text options. A more sophisticated method of text generation would be desirable, but our main concern in developing this model is to present a plausible affective dialogue generation approach in the field of instructional systems[2]. Nevertheless, the design of the model (and of the AFDI system) is modular, and it would allow for an inclusion of a more sophisticated Natural Language Generation engine.

The tutor dialogue moves are selected from a set of possible sentences which are classified according to their content and their affective characteristics. The basic possible themes to generate tutor moves are given by the rectangular boxes in figure 1 (e.g. "Introduction", "Give help", etc.). For each tutor move, a number of possible student replies are attached. The selection of which particular tutor move to generate, or which possible replies to present to the student, is determined by the particular characteristics of the student and the history of the interaction.

When the model is in a text generation step (e.g. 'Introduction'), the program issues a request to generate a tutor move of certain characteristics (i.e. topic: help; subtopic: lesson1; flow: provide; politeness: high,; etc.)[3]. Another procedure is responsible for returning an appropriate move of those characteristics. In the current AFDI version this is done by looking at a database of "typical" tutor moves.

Tutor moves Given a request for a particular type of tutor move, our model will generate a move that matches the given requirements among all the possible tutor moves. Some examples of possible tutor moves are given in table 1(a). The three columns of this table represent respectively: the move characteristics; the actual text, and the student reply options.

In order to add more flexibility to the language used in the model, the text can be pre-determined or selected during run-time. Thus, in the first tutor move in table 1(a), the text is made of a variable part ([sel GEN_INTRO]), which selects randomly from a list of possibilities for generic introductions. These (and examples of other sub-sentence variables) are given in table 1(b). This allows for a more varied dialogue output, while keeping the moves database simple.

Student replies The tutor moves also have information about which student replies are appropriate for each of them. For example, in the first move in table 1(a), this is given by

[2]For an approach in which Natural Language Generation techniques are used for affective text generation see (de Rosis, Grasso, & Berry 1999; de Rosis & Grasso 1999) and (Porayska-Pomsta, Mellish, & Pain to appear 2000) for more linguistically motivated research on generating teachers' language in educational dialogues.

[3]The politeness characteristic is not used in the current version of AFDI.

	Characteristics	Text	Student reply
1	introduction:intro	[sel GEN_INTRO]	reply_intro
2	feedback:pos_inc_conf	[sel POS_FEED] [sel INC_CONF]	reply_pos_inc_conf
3	help:talk_about_help	It seems that perhaps you would benefit from some help. Would you like some?	reply_y_n
4	ending:due_system	Well, I think is perhaps time to leave it for today.	reply_bye

(a) Sample tutor moves

GEN_INTRO	Welcome to Moods, your affective tutor! Welcome to Moods Good morning, and welcome to Moods
INC_CONF	You see, you can do it! And you thought you could not do it?
POS_FEED	Well done! Congratualations, you are doing great.

(b) Sample sub-sentences

reply_intro	Hi Thanks
reply_y_n	Yes No
reply_pos_inc_conf	Thanks, it is true. It is easier than I thought! Well, it was probably just luck!

(c) Sample student replies

Table 1: AFDI language

the variable 'reply_intro'. This variable represents the possible student replies, which can be generated at run-time or pre-determined. In the current AFDI version, only the selection of lessons by the student is generated at run-time (based on the previous history of interaction), while the other possible replies are predetermined. Examples of possible student replies are given in table 1(c).

The affective knowledge

The main aspect of AFDI is its ability to make affect related decisions in order to generate the dialogue. These decisions are guided by the affective knowledge, which is implemented by two types of knowledge-based rules: *dialogue planning rules* (those shown in figure 1 as elliptic nodes) and *modelling rules*.

The *dialogue planning rules* represent knowledge about the generation of Educational Dialogues moves, given the history of interaction and the student's characteristics. The *modelling rules* represent knowledge about which information concerning the student's affective state can be inferred from his interaction with the system. These two types of rules are discussed in the following sections.

Dialogue planning rules The dialogue planning rules implemented in the current version of AFDI are very simple. For brevity, we give here just one example in detail and then present a summarised version of all the rules in the current AFDI version.

The rule *stud_choose* shown in table 2 represents a dialogue planning rule concerned with *control*, a major affective factor during an instructional interaction. In AFDI the selection of the next lesson to study can be done entirely by the student, entirely by the system, or by the collaboration of both. This is reflected in the rule *'stud-choose'*. There we

control::trait	**relevance**::state	Output
very high	any value	stu_chooses_all
average–high	below very high	stu_chooses_type
below average	below average	stu_chooses_type
average–high	very high	system_chooses
below average	average or higher	system_chooses

Table 2: Rule stud_choose

can see that the decision at this point is influenced by two variables: the student's *control* trait characteristic, and the *relevance* state characteristic. Thus, we see that if the student's *control* is very high[4], then we always let the student choose the next lesson. If his *control* is *average* or *high*, then the decision depends on whether he thinks that the instruction is of relevance to him. If it is *very relevant*, then we assume the system is doing a good selection of lessons, and we should let the system choose the next lesson to perform. If the student thinks that the materials are not really relevant to his learning, then we let him choose the type of exercise to do next.

A summarised version of all the rules in the current AFDI version are given in table 3.

Modelling rules These rules are similar in syntax to the dialogue planning rules but they are formalisations of the inferences about the student affective state that can be drawn from the student's replies. As explained earlier, every time that the system generates some text, the student is offered

[4]The values for all the variables in the student model can take five different values: -10, -5, 0, 5 and 10, corresponding respectively to: very low, low, average, high, very high.

Rule name	Depends on	Description
stud_choose	control, relevance	Who chooses next lesson (student, partially student, system)
decide_feedback	performance, effort, confidence	What type of feedback to give
student_chooses_to_continue?	how_many_lessons, control	Should the student decide whether to continue?
system_continue_instruction?	how_many_lessons, decaying_effort	Should we continue the instruction?
allow_give_up?	effort, challenge	Should we allow the student to give up?
interrupt_affect_R1	satisfaction	Is general satisfaction so low that we should interrupt the student?
interrupt_affect_R2	sensory_interest, cognitive_interest	Is the interest so low that we should interrupt the student?
interrupt_affect_R3	relevance	Is the material being taught irrelevant to the student's needs?
interrupt_affect_R4	Inferred user_state, Reported user_state	Is the reported model very different from the inferred one?
student_decides_interrupt_affect?	independence	Should the student be asked about whether to solve a motivational problem?
interrupt_help_R1	confidence	Is the confidence so low that we should interrupt to offer help?
student_decides_interrupt_help?	independence	Should the student be asked whether he wants the help or not?

Table 3: Dialogue planning rules

some choices to form his reply. According to this reply, we can sometimes infer certain affective information that can be used to update his affective model. These rules encapsulate some of this knowledge. As for the dialogue planning rules, we present first one of them in detail.

STUDENT REPLY	CHANGES
easy	effort -5
hard	effort 5

Table 4: Modelling rule pos_plain

The rule *pos_plain* in table 4 is a formalisation of the inferred changes in the student model after the system has provided *pos_plain* feedback (positive feedback). The system can provide positive feedback with one of four possible moves (for instance "That was very good!"). To this, the student can reply with two different replies:

1. Thanks, it was easy.
2. Thanks, but it was hard!

Thus, based on the student's reply, this rule helps to infer the amount of effort that the student has put into the task. A "Thanks, it was easy" reply means to our model that the student did not put too much effort into the task[5], while a reply of "Thanks, but it was hard" would mean that the student made a big effort[6].

AFDI has other similar rules to update the student model after replies to various types of feedback, help, etc., but a description of all of them would take too much space in here. We illustrate some of them below with an example of a dialogue generated with AFDI.

Implementation of the model

In order to experiment with the model, we have implemented it in a simple application, whose interface can be seen in figure 2. The working of the system is very simple. In the top part of the interface the representations of the student's trait and state (both inferred and reported) characteristics are given. By having them available all the time we can modify them easily to see their effect on the generation of the dialogue. Similarly, we can also see how different student replies affect the student model. In the top part of the interface, a brief description of the simulated lesson is also given.

The bottom part contains two frames: 1) *Dialogue Log*, where all the dialogue generated and the rules applied to generate it are logged; 2) *Dialogue move*, where the current tutor move and its possible student replies are displayed. In debugging mode (as seen in figure 2) there is a third frame where the history of interaction database is shown as the simulated instruction takes place. In this database we store information about the lessons studied, the outcome for each of them, etc.

In order to interact with the system, we simulate the behaviour of a student[7] and decide on the possible outcome to each lesson (i.e. succeed, give_up, etc.). The time that the student would spend in studying the lesson and a measurement of performance is also simulated through the se-

[5]We decrease the student model's value of effort by 5.

[6]We increase the value of effort by 5.

[7]Using an approach which we would rather call 'Dorothy' than 'Wizard of Oz' ...

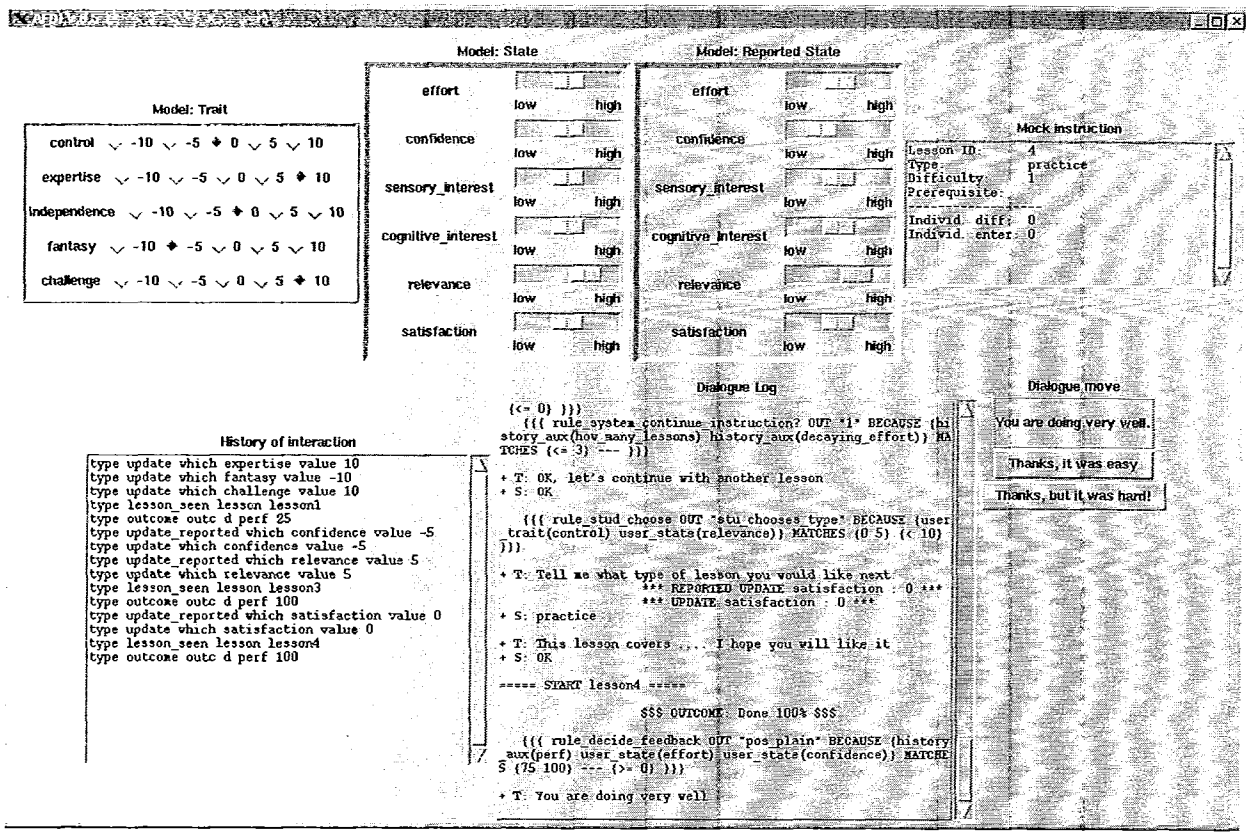

Figure 2: Main interface of AFDI.

lection of the appropriate menu options. As the interaction takes place, the student model will vary depending on the student's performance, the replies to the tutor moves, etc. These changes will also be reflected in the selection of the lessons and future dialogue moves. This enables us to see the model functioning in an interactive mode. An example dialogue generated in this way is given and commented upon in the following section.

AFDI Generated Example Dialogue

Below we present an abridged version of an actual dialogue generated with AFDI. For conciseness, we have omitted certain parts of the dialogue and present only the sections that indicate more clearly the main aspects of AFDI. The text is divided in two columns: to the left, the actual dialogue is presented; to the right, we present a summary of the interaction with the system plus a description of the *dialogue planning* and *modelling* rules that were used to shape the dialogue.

After a brief introduction and the update of the trait characteristics by the (simulated) student, the first point of interest is that marked in the dialogue as [1]. This illustrates how the important motivational factor of *control* (or, more importantly, *feeling of control*) is dealt with in AFDI. The rule used here was described in detail above. At this point AFDI decides that the student should be given some, but not total, control over the next lesson to study. This is motivated by the fact that the student's desire for control is not very high[8], and that the materials were not very relevant to him[9]. Thus, the student is asked to choose the difficulty of the next lesson to study.

The option of exercising *control* over the interaction is also seen in the dialogue at point [11]. There we see that despite reaching the established maximum number of lessons for this session (4 in this example), the student is offered the choice to decide whether he would like to continue with the instruction, as his *control* is high (greater than 0).

Control is also the concern in point [5], but in this case AFDI limits the *control* exercised by the student. Because the student's desire for control is not very high, he has not put much effort into the task, and he likes challenging situations (challenge greater than 0), the system tries to encour-

[8] `user_trait(control) {0 5}` meaning that this variable had one of these values at this point.

[9] At the beginning of the interaction, the variables are initially set to *average* (or 0).

age him to try a bit harder.

Another issue which is of crucial motivational importance is feedback. Whether the feedback is positive or negative, whether it tries to encourage the student, etc. can affect how the student feels about the instruction. An example of this is given in points [7] and [9]. In [9] the feedback is positive, since the student's performance was very high (75% or higher; 100% in this example). But in [7] we see that, despite similar performance, student's self-confidence is low (less than 0), and thus the feedback is characterised as 'pos-inc-conf' (positive feedback, plus increase confidence). This is realized in our example as "Well done! And you thought you could not do it?"

As seen above, the purpose of the feedback 'pos-inc-conf' is to increase student's self-confidence, but it is equally important to understand how the student reacts to it. Given the example tutor move in previous paragraph, the student can select from two possible replies: 1) "Thanks, it is true. It is easier than I thought!"; 2) "Well, it was probably just luck!". These are meant to provide information about whether or not the intended tactic had the desired effect. In this case, the student has selected the first reply, which would indicate (as given by the rule 'pos-inc-conf' in [8]) that his confidence has increased slightly (by a step of 5 in this example). If the other option had been chosen (i.e. "Well, it was probably just luck!"), the model would infer that the confidence was actually decreasing. Other points in the dialogue where other *modelling* rules are fired are [3], [4], [6] and [10].

In [2] we illustrate the 'Interrupt' options of our model. In (de Vicente & Pain 1998) we have argued that in order to tackle motivational problems, an affective tutor would benefit from an ability to interrupt the instruction if the conditions required it. This ability to interrupt the instructional interaction, but not following a predetermined path, has two advantages:

1. It can create the illusion of a more flexible tutor who does not follow a strict instructional plan.
2. It can detect motivational problems as soon as they occur and it can take remedial actions.

AFDI checks regularly[10] whether it should interrupt. We see in [2] one of these situations. The student has updated the reported variables of *sensory interest* and *cognitive interest* to *very low*. This indicates that the student does not find the instruction interesting at all, and therefore the rule *interrupt_affect_R2* is fired, which starts a sub-dialogue to find out the type of instruction that the student would prefer[11]. Thanks to this capability of interrupting, the system can offer help and can try to solve motivational problems as soon as their need is detected. At the same time, it would encourage the use of the self-report facilities, since the student would perceive that his interaction with the interface have immediate effect (de Vicente & Pain 1999).

[10] Every 10 seconds in the current version.

[11] This is preceded by a decision to ask the student whether he would like this interruption, since his 'control' is high.

Conclusions

In this paper we have presented a computational model of educational affective dialogues, which is being developed in the context of developing 'Affective Tutors'. This model does not attempt to present a psychologically plausible model of affective communication, but rather to give a first approximation of the neglected issue of affect in Educational Dialogues.

Currently the model generates simple, although plausible, educational dialogues. Nevertheless, evaluation of them is necessary. We are planning an evaluation of the model in which excerpts of dialogues will be presented to tutors, in order to evaluate the appropriateness of the tactics used in our model. We hope that this evaluation will help to refine the model and to create a more appropriate model of affective Educational Dialogues.

Acknowledgements

We would like to thank the anonymous reviewers and our colleagues in the AI and Education Group at the University of Edinburgh for useful comments on previous versions of this paper.

Raw dialogue text (abridged)

Tutor: Welcome to Moods, your affective tutor!
Student: Hi
Tutor: Please, let me know about you by filling in the questionnaire on personal traits.

```
UPDATE control : 5
UPDATE expertise : -5
[...]
```

Student: OK, it is done
Tutor: Shall we start? The material to be learnt covers basic Japanese numbers.
Student: OK

```
rule_stud_choose
user_trait(control) {0 5}
user_state(relevance) {< 10}
---> stu_chooses_type
1
```

Tutor: Tell me what type of lesson you would like next.
Student: difficult

```
CALCULATING AVAILABLE LESSONS
```

Tutor: Let's now study .... Good luck
Student: OK

```
REPORTED UPDATE sensory_interest : -10
UPDATE sensory_interest : -10
REPORTED UPDATE cognitive_interest : -10
UPDATE cognitive_interest : -10
```

```
rule_interrupt_affect_R2
user_state(sensory_interest) {< 0}
user_state(cognitive_interest) {< 0}
---> 1
2
```

```
rule_student_decides_interrupt_affect?
user_trait(control) {> 0}
---> 1
```

Tutor: I'm a bit worried about your motivation. Would you like to talk about this?
Student: Yes
Tutor: Apparently you are not enjoying this class very much. Is this correct?
Student: Yes
Tutor: I see ... Maybe I could do better if you answer the following questions.
Student: OK
Tutor: You would like more ...
Student: theory?

```
affect_presentation
STUDENT_REPLY MATCHES theory
---> fantasy -5
3
```

Tutor: Thanks, I will try to follow your advice. Now, would you like to continue with this lesson?
Student: No

```
[...]
```

Tutor: OK, let's continue with another lesson
Student: OK
[...]
Tutor: In this lesson we will study .... Good luck
Student: OK

```
OUTCOME: Help (slow) 25%
```

Tutor: Look at .... Perhaps that helps you
Student: I already knew that, but thanks

```
help
STUDENT_REPLY MATCHES already
---> relevance -5
4
```

OUTCOME: Give up 50%

```
rule_allow_give_up?
user_trait(control) {< 10}
user_state(effort) {< 10}
user_trait(challenge) {> 0}
---> 0
5
```

Tutor: Come on, you cannot give up now. You have to try a bit harder
Student: OK, but I'm not sure I can do it

```
force_continue
STUDENT_REPLY MATCHES sure
---> confidence -5
6
```

OUTCOME: Done (slow) 75%

```
rule_decide_feedback
history_aux(perf) {75 100}
user_state(effort) ---
user_state(confidence) {< 0}
---> pos_inc_conf
7
```

Tutor: Well done! And you thought you could not do it?
Student: Thanks, it is true. It is easier than I thought!

```
pos_inc_conf
STUDENT_REPLY MATCHES easier
---> confidence 5
8
```

Tutor: OK, let's continue with another lesson
Student: OK
[...]
Tutor: Let's now study .... I hope you will like it
Student: OK

OUTCOME: Done 100%

```
rule_decide_feedback
history_aux(perf) {75 100}
user_state(effort) ---
user_state(confidence) {>= 0}
---> pos_plain
9
```

Tutor: That was very good!
Student: Thanks, it was easy

```
pos_plain
STUDENT_REPLY MATCHES easy
---> effort -5
10
```

[...]

```
rule_student_chooses_to_continue?
history_aux(how_many_lessons) {> 3}
user_trait(control) {> 0}
---> 1
11
```

Tutor: Perhaps you would like to finish now?
Student: Yes
Tutor: I hope you learnt something useful. See you next time.
Student: Bye

References

Affect in Interactions 1999. International Workshop on Affect in Interactions. Towards a New Generation of Interfaces (Annual Conference of the EC I3 Programme). Available at the World Wide Web: http://gaiva.inesc.pt/i3ws/i3workshop.html (October 18, 1999).

Allport, G. 1992. Towards a computational theory of dialogues. CSRP Report 92-3, University of Birmingham. School of Computer Science. Cognitive Science Research Centre.

de Rosis, F. and Grasso, F. 1999. Affective text generation. In Affect in Interactions (1999). Available at the World Wide Web: http://gaiva.inesc.pt/i3ws/accepted-papers.html (November 25, 1999).

de Rosis, F., Grasso, F. and Berry, D. C. 1999. Refining instructional text generation after evaluation. *Artificial Intelligence in Medicine* 17(1):1–36.

de Vicente, A. and Pain, H. 1998. Motivation diagnosis in intelligent tutoring systems. In Goettl, B. P., Halff, H. M., Redfield, C. L. and Shute, V. J., eds., *Proceedings of the Fourth International Conference on Intelligent Tutoring Systems*, volume 1452 of *Lecture Notes in Computer Science*, 86–95. Berlin. Heidelberg: Springer.

de Vicente, A. and Pain, H. 1999. Motivation self-report in ITS. In Lajoie, S. P. and Vivet, M., eds., *Proceedings of the Ninth World Conference on Artificial Intelligence in Education*, 648–650. Amsterdam: IOS Press.

Emotion-based Agent Architectures 1999. Workshop on emotion-based agent architectures. Available at the World Wide Web: http://www.ai.mit.edu/people/jvelas/ebaa.html (October 18, 1999).

Emotion in HCI 1999. A British HCI Group one-day meeting in conjunction with University College London on "Affective Computing: The Role of Emotion in HCI". Available at the World Wide Web: http://www-users.york.ac.uk/~am1/affective.html (April 8, 1999).

Horvitz, E. and Paek, T. 1999. A computational architecture for conversation. In Kay, J., ed., *Proceedings of the Seventh International Conference on User Modeling*, 201–210. Wien New York: Springer. Available at the World Wide Web: http://www.cs.usask.ca/UM99/ (May 24, 2000).

Person, N., Klettke, B., Link, K. and Kreuz, R. 1999. The integration of affective responses into AutoTutor. Available at the World Wide Web: http://gaiva.inesc.pt/i3ws/accepted-papers.html (June 5, 2000).

Porayska-Pomsta, K., Mellish, C. and Pain, H. to appear, 2000. Aspects of speech act categorisation: Towards generating teachers' language. *International Journal of Artificial Intelligence in Education. Special Issue on Analysing Educational Dialogue Interaction* 11.