Generating Explanations using an Automated Planner and Modelling Reasoning Processes, Skills and Knowledge

Alan Wheeldon and Jim Reye

Queensland University of Technology
Brisbane, Australia
a.wheeldon@qut.edu.au and j.reye@qut.edu.au

Abstract

In this paper, we describe a user focused approach to generating goal oriented explanations that motivate the user to achieve their task. The planner is combined with the knowledge base in a way that distinguishes between the user's and tutor's reasoning processes, skills and knowledge and that reflects the way humans' reason in the context of explanatory dialogue. Personalised information is provided to the user to match their needs for motivation to complete a task, and information that enables them to complete a task.

Introduction

A requirement of a successful ITS (Intelligent Tutoring System) is that it can provide explanations the user finds appropriate, motivational and goal directed. This focus requires that an ITS takes into consideration different levels of users' understanding, any past dialogue with the user, and that it is able to justify its own actions in a way that is acceptable to the user. In order to generate effective explanations a system needs to maintain a model of the user. A user model is necessary for an ITS to be able to tailor its output to the individual needs of the user (du Boulay 2000).

ITSs require input from a wide range of subject areas and as such are not easy to design or create. Although ITSs are becoming more common and proving to be increasingly effective, they are difficult and expensive to build. Reye provides a good summary of why it is hard to develop an ITS that is in regular use, citing such problems as "basic AI research issues, the knowledge acquisition problem, user-interface design issues and the inherently multidisciplinary nature of ITS research" (Reye, 1997, p.12).

In this paper, we describe a user focused approach to generating goal oriented explanations that motivate the user to achieve their task. We describe how this system can generate explanations that are tailored to the user's needs and background knowledge; we discuss the components of the system and how they work together. First, some relatively recent systems with similar research goals are discussed.

Background

In an ITS, it is generally assumed that dialogue is co-operative. The system must have a record of the user's goals, plans, skills, reasoning and beliefs in order for the system to tailor its output appropriately. In this way the system ensures the best chance of the user accepting and, where appropriate, acting on the text generated. Although it is often difficult for a system to obtain an accurate view of the users goals, plans, skills, reasoning and beliefs this problem is obviated when considering systems designed to "perform a limited task for a limited audience" (Sormo & Cassens, 2004).

These requirements are supported by current research work on users' motivation and challenge, interest and use (Mitchell, Sheard & Markham, 2000). They are also supported by the aims of research reported in some relatively recent systems e.g. (VanLehn, et al, 2005). In addition, related research in the areas of *RST* (Rhetorical Structure Theory), for example (Hovy, 1993, Mann & Thompson, 1987), and Discourse Theory (Grosz & Sidner, 1986) as part of Linguistics and, Classical Planning Theory (Nguyen & Kambhampati, 2001) is also applicable.

RST (Mann & Thompson, 1987) is a means for describing the structure of rhetoric, or effective language as based on the observation that the structure evident in a text gives rise to rhetorical relations. It is a descriptive theory. To use RST to guide text construction requires transforming rhetorical relation schema and definitions into plan actions. These plan actions can achieve goals via RST effects and pre-conditions are formed from RST constraints. Such plan actions can be used in an ITS to organise text for coherence and structure.

Some relatively recent systems have adopted a hybrid approach to planning explanations, e.g. (VanLehn, et al, 2005) based on a NOAH-style approach (Sacerdoti, 1975). In these systems, the same type of planning actions are often used to model the user's knowledge and reasoning as for modelling the real world actions the tutor may take. Taking this approach, the system considers only the actions of the tutor with no consideration of how the user may reason about and react to these actions of the tutor. For example, the system considers a tutor action designed to motivate the user has the direct result that the user is motivated. This may be useful as it enables a consistent approach (Moore, 1995), however the user's knowledge and reasoning is not distinguished from their actions.

A User Focused Approach

An underlying theme of the paper is that explanations require a lot of preparation of ideas. This preparation, commonly referred to as 'thinking' in humans, requires selecting the content or ideas, arranging the ideas into a sequence, and then finally conveying them as words in phrases and sentences.

In our approach, the system models the user's reasoning processes on being supplied with information by the tutor. For example, the system models the way the user reasons about a recommendation to take an action; are they motivated to want to take the action and do they have sufficient knowledge to take it[1]. We believe this is a more realistic model of how users learn and in particular how they respond to an ITS.

Our knowledge base records the skills of the user (i.e. actions that the user knows about) and rules of the user that reflect the reasoning processes of the user, as well as facts (i.e. relations in the domain that the user knows) plans and goals. As these components of the knowledge base are identified separately they can be reasoned with independently and form an effective means of developing a realistic model of the user's reasoning processes as well as a realistic model of the user's skills.

Two-way street

The reasoning process of humans is often a two-way street. Given a recommendation to take an action, we are likely to reason towards the outcome, or effect, of following the recommendation, i.e. the effect of the action. Having reasoned about the outcome of the recommended action we are likely to reason about all the conditions that must hold in order to realise the outcome, i.e. we reason about the conditions that must be satisfied in order to achieve the goal. This approach is reflected by the planner as it reasons about the reasoning of the user.

The system proceeds by constructing a plan of the actions the user needs to take to complete a task, i.e. what the tutor would like the user to do. In constructing this plan the system uses the tutor model which is able to reference all aspects of the domain model. This ability is referred to as the *perspective of the tutor*. It then tailors this plan to the individual requirements of the user by considering whether they will be motivated to follow the plan and whether they are able to achieve the plan. In doing this, the system makes reference to its models of the user's knowledge, reasoning processes and skills. This ability is referred to as the *perspective of the user*. One of the benefits of this approach is the system is able to personalise the initial plan of what it wants the user to do in a way that considers different means of motivating the user.

Figure1 shows how the tutoring module guides the interaction of the components of the knowledge base. Dashed-line-arrows indicate references between the models. The dotted lines, connecting the paths of the dashed-line-arrows between the plan and the tutor and user models, indicate the ability of the tutoring module to veil the references from the planner to the domain and user models. The solid-line-arrow from the plan to the user model via the tutoring module indicates the user model being updated.

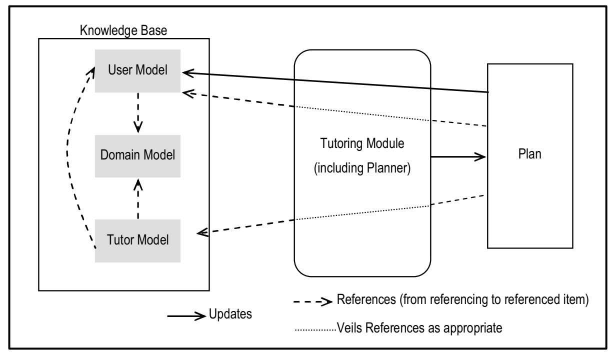

Figure1: Interaction of Knowledge Base Components.

Developing a Plan

Rather than the system trying to create a single plan in one go that solves all the problems it is better to break it up into smaller problems and work on each of those. One of the key ideas of this paper is that the system starts by working out what the user should do if the user was as skilled as the system. It starts by taking the perspective of the tutor to develop a basic plan. The system then has to deal with the fact that the user doesn't necessarily have all the knowledge needed to do what the tutor wants them to do, and further more may not be motivated, and so then we need subsequent plans to address those aspects.

We describe how the system develops an explanation by way of an example in the domain of teaching Microsoft Office Access. The system receives a goal to get the user to change the value of a Property of their Form[2].

Planning What the Tutor Wants the User To Do. The first plan the system creates is called the *Base Plan*, this represents what the tutor would like the user to do, assuming they have all the pre-requisite skills and factual knowledge. The tutoring module adopts the perspective of the tutor. For the action that directly satisfies the goal condition to be applicable, the Form should be open in

[1] We are concerned with generating explanations for users learning in the area of computer programming. We assume they have the skills required, e.g. selecting or inserting text etc, but they may not have the reasoning and factual knowledge required to undertake the actions recommended.

[2] For readers who are unfamiliar with Microsoft Access, and Forms, see for example Adamski & Finnegan 2006.

Design mode and the Properties Tool selected. The action of selecting this tool requires that the appropriate tool bar is visible. The planner selects domain rules modelling this condition as an indirect consequence of opening the Form in Design mode. The actions and rules are inserted into the plan – the plan is complete, as represented in Figure2 (actions shown in boxes, rules are shaded).

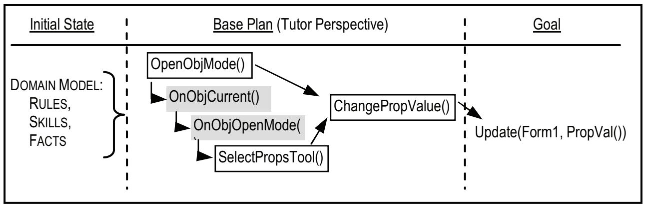

Figure2: Base Plan – Complete.

Planning What the User Needs to Achieve the Goal. Next, the tutoring module considers whether the user is able to achieve the Base Plan. It creates a space to hold a plan that enables the user to achieve the goal - the *Enablement Plan*.

The tutoring module transfers the Base Plan to the Enablement Plan and adopts the perspective of the user - references to the domain model are veiled to the planner and where possible are overlaid with references to the user model. In this example, the user has all the reasoning and knowledge required and the Enablement Plan is complete.

Planning to Motivate the User to Adopt the Goal. When we reason about a recommendation, we tend to distinguish between the recognition of a goal and the adoption of that goal, i.e. by distinguishing between a *potential goal* and a(n) (adopted) goal. In other words, we uncouple the two ideas, thus leaving a gap between the potential goal and the goal. The system recognises the problem of this gap and creates a plan that fills this gap between the user's recognition of a potential goal, on being recommended to take an action, and adopting it as a goal. This plan is referred to as the *Motivated Plan*; it is created from the perspective of the user.

The user has not yet been recommended to take an action so the tutoring module sets-up an assumed initial state for the Motivated Plan, containing facts that will be made true in another stage of the planning. It starts by creating the fact, in this assumed initial state, that the user has been recommended to take the action. It considers the action, from the Enablement Plan, that would directly enable the user to realise the goal and uses the predicate *Recommended*() to create the fact the user has been recommended to take this action.

Next, the tutoring module transfers the goal condition from the Enablement Plan to the Motivated Plan goal condition that the user has adopted the goal – using the predicate *Goal*(). The planner seeks to complete the plan and selects rules that model the user's reasoning. The facts in the user model and this assumed initial state are not sufficient to satisfy the open preconditions in these rules and the Motivated Plan is incomplete as represented in Figure3 (dashed arrowed line represents open precondition).

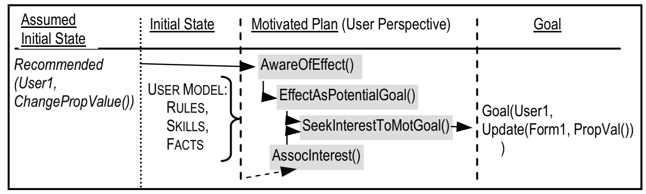

Figure3: Motivated Plan - Incomplete.

To satisfy the open preconditions in these rules the user must be aware of the facts associating the potential goal with their existing interests. Being interested in some aspect of a potential goal will lead the user to recognise some benefit from it and adopt it as a goal. In the example, the user is interested in the purpose of the value of the Property required on their Form. So to complete the Motivated Plan the tutoring module uses the predicate *Aware*() to create the facts that the user is aware of these facts in the assumed initial state of the Motivated Plan.

Delivering the Plan to the User. A plan is prepared that represents the set of actions the tutor takes to recommend the action for the user to take and to make them aware of facts to motivate them to adopt the goal. It includes actions to inform the user of what they need to do and make them aware of the factual knowledge to enable them to achieve the goal. This plan is referred to as the *Delivery Plan*. It is constructed from the perspective of the tutor.

The tutor wants to present the explanation to the user in a way that they can easily absorb it. Information to motivate the user can be best absorbed by them when they have established a context for that information - a potential goal. In a similar way, information to enable the user can be best absorbed by them when they have adopted a goal. The tutor should be able to reason about recommending an action that represents a potential goal to the user independently from reasoning about informing them of what they need to do to achieve the goal.

Explanations should be presented so that the user requires minimum effort in following the flow of the information. Explanations often involve a change of topic, as new topics are introduced or lapsed topics are taken up, and the user is expecting, and actively engaged in, establishing connections between topics, "interrelating information that comes from different constituents such as clauses or sentences" (Just & Carpenter, 1992). The load on the user's short term memory should be minimised in establishing those connections.

The tutoring module transfers the actions from the Enablement Plan to the goal condition in the Delivery Plan that these actions are recommended to the user. Similarly, it transfers the facts it created in the assumed initial state of the Motivated Plan to the goal condition of the Delivery Plan. The planner satisfies the goal condition of the Delivery Plan with the models of the tutor's skills of delivering information to the user (represented in Figure4).

Executing the Delivery Plan provides the user with a plan that recommends them to take an action, motivates them to take the action, and sufficient knowledge enabling them to achieve the plan.

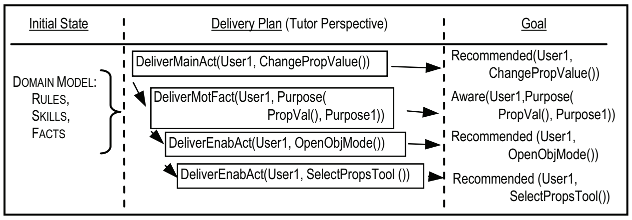

Figure4: Delivery Plan – Complete.

We adopt a principled and rigorous approach to modelling skills and reasoning, using an ADL (Action Description Language) style representation for actions (Pednault, 1989) and rules. Figure5 models the skill of the tutor to deliver an action that enables the user to achieve a goal. The user should have an established goal; the action being delivered enables them to achieve that goal (see (1)), and its effects are not a goal of the user (see (2)). Also, the action concerns a topic that is current (see (3)).

DeliverEnabAct (?user : *Agent*, ?e_act : *PlanAction*)

Precons:	∃ ?g_fact	Goal(?user, ?g_fact)	(1)
	∧	?e_act ∈ {set of actions in Enablement Plan}	
∧	¬ ∃ ?e_fact	(Effect(?e_act, ?e_fact) ∧ Goal(?user, ?e_fact))	(2)
∧	∃ ?e_fact2	(Effect(?e_act, ?e_fact2)	(3)
	∧	∃ ?obj : *Object* (ObjectIn(?obj, ?e_fact2)	
	∧	CurrentTopic(?obj)))	
Effects:		Recommended(?user, ?e_act)	
∧ ∀ ?obj1 : *Object*		(¬ CurrentTopic(?obj1)) WHEN (CurrentTopic(?obj1))	
∧ ∀ ?obj2 : *Object*		(CurrentTopic(?obj2)) WHEN (ObjectIn(?obj2, ?e_fact2))	

Figure5: The Skill of the Tutor to Deliver an Action.

Summary

In this paper, we have described a user focused approach to generating goal oriented explanations that motivate the user to achieve their task. We described how the planner is combined with the knowledge base in a way that distinguishes between the user's and tutor's reasoning processes, skills and knowledge and (we believe) that reflects the way humans' reason in the context of explanatory dialogue.

The classical planning approach ensures the plan is complete and consistent (Russell & Norvig, 2003) so the user can have confidence in the construction process for generating explanations.

The components of the knowledge base are decoupled thus enabling them to be reasoned with independently. Firstly, distinguishing between models for knowledge, skills and reasoning provides a basis from which realistic and explicit models of the user can be developed. Second, the different perspectives of the user and the tutor are formally identified.

Tackling the difficult problem of the preparation of ideas for an explanation by breaking it up into smaller problems enables us to focus on the important parts of each aspect of that preparation - selecting the ideas, arranging them into a sequence, and finally conveying them as words in phrases and sentences. This is achieved through the different aspects of the two-way street approach – combining the use of the Base Plan, Enablement Plan, Motivated Plan and Delivery Plan spaces with the use of perspectives. The importance of this is that it provides a way to explicitly model the reasoning process of humans described above. Also, an important part of the Delivery Plan, and its supporting plans, is incorporating the effects an explanation may have on the user, as explored in the paper. This decoupling of the components of the knowledge base combined with the breaking of the preparation process into smaller parts provides a solid foundation, we believe, for generating explanations. In particular, by formalising the separate components of the knowledge base and parts of the preparation process opens opportunities for examining how explanation can be applied to illuminate system processes to increase user acceptance.

Having made substantial progress in addressing the theoretical side of this work, we are keen to put it into practice.

References

du Boulay, B, 2000, Can we learn from ITSs?. In: *Intelligent Tutoring Systems*, (pp.9-17).

Grosz, B.J. and Sidner, C.L., 1986, Attention, Intentions, and the Structure of Discourse. *Computational Linguistics*, 12(3):175-204.

Hovy, E.H., 1993, Automated Discourse Generation Using Discourse Structure Relations. In *Artificial Intelligence*, (63), (pp.341-385).

Just, M.A. & Carpenter, P.A., 1992, A Capacity Theory of Comprehension: Individual Differences in Working Memory. In: *Psychological Review*, (99):1, (pp.47).

Mann, W.C. and Thompson, S.A., 1987, Rhetorical Structure Theory: A Theory of Text Organization. Technical Paper, ISI Reprint, USC.

Mitchell, M., Sheard, J. & Markham, S., 2000, Student Motivation and Positive Impressions of Computing Subjects. In: *ACE2000.* pp.189-194.

Moore, J.D., 1995, Participating In Explanatory Dialogues. MIT – Massachusetts Institute of Technology.

Nguyen, X.L. and Kambhampati, S., 2001, Reviving Partial Order Planning. In: *IJCAI-01* pp.459-464.

Pednault, E.P., 1989, Exploring the Middle Ground Between STRIPS and the situation calculus. In: *1st Int'l Conf. on Principles of Knowledge Representation and Reasoning (KR'89)*, 324-332, Morgan Kaufmann.

Reye, J.D., 1997, Dynamic Planning for Flexible Tutoring (PhD Thesis: Griffith University).

Russell, S.J. and Norvig, P., 2003, *Artificial Intelligence A Modern Approach* (2nd Ed) Prentice Hall.

Sacerdoti, E, 1975, The Nonlinear Nature of Plans, IJCAI-75, 206-214.

Sormo & Cassens, 2004, Explanation Goals in Case-Based Reasoning, In ECCBR-04, W'shop on Explanation in CBR.

VanLehn, K.A., et.al., 2005, The Andes physics tutoring system: Five years of evaluations. In: *Artificial Intelligence In Education* (AI-ED 05), (pp.678-685).