Issues in Designing Learning by Teaching Systems

DAVID NICHOLS
Computing Department
Lancaster University
Lancaster LA1 4YR, United Kingdom
dmn@comp.lancs.ac.uk

Abstract: Learning by teaching systems are a relatively recent approach to designing Intelligent Learning Environments that place learners in the role of tutors. These systems are based on the practice of peer tutoring where students take on defined roles of tutor and tutee. An architecture for learning by teaching systems is described that does not require the domain model of an Intelligent Tutoring System. However a mutual communication language is needed and is defined by a conceptual syntax that delimits the domain content of the dialogue. An example learning by teaching system is described for the domain of qualitative economics. The construction and testing of this system inform a discussion of the major design issues involved: the nature of the learnt model, the form of the conceptual syntax, the control of the interaction and the possible introduction of domain knowledge.

1. Introduction

The history of AI in Education has been dominated by Intelligent Tutoring Systems (ITSs) – embodying the computer-as-tutor paradigm. Suggested alternatives to this approach have included the computer-as-collaborator (Blandford, 1994; Dillenbourg & Self, 1992) and the computer as a Learning Companion (Chan & Baskin, 1990). This paper addresses a further alternative: learning by teaching systems. A learning by teaching system is an Intelligent Learning Environment (ILE) in which the computer takes the role of a student that is taught by a human learner (Palthepu, Greer & McCalla, 1991). Such an approach emphasises the learner's beliefs about the domain at the expense of the 'correct' world view embodied in the domain model of an ITS.

Learning by teaching systems, or Intelligent Student Systems (ISSs) (Nichols, 1994), have so far received little attention in the literature and there is no accepted system architecture – as exists for ITSs (Polson & Richardson, 1988). The research reported in this paper attempts to highlight and clarify the major design issues involved in the construction of learning by teaching systems. These topics are discussed in the context of the development and testing of an example ISS in the domain of qualitative economics.

2. Background

The idea of a learning by teaching system is based on the widespread educational practice of peer tutoring; where a more-able student (the *tutor*) is paired with a less-able partner (the *tutee*). The *tutor* takes on the role of a teacher and provides knowledge and support when the *tutee* encounters difficulties. The tutor-tutee relationship in peer tutoring represents one extreme along a spectrum of collaborative activities, at the other end 'naive learners work together, coaching each other whenever one student possesses knowledge the other(s) lack' (Katz & Lesgold, 1993). In addition are the informal reports of teachers who find that teaching students is effective in improving their own understanding of the subject (Berliner, 1989).

Peer tutoring studies report cognitive benefits for both tutors and tutees (Goodlad & Hirst, 1989). Tutors on peer tutoring schemes report positively on reinforcing subject knowledge, practice in communication of ideas and confronting other perceptions of the domain. Tutors also develop insight into the teaching/learning process and report improved co-operation with their own teachers.

Tutees on peer tutoring projects do not always behave in a manner conducive to the learning of the tutors. When tutors report what they 'liked least' about their experiences their most frequent answers include 'inattentiveness and indiscipline among pupils' and 'not being fully used by teachers' (Goodlad, 1985) These drawbacks to human-human peer tutoring are exactly the type of problems that a computerised alternative could be expected to address.

ISSs are intended to replace the tutee in a peer tutoring interaction. This means that, in contrast to most of the work on ILEs, they are taking on the role of the less-knowledgeable agent. Learning by teaching systems do not attempt to introduce new knowledge but instead force learners to make their implicit knowledge and reasoning explicit; they are therefore complements to ITSs rather than substitutes.

The only known previously reported experiments involving a learning by teaching system are those of (Michie, Paterson & Hayes-Michie, 1989). Learners taught a system based on the inductive machine learning algorithm ID3 (Quinlan, 1986) in the domain of simple linear equations. Two versions of the system were tested – in one the learning component was disabled. The subjects that used the learning version recorded the larger pre-test to post-test gains. A control group and a group using a commercial maths-education package performed less well than the learning-by-teaching subjects. Conjectured motivational gains (often reported in peer tutoring studies), as measured by questionnaire responses, were not observed.

(Palthepu, Greer & McCalla, 1991) describe an architecture for a learning by teaching system in a declarative domain represented by a standard semantic network. The system has several typical ILE components (Student Model, Pedagogical Mediation, Dialogue Control) but is distinguished by an initially empty Domain Knowledge Base. This is the *tabula rasa* assumption: that the system does not need any initial domain knowledge in order to drive the interaction, merely knowledge about how to construct a dialogue.

> *The knowledge in the system stems from the heuristics that direct the flow of dialogue and the selection of what to say when certain types of changes in domain knowledge occur. All of this means that the system ... is independent of any particular such domain.* (Palthepu, Greer & McCalla, 1991)

This potential domain independence is one of the attractive features of learning by teaching systems. The (Michie, Paterson & Hayes-Michie, 1989) system and the proposed (Palthepu, Greer & McCalla, 1991) system are in different domains but share common features which can be subsumed under a common architecture.

3. Intelligent Student Systems

The two systems described above both learnt about a domain from user input – however neither was a true *tabula rasa*. Although the (Michie, Paterson & Hayes-Michie, 1989) system did not know how to solve an equation it did have knowledge of all the possible actions that could be used. The user supplied the strategic knowledge by entering example equations at various stages in the solution process. Thus the initial domain knowledge of the system was of a different form to that which it acquired from the users. The system constrained the learner's input to equations and then combined them with knowledge of domain actions to synthesise new solution rules. Similarly the (Palthepu, Greer & McCalla, 1991) system constrains the user to a dialogue about the properties and relationships of nodes in the semantic network. However, here the same constraints operate for both the system and the learner – the internal representation restricts the type of knowledge the learner can teach the system.

These systems did not have knowledge of the contents of a model but of the type of elements that were permitted in the model. This *conceptual syntax* is similar to the syntax of a programming language – defining valid statements and programs. Users of the (Michie, Paterson & Hayes-Michie, 1989) system found that the interaction was restricted to equations and that they could not directly express beliefs about operations, coefficients, terms etc. Both of the systems allowed some general dialogue moves (agreement, disagreement, ask-question etc.) but constrained the domain-related interaction to their conceptual syntax. The (Palthepu, Greer & McCalla, 1991) system uses the same representational scheme throughout but the (Michie, Paterson & Hayes-Michie, 1989) system shows that there can be a meaningful distinction between the *user conceptual syntax* (the example equations) and the *system conceptual syntax* (the algebraic actions, e.g. cancel terms, divide by common factor).

An essential component of an ISS is a representation of the knowledge the system has acquired from the user: a Learnt Model, or Domain Knowledge Base (Palthepu, Greer & McCalla, 1991). As the interaction progresses the Learnt Model is maintained by functions based on the system conceptual syntax. The Learnt Model is initially empty in both systems and it is in this sense only that a learning by teaching system can be said to be a *tabula rasa*.

The overall behaviour of an ISS is guided by a dialogue strategy that takes the user input (mediated by the user conceptual syntax) and makes decisions on the updating of the Learnt Model and the appropriate action to take next. The tutees in peer tutoring schemes show significant learning gains (Goodlad & Hirst, 1989) and it is reasonable to assume that tutee learning is an essential part of such interactions.

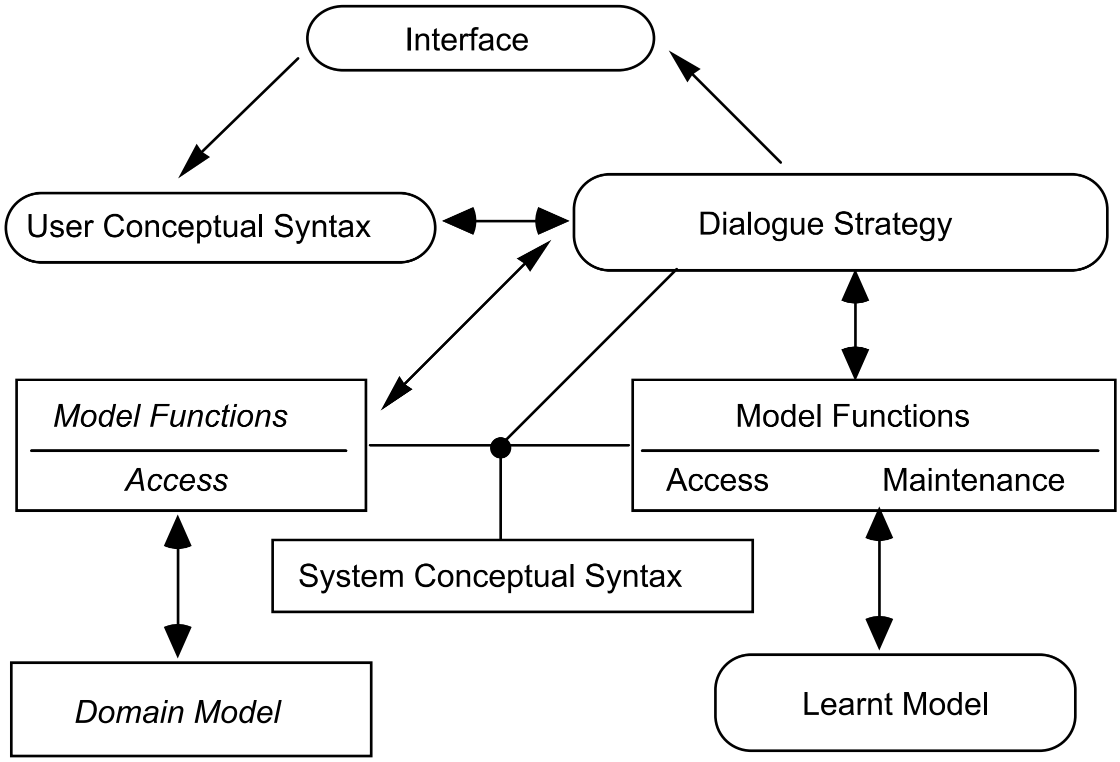

Figure 1: Architecture of an Intelligent Student System Rounded boxes are dynamic components.

The dialogue strategy in the (Michie, Paterson & Hayes-Michie, 1989) system is a machine learning algorithm with the user controlling the interaction whereas the proposed (Palthepu, Greer & McCalla, 1991) system envisages a mixed-initiative dialogue. Several heuristics (based on the conceptual syntax) are suggested to guide the dialogue including: *find contradictions*, *check completeness* and *confirm existing knowledge*. The design of the dialogue strategy of an ISS is an answer to the unusual question 'what is the best way for a system to learn such that its tutor gains the maximum benefit?' A particularly interesting issue is whether the ISS should learn in a human-like manner or whether machine-oriented alternatives are equally effective. Possible sources of dialogue strategies are automated knowledge acquisition systems (Marcus, 1988) and machine learning algorithms that include queries (e.g. (Gasarch & Smith, 1992)).

Domain knowledge, outside of the conceptual syntax, is not a necessary component of an ISS. However, it may be the case that a dialogue strategy with access to a full domain model is more effective than one that relies solely on the Learnt Model. Figure 1 shows the relationships of the components of an ISS – including the optional domain model. The diagram shows that the choice of the system conceptual syntax is crucial as it co-defines the dialogue strategy and the functions that access and maintain the Learnt Model.

4. DENISE: an example ISS

An example ISS, DENISE (Development Environment for an Intelligent Student in Economics), has been designed and built. The conceptual syntax for DENISE is a causal qualitative model of economics and this constrains both the learner's input at the interface and the contents of the (initially empty) Learnt Model. In other words, the user conceptual syntax and the system conceptual syntax are identical. Thus dialogue and system reasoning are both built around statements such as 'an increase in the money supply causes an increase in inflation.'

The dialogue strategy of DENISE largely follows the suggestions of (Palthepu, Greer & McCalla, 1991) in seeking to expand and complete the knowledge in the learnt model without any other domain knowledge. Templates encode the dialogue heuristics which are instantiated by items placed into the Learnt Model by the learner. Typical questions produced by DENISE are 'what else is positively affected by interest rates?' and 'is there anything else that affects unemployment?' In addition there is also a default non-committal response which simply asks the learner to contribute something new.

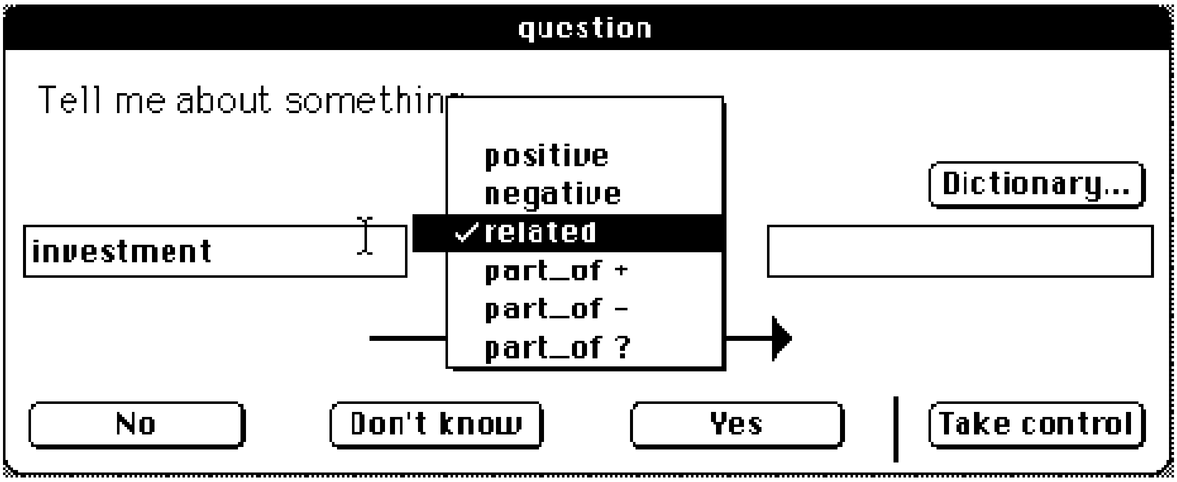

Figure 2: The DENISE interface showing the user conceptual syntax restricting possible input

DENISE also maintains distinct hierarchical viewpoints (Self, 1992) by grouping together beliefs in the Learnt Model; this enables different 'schools of thought' to be represented, e.g. Keynesian, Monetarist etc.. Figure 2 shows a typical window from the DENISE interface. In this case the system's question is a default response that gives the user temporary control over the direction of the dialogue. The conceptual syntax has been 'pushed out' to the interface to constrain the user's input.

The menu shows the six possible causal relationships between two concepts – the dependent variable on the right and the independent variable on the left. The relations in the menu are in two groups: the top three are examples of sequential causality (Berndsen & Daniels, 1989) and the part_of relations are examples of contemporaneous causality (Hicks, 1979) – where the relationship exists in the same time period. The buttons at the bottom of the window allow various dialogue moves to be made; including the ability for the user to take control of the interaction and force the system to become passive. The 'Dictionary…' button simply allows quick recall of previously used concepts.

5. User Trials

Subjects, with varying degrees of economics experience, were simply asked to teach DENISE about macro-economics. The subjects found interactions with DENISE taxing, often pausing for tens of seconds before continuing. Several commented that on the unusual nature of the interaction, expressing discomfort at having to feed a 'knowledge-hungry' agent. One subject even commented: 'it put me off a career in teaching for life.'

The longest pauses occurred when faced with the default non-committal response which effectively transferred control of the interaction back to the learner. Interaction rates were considerably higher for questions which referred to previously referenced concepts. Post-session interviews confirmed a marked preference for the system to lead the dialogue. Although DENISE allows learners to take control of the interaction and to represent several viewpoints none of the subjects utilised these facilities preferring the system to lead the dialogue inside a single viewpoint.

The resulting Learnt Models showed considerable variation in the terms used (the relationships between terms being constrained by the conceptual syntax); typically 65% of terms in a subject's final Learnt Model were not used by other subjects. Although common terms such as 'unemployment' and 'inflation' occurred frequently other terms, such as 'religion' and 'civil war', also appeared. These terms, which are not normally associated with economics, occurred when subjects 'drifted' across the domain – creating long chains of causal links rather than smaller webs of concepts. This 'domain drift' was observed in all of the subjects' interactions.

6. Design Issues

The initial experiments with the DENISE system tend to support the basic position that a learning by teaching system is capable of extracting individualised models from learners and forcing them to examine their own beliefs about the domain. Further discussion of the results is discussed under the four major design issues which arose from the construction and testing of DENISE.

6.1 Learnt Model

The Learnt Model in DENISE was empty at the beginning of each subject's session. This means that there are no initial reference points for the dialogue strategy and the system has no curriculum to follow. The learner can 'drift' around the domain as they retained overall control of the dialogue. The initially empty Learnt Model contrasts sharply with peer tutoring studies as real tutees have considerable knowledge about the domain. A possible alternative is to 'prime' the Learnt Model with some initial beliefs to more precisely locate the dialogue in a particular domain.

Subjects found it difficult to remember which concepts and relationships they had previously used and frequently referred to the dictionary mechanism. This appears to be because the only external reflection of the system learning came in the content of DENISE's questions. This was particularly noticeable when the system produced a default non-committal response. DENISE also contained a query mechanism to allow users to question the Learnt Model although this was rarely used. In the (Michie, Paterson & Hayes-Michie, 1989) system users could test the effectiveness of the currently induced rule against equations thus providing an indication of the state of the Learnt Model. In a declarative domain it may be that some sort of assessment or visualisation of the system's learning is needed to maintain focus in a mixed-initiative dialogue. A graphical representation such as presented in the SemNet system (Fisher, 1992) may be appropriate.

The notion of 'drifting' out of a domain is a value judgement; if it is a realistic representation of the subject's conception of the domain in question then it is impossible for the subject to be 'out' of their own domain. When a system stets out to explore a *subject's* viewpoint of a domain it seems unreasonable to object if it is different to someone else's viewpoint.

6.2 Control of the Interaction

A human tutor has both domain and pedagogical knowledge with which to control the direction of a dialogue with a student. When a learner is placed in the role of a tutor there is considerably less control knowledge and the dialogue can easily drift across the domain. In order to prevent this drift the ISS must take some responsibility for the direction of the interaction.

Task-based interaction, such as most peer tutoring and the (Michie, Paterson & Hayes-Michie, 1989) system's equation-solving, provides an implicit control of the interaction as there are clear goals and sub-goals to be attained. This task-based control allowed the system to function with a minimal dialogue strategy which left learners in complete control. Several subjects expressed a preference for explicit direction in the interaction with DENISE rather than an open-ended dialogue.

In a declarative domain implicit task-based control is easily lost and the ISS needs to provide alternatives if the system is to be used for specific pedagogical objectives. It may be that an explicit curriculum or domain knowledge (either a 'primed' Learnt Model or a conventional domain model) needs to be added to the ISS architecture for declarative domains.

6.3 Domain Knowledge

DENISE functions without any prior knowledge of economics; other than the knowledge encoded in the conceptual syntax. Thus DENISE currently represents one extreme along a spectrum of possible ISS configurations – a dialogue strategy which does not refer to a domain model. At the other end is an ISS with a full domain model in the classical ITS tradition. Here the dialogue strategy can compare user statements with some 'correct' model. In such a situation the ISS is deceiving the learner – externally appearing to be a novice whilst internally having the knowledge of an expert. Such a system would go beyond a straightforward analogue of peer tutoring to move towards an 'optimal' student.

6.4 Conceptual Syntax

The conceptual syntax (system and user) in DENISE is fixed at compile-time and cannot be changed by the learner or the system. The constrained nature of the DENISE system meant that some subjects wished to express relationships that went beyond the qualitative causal links in the system.

There are three complementary approaches to producing a system with a more complicated conceptual syntax. Firstly, simply use a richer conceptual syntax; this could include structural knowledge, order of magnitude reasoning or knowledge about the temporal aspects of relationships. Secondly, the initial user conceptual syntax could be added to during the interaction – either by the system or a selection by the user. This could be achieved through a differential model – with a more complex system conceptual syntax gradually being added to the user conceptual syntax. Thirdly, allow the user to modify the conceptual syntax. This solution produces an adaptable system but requires that the dialogue strategy is constructed on the 'shifting foundations' of a changing conceptual syntax. This may be technically possible but greatly increases the difficulty of authoring the dialogue strategy.

7. Conclusions

Research into learning by teaching systems is still in an embryonic stage. These systems have the potential to produce novel forms of learning interactions which do not require the computational sophistication of many ILEs. The lack of a domain model produces domain independence at the expense of the designer's ability to control the direction of the interactions. Initial experiments with DENISE tend to support the basic premise that these systems can acquire individualised models from users.

These systems can also be used as a research tool to explore different aspects of learning by teaching experiences. (Kennedy, 1990) contends that much peer tutoring research lacks rigour and by providing an a controlled environment to study such interactions ISSs may also be able to contribute to better human-human learning by teaching.

References

Berliner, D. (1989), Being the teacher helps students learn, *Instructor*, 98(9), 12-13.

Berndsen, R. and Daniels, H. (1989), Sequential causality and qualitative reasoning in economics, in Pau, L.F. (Ed.), *Expert Systems in Economics, Banking and Management*, Elsevier Science, 125-135.

Blandford, A.E. (1994), Teaching through collaborative problem solving, *Journal of Artificial Intelligence in Education*, 5(1), 51-84.

Chan, T.W. and Baskin, A.B. (1990), Learning companion systems, in Frasson, C. and Gauthier, G. (Eds.), *Intelligent Tutoring Systems: at the Crossroads of Artificial Intelligence and Education*, Norwood, NJ: Ablex, 6-33.

Dillenbourg, P. and Self, J.A. (1992), PEOPLE POWER: A human-computer collaborative learning system, *Second International Conference on Intelligent Tutoring Systems,* Montreal, Canada, Springer-Verlag, 651-660.

Fisher, K. (1992), SemNet: a tool for personal knowledge construction, in Kommers, P.A.M., Jonassen, D.H. and Mayes, J.T. (Eds.), *Cognitive Tools for Learning*, Berlin: Springer-Verlag, 63-75.

Gasarch, W.I. and Smith, C.H. (1992), Learning via queries, *Journal of the ACM*, 39(3), 649-674.

Goodlad, S. (1985), Putting science in context, *Educational Research*, 27(1), 61-67.

Goodlad, S. and Hirst, B. (1989), *Peer Tutoring: a Guide to Learning by Teaching,* London: Kogan Page.

Hicks, J. (1979), *Causality in economics,* Oxford, UK: Blackwell.

Katz, S. and Lesgold, A. (1993), The role of the tutor in computer-based collaborative learning situations, in Lajoie, S.P. and Derry, S.J. (Eds.), *Computers as Cognitive Tools*, Hillsdale, NJ: Lawrence Erlbaum, 289-317.

Kennedy, M. (1990), Controlled evaluation of the effects of peer tutoring: are the 'learning by teaching' theories viable?, in Goodlad, S. and Hirst, B. (Eds.), *Explorations in peer tutoring*, Oxford, UK: Blackwell, 58-72.

Marcus, S. (Ed.) (1988), *Automating Knowledge Acquisition for Expert Systems*, London: Kluwer Academic.

Michie, D., Paterson, A. and Hayes-Michie, J. (1989), Learning by teaching, *2nd Scandinavian Conference on Artificial Intelligence '89,* Tampere, Finland, IOS, 307-331.

Nichols, D. (1994), *Intelligent Student Systems 1: placing learners in the role of tutors*, AAI/AI-ED Technical Report No. 103, Computing Department, Lancaster University.

Palthepu, S., Greer, J. and McCalla, G. (1991), Learning by teaching, *The International Conference on the Learning Sciences,* Evanston, Illinois, AACE, 357-363.

Polson, M.C. and Richardson, J.J. (Eds.) (1988), *Foundations of Intelligent Tutoring Systems*, Hillsdale, NJ: Lawrence Erlbaum.

Quinlan, J.R. (1986), Induction of decision trees, *Machine Learning*, 1(1), 81-106.

Self, J.A. (1992), Computational viewpoints, in Moyse, R. and Elsom-Cook, M. (Eds.), *Knowledge Negotiation*, London: Academic Press, 21-40.

Acknowledgements

Financial support for this research was provided by the Science and Engineering Research Council, Logica Cambridge Ltd and the "Student modelling in intelligent learning environments" (SMILE) project funded by the UK Research Councils' initiative on cognitive science and human-computer interaction.