A Belief Net Backbone for Student Modelling

Jim Reye

Queensland University of Technology
Brisbane, Q 4001, Australia
j.reye@qut.edu.au

Abstract. In this paper, I present a belief-net-based approach to student modelling which assists an ITS make determinations as to the extent of the student's knowledge. This approach also has advantages for structuring and ensuring the consistency of the student model. As well, the paper shows the desirability of using dynamic belief networks for modelling the dynamic evolution of the student's state of knowledge.

1 Introduction

In discussing the approaches of human instructors, Collins and Stevens (1982) state:

> Rather we assume only a partial ordering on the elements in the teacher's theory of the domain. ... The teacher's assumption is that students learn the elements in approximately this same order. Therefore, it is possible to gauge what the student will know or not know based on a few correct and incorrect responses.

In this paper, I present a belief-net-based approach to student modelling which assists an ITS make similar determinations as to the extent of the student's knowledge. This approach also has advantages for structuring and ensuring the consistency of the student model.

2 Belief Net Backbone

2.1 Probabilistic student modelling

For the purposes of this paper, I (minimally) assume that the domain may be viewed as an abstract collection of *topics* (or "curriculum elements"), each of which represents some piece of conceptual or skill knowledge which the student should acquire. For each such topic, there is a corresponding *student-knows* probability measure in the student model, e.g. p (student-knows (A)) = 0.7. In general, such measures of belief are dynamically updated as tutoring proceeds, e.g. see Corbett and Anderson (1992); Shute (1995). This initial student-modelling approach allows each probability in the student model to be updated independently of all the others. In the following section, I show why this is generally undesirable and why we need to model the impact of a change of belief on related beliefs.

2.2 The importance of the prerequisite relationship for structuring beliefs

When building a domain model (based on an expert's knowledge), there is really no need to specify prerequisite relationships between parts of that knowledge, as such relationships are only important for educational purposes and not for problem-solving within the domain. However, when student models are considered, the prerequisite relationship becomes very important. In the earlier quote from Collins and Stevens, the "partial ordering" is effectively a prerequisite relationship.

This prerequisite relationship is important both for modelling student beliefs at any one point in time and for deciding how best to alter those beliefs over a period of time. At any point in time, if knowledge of topic A is a *prerequisite* for knowledge of topic B, then it is inconsistent to assert that both p (student-knows (A)) = 0 and p (student-knows (B)) = 1. Expressed in terms of predicate logic, we want to enforce the constraint: ¬student-knows (A) ⇒ ¬student-knows (B), or student-knows (B) ⇒ student-knows (A) (equivalently). In probability theory, this constraint is represented as a *conditional probability*: p (student-knows (A) | student-knows (B)) = 1. This can also be expressed in a variety of logically-equivalent forms, e.g.: p (student-knows (B) | ¬student-knows (A)) = 0, which can be paraphrased as: "You can't know B if you don't know A."

2.3 Student modelling via belief networks

Probability theory provides the necessary methods for automated inferencing based on conditional probabilities. *Belief networks* (Pearl, 1988) provide a graphical way of designing probabilistic models based on the concept of conditional probability. As well, the resulting structure is used for subsequent automated reasoning about such models, in the most efficient manner for that structure. Being based on probability theory, belief networks also allow the representation of constraints which are not enirely certain. For example: p (student-knows (A) | student-knows (B)) = 0.9 can be interpreted as "Most students who know B, also know A." Being able to represent and reason with such knowledge is a valuable advantage over approaches based on traditional logic alone. Although the earlier description was only in terms of a pair of related topics, conditional probabilities allow the specification of relationships which are more complex than those given above. For example, we can specify that *both* P and Q are prerequisites for R.

As well as numeric values for the conditional probabilities, we also must specify the prior probabilities of all propositions which are not determined by the conditional probabilities. These prior probabilities specify the system's initial set of beliefs about a (typical) student, prior to the first interaction with that student. As the student uses the system, it directly updates its beliefs about the student's knowledge of topics where these are observable. These changes in belief are then propagated through the belief net, changing the system's belief in the likelihood that the student knows other (as yet) unobserved topics.

2.4 A backbone of conditional probability links

The use of conditional probabilities easily extends to much longer chains and networks of prerequisite dependencies. For example, if we know that A is a prerequisite for B, which is a prerequisite for C, and so on up until Z (say), then if we discover that the student knows Z, then we don't have to ask about the earlier prerequisites. Likewise, if we find out that the student knows F, but not G, then we don't have to ask about A..E or H..Z.

In the general theory of belief networks, there are no restrictions on the structure of the network (apart from the prohibition of directed cycles). However, in any system that uses a belief network, an actual structure must be specified. In this paper, I propose that the appropriate belief network structure for an ITS is based on two categories of nodes:

(a) a *belief net backbone*, which links all the "student-knows" nodes together in a partial ordering, according to their prerequisite relationships;

(b) a *topic cluster* for each node in the backbone, which consists of a single "student-knows" node together with a standard set of additional belief nodes. Most such nodes are *local* to the topic cluster, e.g. "student-interested-in (*topic*)". That is, there is a separate instance for each topic. However, some nodes are *global* in that there is only one instance in the whole student model, e.g. student-overall-aptitude ().

Thus these two categories overlap in that each of the "student-knows" *nodes* occurs once in the backbone and once in its topic cluster. However, the links (or "arcs") between these nodes (i.e. the conditional probabilities) do not overlap.

This proposal has two advantages. Firstly, it gives the designer a standard methodology for creating the structure of an ITS belief network, regardless of the particular domain. Secondly, there are computational advantages in that updates to the beliefs in any one topic subnet only affects the other topic subnets via the backbone, rather than there being any direct connection. In particular, this means that the impact of belief updates in a given topic subnet on its "student-knows" node can be calculated locally by considering just the nodes in that topic subnet, rather than having to propagate such updates through the entire network in order to determine their nett results. The efficiency gained by such local computation is very important during instructional planning, when the impacts of large numbers of possible plans must be determined rapidly. (Once a plan has been chosen and is executed, its effects update one or more topic subnets, and these effects must be propagated through the backbone. Although computationally more expensive, such updates occur much less frequently than those needed during planning.)

3 Dynamic belief networks

In ordinary belief networks, it is assumed that the properties of the external world, modelled by the network, are unchanging. That is, even though the system may gather information from the external world which causes it to modify its measures of belief about items in that world, those items remain either true or false.

Such an approach is clearly inadequate for student modelling in a tutoring system, where we must be able to represent the dynamic evolution of a student's knowledge over a period of time. Dynamic belief networks (Dean and Kanazawa (1989)) allow for reasoning about change over time. This is achieved by having a sequence of nodes which represent the state of the external item over a period of time, rather than having just a single temporally-invariant node. For real-world continuous processes, the sequence of nodes may represent the external state as it changes over a sequence of time-slices. For tutoring, it is often more useful to represent changes in the student model over a sequence of interactions, rather than time-slices, as the following example illustrates.

To avoid any possible misunderstanding, I point out that dynamic belief networks model a (moving) snapshot of the student's knowledge. They do not model the evolution itself, in the sense of providing a history of the development of such knowledge. Such a history may well be required, e.g. (i) for deciding to revisit domain areas which were previously of major difficulty to the student; and (ii) to dynamically adapt the system to the student's learning style. Clearly, the complete history of any dynamic model can be captured (and reviewed) by keeping a tuple-based log of redo/undo changes, analogous to the logging done by many database systems. (Such a log may already be present in any system which is able to resume a tutorial session following a system failure.) The subsequent usage of such historical data for instructional planning is outside the scope of this paper.

3.1 A basic dynamic belief network: probabilistic modelling in the ACT Programming Languages Tutor

The ACT Programming Languages Tutor (Corbett and Anderson (1992)) uses a simple two-state psychological learning model with no forgetting, which is updated each time that the student has an opportunity to show their knowledge of a production rule in the ideal student model. There are four parameters associated with each rule:

- $p(L_0)$ the probability that a rule is in the *learned* state prior to the first opportunity to apply the rule (i.e. from reading text);
- $p(T)$ the probability that a rule will make the transition from the *unlearned* state to the *learned* state following an opportunity to apply the rule;
- $p(C|U)$ the probability that a student will guess correctly if the applicable rule is in the *unlearned* state;
- $p(E|L)$ the probability that a student will slip and make an error when the applicable rule is in the *learned* state.

In general, the values of these parameters may be set empirically and may vary from rule to rule, but Corbett and Anderson describe a study in which these parameters were held constant across 21 rules, with $p(L_0) = 0.5$, $p(T) = 0.4$, $p(C|U) = 0.2$ and $p(E|L) = 0.2$. Note that there are no conditional probabilities linking different rules i.e. no prerequisite constraints. In an appendix to their paper, the authors briefly state equations for calculating $p(L_n|C_n)$ and $p(L_n|E_n)$, which then can be used to determine the probability that a production rule is in the learned state following a correct (C_n) or erroneous (E_n) student response, at the nth opportunity.

In this section, I illustrate the applicability of dynamic belief networks by showing how Corbett and Anderson's equations can be rederived from the dynamic belief net shown in Figure 1. Structurally, this is the simplest possible dynamic belief network for student modelling (where simplicity may be a virtue rather than a vice). (In their paper, Corbett and Anderson did not describe how they derived these equations. Even though they did not refer to dynamic belief networks, their learning model is most likely mathematically isomorphic to Figure 1, leading to an analogous derivation of the formulae below. My goal here is not just to rederive their results, but to show that their formulae may be viewed as a special case of the general approach of using dynamic belief networks.) To fully specify the dynamic belief net in Figure 1, we need:

(a) the prior probability that a rule is in the learned state, prior to the first opportunity, i.e. $p(L_0)$, as given by the authors;

(b) the conditional probabilities linking L_{n-1} and C_n in Figure 1, i.e.:
- the probability of a correct response when the rule is in the learned state just prior to the nth opportunity, i.e. $p(C_n|L_{n-1}) = p(C|L) = 1 - p(E|L)$, where the latter two parameters are used by the authors; and
- the probability of a correct response when the rule is in the unlearned state just prior to the nth opportunity, i.e. $p(C_n|\neg L_{n-1}) = p(C|U) = 1 - p(E|U)$, where the latter two parameters are used by the authors.

(c) the conditional probabilities linking L_{n-1} and L_n in Figure 1, i.e.:
- the probability of remaining in the learned state when the rule is already in the learned state just prior to the nth opportunity, i.e. $p(L_n|L_{n-1}) = 1$, where this value is specified by the authors' assumption of no forgetting; and
- the probability of a transition to the learned state when the rule is in the unlearned state just prior to the nth opportunity, i.e. $p(L_n|\neg L_{n-1}) = p(T)$, where the latter parameter is used by the authors.

Given a value for $p(L_{n-1})$, it is easy to calculate $p(C_n)$ via:

$$p(C_n) = p(C_n|L_{n-1})\ p(L_{n-1}) + p(C_n|\neg L_{n-1})\ p(\neg L_{n-1}) \quad (1)$$

This value can then be used to revise the belief in $p(L_{n-1})$ when C_n is true (i.e. the response is correct):

$$p(L_{n-1}|C_n) = p(C_n|L_{n-1})\ p(L_{n-1})\ /\ p(C_n) \quad \text{(Bayes' theorem)} \quad (2)$$

Finally, the revised belief in $p(L_{n-1})$ can be used to calculate the new belief $p(L_n)$, when C_n is true:

$$p(L_n|C_n) = p(L_n|L_{n-1})\ p(L_{n-1}|C_n) + p(L_n|\neg L_{n-1})\ p(\neg L_{n-1}|C_n) \quad (3)$$

Under Corbett and Anderson's assumption that $p(L_n|L_{n-1}) = 1$, equation (3) becomes:

$$p(L_n|C_n) = p(L_{n-1}|C_n) + p(L_n|\neg L_{n-1})\ p(\neg L_{n-1}|C_n)$$

or equivalently (rewriting in terms of their original parameters):

$$p(L_n|C_n) = p(L_{n-1}|C_n) + p(T)\ (1 - p(L_{n-1}|C_n))$$

which is the same as their equation [1]. Likewise, equation (1) above can be substituted into equation (2) and then also be rewritten in terms of their original

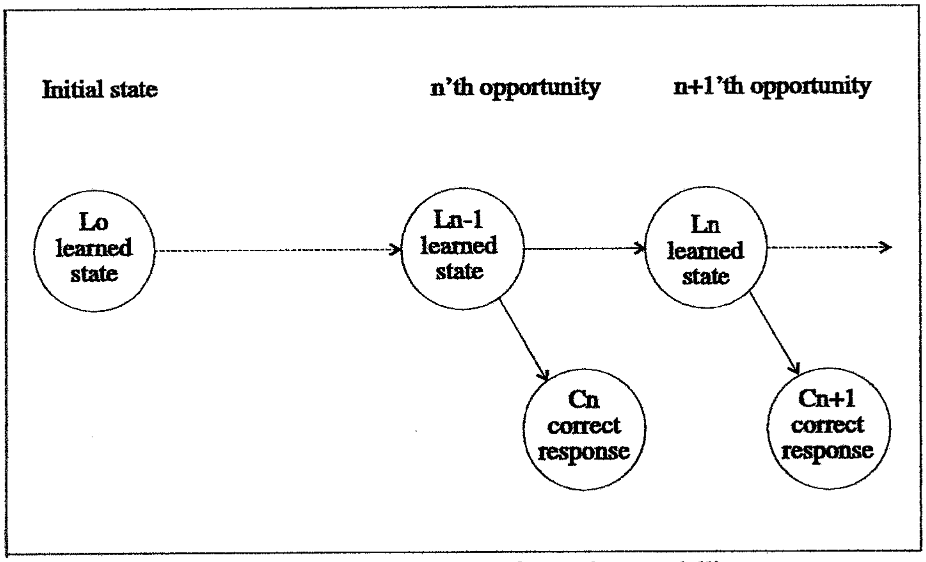

Figure 1 A basic dynamic belief network for student modelling

parameters:

$$p(L_{n-1}|C_n) = p(C|L)\ p(L_{n-1})\ /\ (\ p(C|L)\ p(L_{n-1})\ +\ p(C|U)\ p(U_{n-1})\ )$$

which is their equation [3]. For space reasons, I omit the analogous derivation of equations for $p(L_n|\neg C_n)$ and $p(L_{n-1}|\neg C_n)$, equivalent to Corbett and Anderson's equations [2] and [4].

3.2 A more elaborate dynamic belief network

Figure 2 shows a somewhat more elaborate dynamic belief network for a topic cluster, corresponding to the topic cluster attributes introduced earlier in this paper. This figure is intended to further illustrate the general approach, rather than attempting to be a comprehensive network. As before, each arrow represents a conditional probability which is usually a cause-effect relationship.

4 Diagnosis: determining the state of student knowledge

The earlier quote from Collins and Stevens described how human teachers are able to gauge the extent of a student's knowledge based on a relatively small number of probing questions. This teaching strategy may be modelled as a problem-solving procedure within the framework of a classic AI diagnostic task. In particular, de Kleer and Williams's (1987) research on their General Diagnostic Engine (GDE) system is helpful, even though it cannot be used directly as it is based on some assumptions which do not apply to student modelling. That research is based on the idea of minimising the number of measurements (analogously, minimising the number of questions asked of the student) by making a series of measurements,

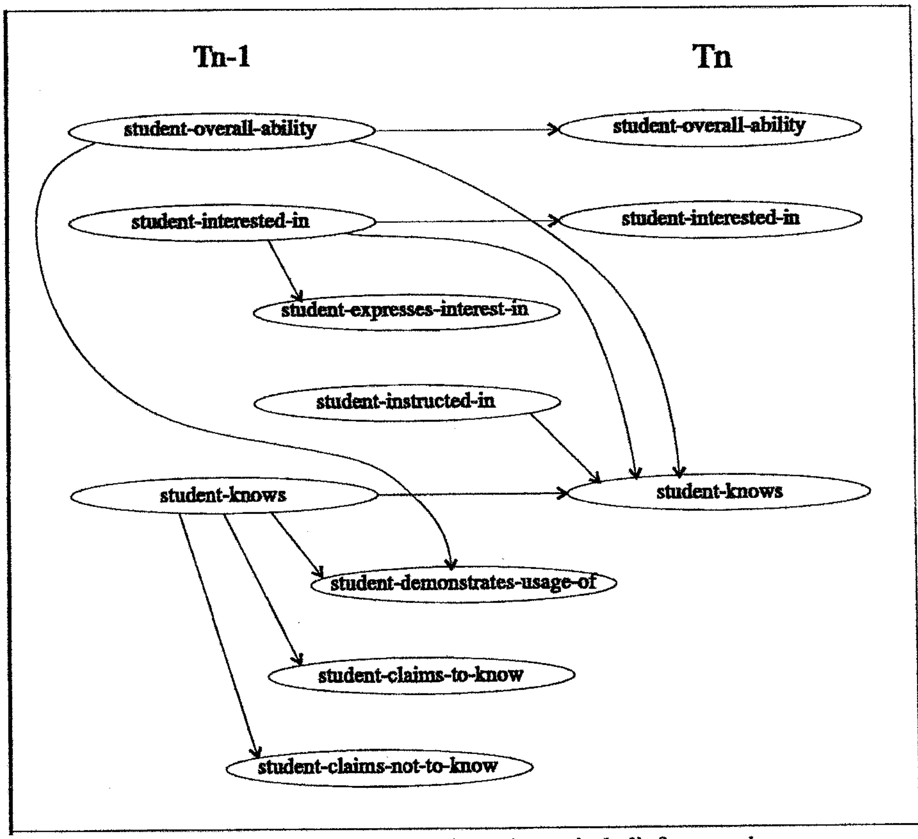

Figure 2 An example topic cluster, in a dynamic belief network

each of which maximises the expected amount of information gained by that measurement. Technically, minimising the expected entropy (H) of the belief network after making that measurement: $H = -\sum p_i \log p_i$. So, the expected entropy (H_e) after asking a student whether they know topic t_n is given by the weighted sum of the two possible responses:

$$H_e(t_n) = p\,(t_n)\;H\,(t_n) \;+\; p\,(\neg t_n)\;H\,(\neg t_n)$$

One of the difficulties faced by the GDE procedure is that the number of possible combinations of faults grows exponentially with the number of components. Fortunately, for student diagnosis, the number of possible combinations is far less. This is because, whenever an ITS considers a possible diagnosis involving a particular faulty node, then all subsequent (partially ordered) nodes must also be faulty (at any one point in time). By comparison, in an electronic circuit, subsequent nodes need not be faulty and so there are more cases to consider.

For example, consider a simple chain of four items: A → B → C → D. If this is taken as representing an electronic circuit of buffers, then there are 16 (= 2^4) possible combinations of possibly faulty components. Alternatively, if this chain is a belief net backbone representing the prerequisite relationships linking four topics, then there are only five combinations which model possible states of the student's knowledge, as given by the following sets: {}, {A}, {A, B}, {A, B, C}, {A, B, C, D}. Such linear growth is clearly better than exponential growth, especially when creating domains containing hundreds of topics.

I now give a small example to illustrate the approach. In this example, I omit the "student-knows" predicate for conciseness and clarity, e.g. "p (A)" should be taken as "p (student-knows (A))". *Example:* A → B → C → D, with

$$p(A) = 0.75 \quad \therefore p_{prior}(A) = 0.75$$
$$p(B \mid A) = 0.75 \quad \therefore p_{prior}(B) = 0.56$$
$$p(C \mid B) = 0.75 \quad \therefore p_{prior}(C) = 0.42$$
$$p(D \mid C) = 0.75 \quad \therefore p_{prior}(D) = 0.32$$

There are four topics, so there are four possible questions which could be asked. The expected entropy for each of these possibilities are calculated as:

$$H_e(A) = p(A)\,H(A) + p(\neg A)\,H(\neg A) = 0.98$$
$$H_e(B) = p(B)\,H(B) + p(\neg B)\,H(\neg B) = 0.67$$
$$H_e(C) = p(C)\,H(C) + p(\neg C)\,H(\neg C) = 0.69$$
$$H_e(D) = p(D)\,H(D) + p(\neg D)\,H(\neg D) = 0.95$$

These values confirm what one would expect intuitively in this case, i.e. that more information is gained by asking about B or C, rather than A or D. More precisely, topic B has the lowest expected entropy (i.e. highest expected gain of information) and so should be asked first. The student's reply can then be used to update the probabilities in the student model. These revised probabilities can then be used in subsequent calculations of expected entropy, in order to determine which topic should be queried next. The resulting dialogue models the behaviour of the human teachers described above.

5 Related research

Most existing ITSs use fairly coarse-grained measures for representing the student's knowledge. For example, some systems associate one of three values with each domain topic: student knows, student does not know, and not sure if student knows or does not know. Such a coarse-grained measure limits the modelling power of the system and so limits its decision-making capabilities.

For example, consider a situation in which an ITS has to choose between two equally important topics to teach next (both with prerequisites satisfied). Assume that the probability that the student already knows the topic is 0.4 in one case and 0.7 in the other (based on observations of the student). Most systems would be forced to represent both these as "not sure if student knows or does not know", and thus could pick either topic to teach next. This is an inferior approach, as it is clearly better to be able to distinguish the two. For example, if the student has been doing well so far, then we may wish to cover the topic which the student appears more likely not to know. On the other hand, if the student has been doing poorly, then we may wish to work on the topic which the student is more likely to know, in order to improve their self-confidence.

It is only recently that ITS researchers have started investigating the use of probability theory for student modelling. Villano (1992) provides a good but very brief discussion of some of the basic issues in using a belief network for student modelling, such as the need to obtain the structure, prior probabilities and conditional probabilities for the belief network — issues which are also covered in this paper. Pirolli and Wilson (1992) discuss the important diagnostic issue: how to estimate probabilistic student modelling parameters from complex events. The last section of their paper also addresses the basic idea of using belief networks for student modelling. While limited in scope, this section illustrates these ideas, by presenting screen snapshots from Hugin (a belief network shell) showing three different states of a small (seven node) belief network. Shute's (1995) work on updating probabilistic measures of a student's skills is important for the future development of more sophisticated student models based on belief networks, even though she just considers the updating of individual measures in the student model.

Acknowledgments

I thank the anonymous referees for their constructive comments.

References

1. Collins, A., and Stevens, A.L. (1982) Goals and strategies for inquiry teachers. In Glaser, R. (ed.) *Advances in Instructional Psychology (vol. II)*, 65-119.
2. Corbett, A.T., and Anderson, J.R. (1992) Student modeling and mastery learning in a computer-based programming tutor. In Frasson, C., Gauthier, C. and McCalla, G.I. (eds.) *Intelligent Tutoring Systems (ITS'92)*, 413-420.
3. de Kleer, J., and Williams, B.C. (1987) Diagnosing multiple faults. *Artificial Intelligence* 32, 97-130.
4. Dean, T., and Kanazawa, K. (1989) A model for reasoning about persistence and causation. *Computational Intelligence* 5, 142-150.
5. Pearl, J. (1988) *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.
6. Pirolli, P., and Wilson, M. (1992) Measuring learning strategies and understanding: a research framework. In Frasson, C., Gauthier, C. and McCalla, G.I. (eds.) *Intelligent Tutoring Systems (ITS'92)*, 539-558.
7. Shute, V. (1995). SMART evaluation: cognitive diagnosis, mastery learning & remediation. In Greer, J. (ed.) *Artificial Intelligence in Education, 1995*, 123-130.
8. Villano, M. (1992) Probabilistic student models: bayesian belief networks and knowledge space theory. In Frasson, C., Gauthier, C. and McCalla, G.I. (eds.) *Intelligent Tutoring Systems (ITS'92)*, 491-498.