A GRAPH BASED APPROACH TO LEARNER PROFILING IN AN INTELLIGENT TUTORING SYSTEM

Sanjay Singh
Research Scholar, Department of Computer Science and Engineering, Chaudhary Devi Lal University,
Sirsa, Haryana, India
bhaambhu@gmail.com
http://bhaambhu.com

Vikram Singh
Professor, Department of Computer Science and Engineering, Chaudhary Devi Lal University,
Sirsa, Haryana, India
vikramsinghkuk@yahoo.com

Abstract

Learner profiling lays the foundations of the personalization that happens in adaptive educational applications and Intelligent Tutoring Systems (ITS). A learner's level of knowledge on a topic is estimated from their performance on certain activities related to the topic. For this, researchers have devised many model extensions throughout the years that incorporate specific cognitive features into student profiling. In this paper, a new graph-based algorithm for learner profiling has been proposed that is able to adapt the course to the current knowledge level of the learner using the topic dependencies fed in by the subject experts and the past response data of learners who have taken this course in the past. This results in learner profiling with minimum number of assessment activities in the best case.

Keywords: **Learner Profiling; Student Profiling; Adaptive Learning; Intelligent Tutoring Systems.**

1. Introduction

Following the invention of computers there have been countless endeavors at using them to enhance the noble industry of tutoring. Facilitating the job of teaching with the use of electronic technologies falls under the huge umbrella of e-learning, which is a far-reaching discipline that covers the analysis of all conjunctions of technology and education. A more appropriate definition by researchers in [1] states "Educational technology is the study and ethical practice of facilitating learning and improving performance by creating, using, and managing appropriate technological processes and resources".

In state-of-the-art e-learning applications, the learner exercises a great deal of interaction with the application. In the brick-and-mortar model of teaching-learning, a teacher, besides delivering a lecture, communicates with the students quite significantly. A teacher adjusts his teaching in response to the student's learning progress. Since more than half-a-century now, researchers are attempting to embody this feature in e-learning systems. Most researches have suggested a solution of this problem through Artificial Intelligence (AI) based approaches. In AI based learning systems the human teacher-human taught interactions are endeavored to be replaced by software teacher-human taught interactions to a considerable extent.

This leads us to the realm of Intelligent Tutoring Systems (ITS). These systems realize established teaching-learning processes by way of AI with an intention of delivering learner-adapted tutoring without any direct mediation of a human teacher. The Intelligent tutoring systems' are aimed to put AI techniques to use in delivery of instruction which would have been branded as "Good Teaching" if it were delivered by a human teacher [2]. An ITS can employ any architecture provided that it delivers intelligent tutoring though the original discussions on the architecture of an ITS de-scribe these four crucial modules: The Student Model, The Domain Model, The Tutor (pedagogical) Model and The Interface Model [3–5].

The Student Model is often considered as the core component of an ITS and it deals with the task of estimating the student's learning progress and comparing it with the conditions or constraints embedded in the tutor. The progress estimated by the student model of an ITS is used by other parts of the system for adjusting the tutoring behavior and for various other tasks. This paper aims to propose a method of learner profiling – accurately profiling the knowledge level of the learner with respect to a given course in the shortest possible time quantum.

Learner profiling is a well-known problem that e-learning and ITS researchers have been trying to solve since decades and it deserves a brief review.

2. Materials and Methods

The study of learner profiling gathers great enthusiasm from cognitive researchers across the globe. Even though years of research on ITSs have shown countless architectural modifications, with some researchers adding more components to the classical four component architecture, while others combining two or more components together, or removing some of them for some specific purpose, the student modelling component holds solid ground in the ITSs, whether it's being mentioned by the same name or being renamed to sound something different. For illustration, model tracing is considered a major research area in the domain modelling component in an ITSs, but it can also deliver as a student model as shown by researchers in [6].

Predicting numerous behaviors of a student and assessing the effect of those behaviors on learning poses greater research-challenges. One such variable which is widely popular for a prediction challenge is the prediction of the student's performance – how well is the student going to perform in the next practice opportunity or question. If an ITS can predict how well the student is going to perform on which areas, the system can adapt to the student and provide them with a better learning experience. For this, there have been learning-curve based student models that plot the performance of the learner with some observed behavior of the learner such as success rate, error rate, mastery rate, etc. that demonstrate how students perform as an outcome of practicing [7]. Additive Factors Model (AFM) is a learning curve model by [8] to conform a learning curve to the student performance data. This model was proposed into the ITS environment by [9] in which the model was renamed by the researchers to Learning Factors Analysis model (LFA). This model captures the capabilities of the learner and the easiness of the necessary skills. Another approach called Learning Decomposition (LD) is a generalization of learning curves and predicts comparative benefits of varied learning opportunities [10].

Questionnaires, exams and quizzes have been widely used traditionally, as well as today, for evaluating a learner's performance. The key idea is that a student's knowledge can be estimated by observing the student's performance in questions (a generalized term used in this paper to represent any practice opportunity). A Hidden Markov Model (HMM) is a Markov Model in which there is a system X that has some hidden (unobservable) states and then there is a process Y whose behavior de-pends upon X and the aim is to determine X by monitoring Y [11]. Using this and Bayesian networks, Bayesian Knowledge Tracing (BKT) was proposed in [12] which takes the form of a HMM and works on the notion that a student's performance is observable, but it depends on the student's knowledge, making the student's performance the observed state and the student knowledge the hidden variable here.

There are a number of caveats which concern the validity of using BKT in a general purpose and expandable system which attempts to teach courses from all domains, the most prominent one of which is that BKT works for skills whereas a general purpose system may consist of topics and courses that may be an imperfect amalgamation of multiple skills or that might not be a skill at all, and moreover certain skills might stretch across topics as demonstrated by researchers in [13] who have integrated scaffolding with BKT to help the learner acquire these skills but for a system that concerns with a learner acquiring the knowledge of certain topics that might not evenly map onto skills, a general purpose learner profiling approach that customizes the course to the learner has been proposed in this paper. The two challenges addressed in this paper are:

- Adapting the course to the learner – With the intelligent profiling algorithm, we are able to adapt any course that the learner wants to take to the knowledge level of the learner in a very short amount of time, shortening the total amount of time needed to complete the course tremendously and hence the overall learning process. The amount of time saved for the learner depends upon the number of topics detected to be known and the individual lengths of those topics.
- Improving the course for everyone – the mappings obtained illustrate the degree of actual dependencies between topics. The subject expert specifies the dependencies between topics and creates relationships mappings illustrating which topic depends on which one. These dependencies assist the overall learning experience for the learner, but the actual degree of dependencies between the topics is further refined by the data obtained from initial learner profiling. This improves the overall mappings between topics, thus creating better learning experience for future students.

In addition to this, understanding our proposed approach requires a brief introduction of the following two terms:

2.1. *Directed acyclic graph:*

A graph is a collection of vertices, pairs of which may be connected together by edges. A directed graph, also known as a digraph, is a graph in which edges have orientations [14]. A dependency graph is a data structure formed by a directed graph that describes the dependency of an entity in the system on other entities of the system. The type of relationships between topics that have been addressed in this paper can be realized through a dependency graph with a vertex for each topic and an edge connecting two topics whenever one of them needs to be assessed earlier than the other. As it must be common sense that a topic cannot have itself as its pre-requirement, we re-quire a dependency graph without any circular dependencies which turns out to be a Directed Acyclic Graph [15].

2.2. *Precision*

In pattern recognition, information retrieval and classification (machine learning), precision is a performance metric that applies to the data retrieved from a sample space, corpus or collection. It is basically the fraction of relevant instances among the retrieved instances.
Precision is defined as

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} = \frac{True\ Positive}{Total\ Predicted\ Positive} \quad (1)$$

Where True Positive are the number of instances which were positive and were correctly predicted whereas False Positive are the number of instances which were negative but were predicted to be positive [16]. Consider a program that is built for recognizing horses (the relevant element) in an image. On processing an image that consists of ten cows and twelve horses, the program identifies eight horses. Out of the eight elements identified as horses, only five actually are horses (true positives), while the other three are cows (false positives). Seven horses were missed (false negatives), and seven cows were correctly excluded (true negatives). The program's precision is then 5/8 (true positives / selected elements).

3. Approach

The idea of a general-purpose learner profiling approach relies upon a general-purpose representation of a study curriculum that can be re-used and adapted to all sorts of subjects and courses. Such a representation is realized in this paper through the notion of a *topic* – a learnable and evaluable unit of knowledge, that may require prior knowledge of one or more topics, as illustrated in the Fig. 1. The structure has theory and questions, as well as an evaluation activity that denotes an activity that helps evaluate that a learner possesses the complete knowledge of this topic – by simply completing the evaluation activity or by more complicated ways.

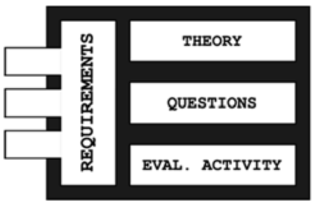

Fig. 1. General purpose representation of a topic of study.

Given that, a sequence of topics that one has to complete in order to learn a subject or a skill has been termed as a course. Any student may need to learn multiple subjects or skills to achieve their desired study goal, as for this have to complete multiple courses. Whenever a student decides to take a course that contains multiple core topics, there may be one or more topics in that course that this specific student already knows, and does not need to study again. If we are able to quickly identify those topics, then we can customize the course according to the currently acquired knowledge of the student, which in turn saves a lot of time for the student that needs to complete the course. This adaptivity can be used in multiple areas but in this case, it has been used to adapt the course to the student.

The task is to estimate the learner's knowledge profile with the minimum set of evaluation activities. Our algorithm requires topic inter-dependency information, which is typically provided by subject experts while drafting the course. In addition to this, data of responses of past students that undertook the same course is used to improve the topic interdependency information – taking a safer approach for a general-purpose tutoring system when there is a possibility that the subject expert might not be able to map accurate relationships between topics perfectly. These two help the minimal learner profiling algorithm build the learner's knowledge profile as illustrated in Fig. 2 and it does that in the minimum amount of time possible.

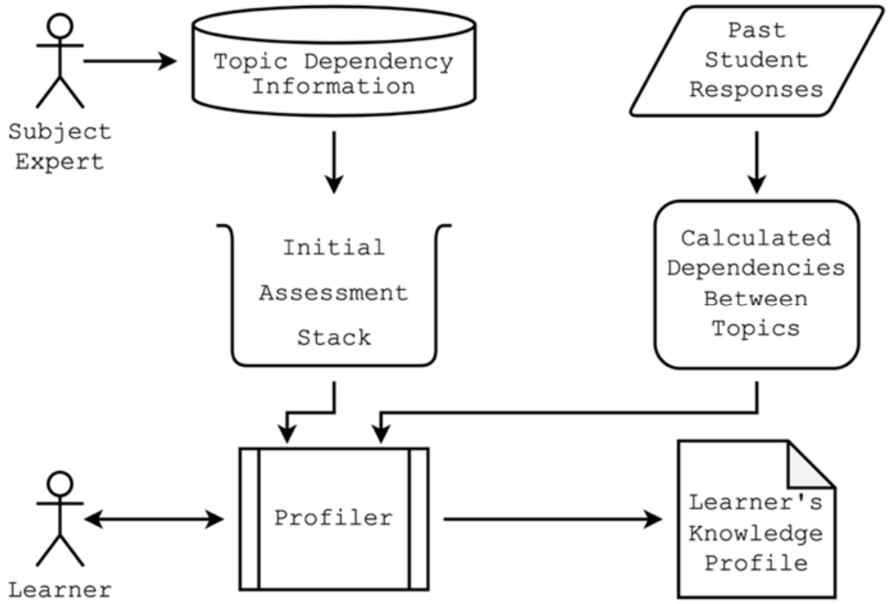

Fig. 2. The proposed model for leaner profiling.

The topics that are contained in any course may not all be independent topics. There are some topics that can be independently learnt and there are topics that rely upon prior knowledge of other topics. For any subject expert that lays out topic dependencies' information, it is always convenient to specify the topics that any topic 'A' pre-requires, rather than looking at a topic 'A' and foreseeing all the other topics this topic 'A' will be required for, even in the future. This requires a representation for storing topic inter-dependencies using every topic's pre-requirement information, and one of these representations is illustrated in Table 1.

Table 1. Topic inter-dependency representation.

topic	1	2	3	4	5	6	7	8	9	10
requires	10	-	-	-	-	-	6,2	10,7	10,2	4,6

These type of relationships between topics can be realized through a dependency graph with a vertex for each topic and an edge connecting two topics whenever one of them needs to be assessed earlier than the other. As it must be common sense that a topic cannot have itself as its pre-requirement, we require a dependency graph without any circular dependencies which turns out to be a Directed Acyclic Graph [15] as illustrated in Fig. 3.

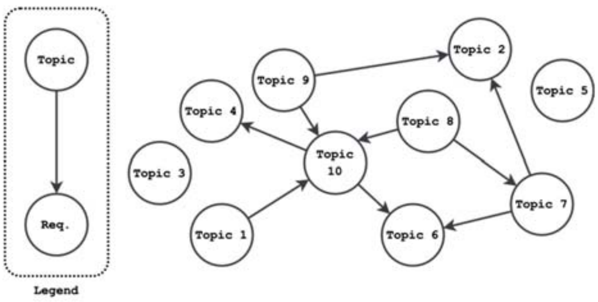

Fig. 3. Topic relationships realized with a directed acyclic graph.

Every topic has a set of one or more assessment activities that are used to evaluate the state of the knowledge of that topic for any learner. First, the algorithm uses the topic-interdependencies information fed by the subject expert to create a topic sequence. For this, the '`requires`' values from topic dependencies are used to create their inverse values labelled as '`required_for`' as illustrated in Table 2.

Table 2. Table with required_for values.

topic	1	2	3	4	5	6	7	8	9	10
requires	10	-	-	-	-	-	6,2	10,7	10,2	4,6
required_for	-	7,9	-	10	-	7,10	8	-	-	1,8,9

To assess knowledge in multiple topics with the minimum set of assessment activities, we start by creating an empty stack and initializing it by pushing those topics in reverse whose ‘required_for’ is empty, the algorithm for which has been illustrated in Fig. 4.

```
SET reversedCourse to the reversed course
FOR topic IN reversedCourse
  IF required_for of topic is empty THEN
    PUSH singleTopic TO assessmentStack
  ENDIF
ENDFOR
```

Fig. 4. Algorithm for creating initial assessment stack.

This gives us an initial assessment stack as illustrated in Fig. 5. As a stack allows reading only one element at a time and allows operations on the top only, the elements popped from the stack will guide the order of assessment of topics in the assessment algorithm.

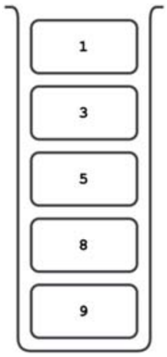

Fig. 5. Initial State of the Assessment Stack.

A dataset is required for calculating the relative degree of dependencies between topics. For this, an initial pre-defined set of learners is subjected to all the assessment activities, without applying the proposed minimal learner profiling algorithm. The data collected in respect to assessment of 60 learners assessed over 10 topics is shown in Table 3.

Table 3. Topic assessment data.

Assessment Activities

Student Responses	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
S_1	1	1	0	1	0	1	0	0	0	1
S_2	0	0	0	1	0	1	0	0	0	1
S_3	1	1	0	0	0	0	1	1	1	1
S_4	1	0	1	1	0	1	0	1	1	0
S_5	0	1	1	0	1	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...
S_{60}	1	0	0	1	0	0	0	1	1	0

The assessment data of 60 learners over 10 different topics from 10 different courses of study have been housed in Table 3 in form of dichotomous responses correct (1) and incorrect (0). This data is used to determine the degree by which a topic's knowledge implies other (0 or more) topics' knowledge and this has been done by calculating the relative precision metrics. Precision metric has been chosen, for it is the appropriate metric to use when the costs of false positives is high. In the present context, a topic that the student actually does not know, but has been predicted otherwise, is a false positive.

The dataset in Table 3 has been sliced to select all those rows where the student knows topic 1, that is, where A_1 is 1. This gives us a representation similar to what is illustrated in Table 4.

Table 4. Sliced table-of-responses for A_1.

Assessment Activities

Student Responses where A_1 is 1		A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
	S_1	1	1	0	1	0	1	0	0	0	1
	S_3	1	1	0	0	0	0	1	1	1	1
	S_4	1	0	1	1	0	1	0	1	1	0
	S_8	1	1	0	1	1	1	1	1	1	1
	S_9	1	1	0	1	0	1	1	1	1	1
	...	...	...	...	...	...	...	...	...	...	...
	S_{60}	1	0	0	1	0	0	0	1	1	0

Having a table of responses of students who know topic 1, we calculate the precision of the prediction of knowledge of other topics by predicting each topic to be known given that topic 1 is known, where the knowledge of a topic's assessment activity is considered analogous to the knowledge of the topic. This gives us Table 5.

Table 5. Relative precision of the predictions of knowledge of other topics when knowledge of topic 1 is confirmed.

	A2	A3	A4	A5	A6	A7	A8	A9	A10
Given A1	0.695652	0.173913	0.869565	0.173913	0.739130	0.695652	0.826087	0.608696	0.826087

Doing this for every topic, we get Table 6 for precision of prediction of knowledge of other topics given knowledge of a topic for every topic.

Table 6. Prediction precision table.

	Given A1	Given A2	Given A3	Given A4	Given A5	Given A6	Given A7	Given A8	Given A9	Given A10
A1	1.000000	0.457143	0.444444	0.512821	0.444444	0.377778	0.551724	0.791667	0.666667	0.542857
A2	0.695652	1.000000	0.666667	0.589744	0.666667	0.644444	0.896552	0.791667	0.809524	0.742857
A3	0.173913	0.171429	1.000000	0.102564	0.777778	0.133333	0.206897	0.250000	0.238095	0.114286
A4	0.869565	0.657143	0.444444	1.000000	0.444444	0.688889	0.620690	0.833333	0.857143	0.857143
A5	0.173913	0.171429	0.777778	0.102564	1.000000	0.111111	0.206897	0.250000	0.238095	0.114286
A6	0.739130	0.828571	0.666667	0.794872	0.555556	1.000000	0.862069	0.833333	0.809524	0.885714
A7	0.695652	0.742857	0.666667	0.461538	0.666667	0.555556	1.000000	0.833333	0.523810	0.628571
A8	0.826087	0.542857	0.666667	0.512821	0.666667	0.444444	0.689655	1.000000	0.714286	0.571429
A9	0.608696	0.485714	0.555556	0.461538	0.555556	0.377778	0.379310	0.625000	1.000000	0.485714
A10	0.826087	0.742857	0.444444	0.769231	0.444444	0.688889	0.758621	0.833333	0.809524	1.000000

For example, when assessment activity 1 is correct there is an approximate 82.61% chance that assessment activity 10 will be correct and an 86.96% chance that assessment activity 4 will be correct. This supports our course's topic dependencies specified in Table 1 that topic 1 requires the knowledge of topic 10, which is evident as the data gives us the precision of 82.61% but the topic dependencies information never specified any direct relation between topic 1 requiring the knowledge of topic 4, but the data tells us it's 86.96%.

There are many such hidden topic dependencies revealed in this precision table, and there are two reasons for this. One is the amount of initial response data we collect on our course. The more data we have, the accurate our precision table reveals the actual dependencies. The second reason is the existence of indirect dependencies formed by a chain of direct dependencies – as seen in the original topic dependency information fed by subject expert topic 1 requires knowledge of topic 10, and topic 10 requires knowledge of topic 4 and 6 both, this explains the higher percentage of A_4 and A_6 given A_1 is known.

Now that we have the precision table and assessment stack ready, we can initiate our learner profiling algorithm illustrated in Fig. 6 that utilizes both of these and an additional ‘threshold’ parameter that controls the overall algorithm.

```
FOR topic IN course
    SET ‘knows’ value of topic to ‘untested’
WHILE assessment stack is not empty
    POP the top of assessment stack to obtain currentTopic
    OBTAIN studentResponse by assessment of currentTopic
    IF studentResponse is correct THEN
        SET ‘knows’ of currentTopic to ‘Yes’
        FOR topic IN ‘requires’ of currentTopic (in reverse)
            IF ‘knows’ of topic is ‘untested’
                IF precisionTable entry for [topic,currentTopic] is less than threshold THEN
                    PUSH topic TO assessmentStack
                ELSE
                    SET ‘knows’ of topic to ‘Yes’
                ENDIF
            ENDIF
        ENDFOR
    ELSE
        SET ‘knows’ of currentTopic to ‘No’
        FOR topic IN ‘requires’ of currentTopic (in reverse)
            IF ‘knows’ of topic is ‘untested’ THEN
                PUSH topic TO assessmentStack
                    ENDIF
        ENDFOR
    ENDIF
ENDWHILE
```

Fig. 6. The learner profiling algorithm.

The learner profiling algorithm works in the following way: If a learner is assessed over a topic (say topic 1) and after assessment it is found that the learner does not know topic 1, the system proceeds to assess the learner’s knowledge on the topics that topic 1 requires the knowledge of, that is, the leaner’s knowledge assessment on topic 1’s pre-requirements is carried out. On the other hand, if after the knowledge assessment of topic 1 it is found that the learner knows topic 1, its pre-requirement topics are only assessed if their dependency prediction precision (from the precision Table 6) is found to be less than the threshold parameter of the algorithm. A stack data structure is maintained called ‘`assessmentStack`’ which houses the topics that need to be assessed. The topic on the top of the `assessmentStack` is assessed, and based on its assessment results and the corresponding values in the precision Table 6 it is decided whether more topics will be pushed on to the stack or not. When the `assessmentStack` becomes empty the learner’s knowledge profile on the whole course has been evaluated in the minimum number of assessment activities.

4. Results and Conclusions

In this paper we proposed an algorithm for adapting a course to the knowledge level of the learner in the minimum possible time. For the basis of our general-purpose algorithm to work we proposed a notion of a general-purpose topic that is not domain specific, and may or may not require prior knowledge of one or more topics and the notion of a course that is a sequencing of a set of topics that a learner may complete in order to learn a subject or skill.

The algorithm is then able to adapt the course to the knowledge levels of a learner using the topic dependencies fed in by the subject experts and the past response data of learners who have taken this course in the past. This results in learner profiling with minimum number of assessment activities in the best case. The trust threshold is a user-tweakable parameter that decides as to how much the algorithm should trust the quality of the data collected from previous learners. The assessment activities with respect to the value of the trust threshold parameter are illustrated in Table 7.

Table 7. Different assessment sequences according to trust thresholds.

Threshold	Best Case Assessment Sequence
0.80	1, 3, 5, 8, 9
0.82	1, 3, 5, 8, 9, 2
0.84	1, 10, 3, 5, 8, 7, 9
0.86	1, 10, 4, 3, 5, 8, 7, 9
0.88	1, 10, 4, 3, 5, 8, 7, 9
0.90	1, 10, 4, 6, 3, 5, 8, 7, 2, 9
0.92	1, 10, 4, 6, 3, 5, 8, 7, 2, 9

As there is no way to actually look inside the brain of the learner and check the neural connections to determine whether the learner knows the topic or not, the actual accuracy of predicting the knowledge level of the student with respect to a specific topic depends upon the quality of the assessment activity of the topic and the honesty of the learner's responses. If the quality of the assessment activities is assumed to be perfect the algorithm guarantees accurate adaptation of the course to the learner in the minimum possible time.

As evident from Table 7. Different assessment sequences according to trust thresholds., a trust threshold of 0.90 and above resulted in assessment of all topics in the learner profiling algorithm whereas a threshold of 0.80 and below yields the most minimal sequence of assessment activities for learner profiling. The topics ignored in the various assessment paths generated by the algorithm for the quickest, minimal and accurate profiling of the learner are illustrated in Fig. 7. This shows how an appropriate trust threshold parameter based on the trust on the amount of data available can result in efficient learner profiling.

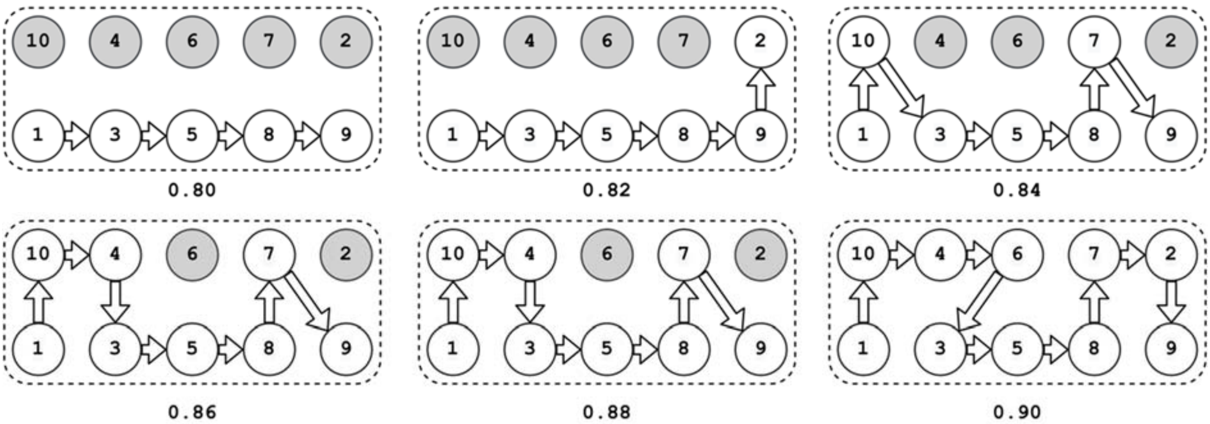

Fig. 7. Minimal profiling paths according to trust threshold

As the algorithm depends upon the responses of the previous learners who took the same course, improving the quantity and the quality of this data improves the overall performance of the algorithm. Since this is designed to be general purpose approach, the accuracy of adaptation of a specific subject to a specific learner depends entirely upon the quality of the assessment activity of a topic.

5. Conflicts of Interest

The authors declare no conflict of interest.

References

[1] A. Januszewski and M. Molenda, *Educational technology : a definition with commentary*. Association for Educational Communications and Technology, 2008.
[2] M. Elsom-Cook, O. University, and M. K. (GB) C. A. L. R. G. O. University, *Intelligent Computer-aided Instruction Research at the Open University*. Open University, 1987. [Online]. Available: https://books.google.co.in/books?id=ptCgHAAACAAJ
[3] R. Freedman, S. S. Ali, and S. McRoy, "What is an Intelligent Tutoring System?," *Int. J. Artif. Intell. Educ.*, vol. 11, no. 3, pp. 15–16, Sep. 2000, doi: 10.1145/350752.350756.
[4] R. Nkambou, R. Mizoguchi, and J. Bourdeau, Eds., *Advances in Intelligent Tutoring Systems*. Berlin Heidelberg: Springer-Verlag, 2010. Accessed: Aug. 08, 2019. [Online]. Available: https://www.springer.com/gp/book/9783642143625
[5] H. S. Nwana, "Intelligent tutoring systems: an overview," *Artif. Intell. Rev.*, vol. 4, no. 4, pp. 251–277, Dec. 1990, doi: 10.1007/BF00168958.
[6] J. R. Anderson and B. J. Reiser, "The LISP tutor: It approaches the effectiveness of a human tutor," *Byte*, vol. 10, no. 4, pp. 159–175, 1985.
[7] J. R. Anderson, "The geometry tutor and skill acquisition," in *Rules of the mind*, Psychology Press, 2014, pp. 175–192.
[8] P. De Boeck, "Random Item IRT Models," *Psychometrika*, vol. 73, no. 4, p. 533, Dec. 2008, doi: 10.1007/s11336-008-9092-x.
[9] H. Cen, K. Koedinger, and B. Junker, "Learning factors analysis - A general method for cognitive model evaluation and improvement," *Lect. Notes Comput. Sci. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinforma.*, vol. 4053 LNCS, pp. 164–175, 2006, doi: 10.1007/11774303_17.
[10] J. E. Beck and J. Mostow, "How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students," *Lect. Notes Comput. Sci. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinforma.*, vol. 5091 LNCS, pp. 353–362, 2008, doi: 10.1007/978-3-540-69132-7-39.
[11] L. R. Rabiner and B. H. Juang, "An Introduction to Hidden Markov Models," *IEEE ASSP Mag.*, vol. 3, no. 1, pp. 4–16, 1986, doi: 10.1109/MASSP.1986.1165342.
[12] A. T. Corbett and J. R. Anderson, "Knowledge Tracing : Modeling the Acquisition of Procedural Knowledge," *User Model. User-Adapt. Interact.*, pp. 253–278, 1995.
[13] M. A. S. Pedro, R. S. J. Baker, and J. D. Gobert, "Incorporating Scaffolding and Tutor Context into Bayesian Knowledge Tracing to Predict Inquiry Skill Acquisition," 2013.
[14] E. A. Bender and S. G. Williamson, *Lists, decisions and graphs*. S. Gill Williamson, 2010.
[15] H. A. Al-Mutawa, J. Dietrich, S. Marsland, and C. McCartin, "On the shape of circular dependencies in java programs," *Proc. Aust. Softw. Eng. Conf. ASWEC*, pp. 48–57, 2014, doi: 10.1109/ASWEC.2014.15.
[16] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation," in *European conference on information retrieval*, 2005, pp. 345–359.

Authors Profile

Sanjay Singh is currently pursuing PhD under UGC-JRF in the Department of Computer Science and Engineering, CDLU, Sirsa. He likes to research software engineering, web development and intelligent tutoring systems.

Prof. Vikram Singh is currently working as a Professor in the Department of Computer Science & Engineering, Chaudhary Devi Lal University, Sirsa, Haryana, INDIA. His research interests include Cyber Security, Network Security, Wireless Sensor Networks, Artificial Intelligence, Surveillance and Privacy, and Big Data Analytics.