IJCSN International Journal of Computer Science and Network, Volume 5, Issue 5, October 2016
ISSN (Online) : 2277-5420 www.IJCSN.org
Impact Factor: 1.02

Artificial Intelligence Approaches in Student Modeling: Half Decade Review (2010-2015)

[1] Salisu Muhammad Sani, [2] Abdullahi Baffa Bichi, [3] Shehu Ayuba

[1] Department of Computer Science, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400, Serdang, Selangor Malaysia.

[2] Department of Computer Science, Faculty of Computer Science and Information Technology, Bayero University Kano, Nigeria.

[3] Department of Computer Science, Faculty of Mathematical and Computer Science, Kano University Science and Technology, Wudil, Kano State, Nigeria.

Abstract - Intelligent Tutoring Systems (ITSs) are special classes of E-learning systems designed using Artificial Intelligence (AI) approaches to provide adaptive and personalized tutoring based on the individuality of students. The student model is an important component of an ITS that provides the base for this personalization. During the course of interaction between student and the ITS, the system observe student's actions and other behavioral properties, create a quantitative representation of these student's attributes called a student model.

Keywords - *Artificial Intelligent Techniques, Intelligent Tutoring Systems, Student Modeling, E-learning Systems.*

1. Introduction

Student modeling is a phenomenon for creating a quantitative representation of the student by the ITS. This process is achieved through acquiring information from the current student using the ITS in order to primarily create his/her profile called a student model (Peña-Ayala, 2012). This model contains information such as the student's knowledge level, learning styles, personal preferences etc. (Drigas, 2009). If designed effectively, the student model can make the ITS to be more adaptive to deal with the uncertainty issues in diagnosing the student. Depending on the domain subject, the student model can be categorized into two parts; domain specific information (DSI) which is dynamic and represent the student's knowledge level in a specific domain and domain independent information (DII) which is static and consist of information like student's personal data, the learning style and motivation measurement (Jeremić, 2009). An important features that make intelligent tutoring systems differ from traditional e-learning systems is their ability observe students actions and draws some useful conclusions from those actions in order to maintain a model of the student (Shute and Psotka, 1996; Grubisic, 2013). The goal of any ITSs is to provide students with interactive assistance aimed at helping them to achieve maximum learning gain, but before an ITS could do so, it needs to finds out what knowledge/skill the student has already acquired and his/her intended goal in moving the knowledge/skill to the next level. In other words, the ITS need to do what is known as assessment and plan recognition on the part of the student. Both of these processes are modeling tasks that involve high level of uncertainty especially in situations where the students are allowed to move through various lines of reasoning without being made to explicitly express those reasoning.

Like most recommender models for non ITSs systems, the student model is a vital component of an intelligent tutoring system that enable the ITS to observe the interactions it has with the students and adapt to their needs. But unlike the non ITSs systems, the goal of an ITS is to ensure that students learn a target instructional objective at the end of a learning session and this also contributes to a great deal of uncertainty to student modeling because it amounts to making an inference out of the student's actions to determine how well the student understood the target domain concepts, this is known as knowledge tracing or assessment (Conati, 2002). However, uncertainty issues in student modeling arise due to factors like insufficient and highly uncertain information received from the student (Conati, 2009). The ITS build the student model based on the observations it makes on the student, interprete the results of those observations into an internal representation called the student model. The student module is considered by many researchers as the heart of the ITSs because both the tutor and the domain modules rely heavily on the student module for both effective pedagogical decision making process and management of domain knowledge concepts.

That is why if the uncertainty issue that characterized the student modeling is not properly handled, the resulting consequence may in turn affect the decisions of all other components of the ITSs that depends on the student model which leads to creating a poorly adaptive teaching and learning environment. Thus, when a student model, which is considered by many researchers in the field of ITSs as the most vital component of the intelligent tutoring systems is so "poor" to the extent that it does not provides clear representation of the students to fully describe them in terms of their characteristics or profiles, then all the decisions of other components of the ITS that depends on the student model such as the tutor or domain models are going to be of poor quality also (Gruibišić, 2012).

2. Student Modeling Approaches

There are quite a number of student modeling approaches in the field of ITSs. However, the following are the most commonly used and are discussed in this literature:

2.1 Bayesian Knowledge Tracing

Bayesian Knowledge Tracing (BKT) was introduced in (Anderson, 1995). The model takes the form of the Hidden Markov Model, where student knowledge is a hidden variable and student performance is an observed variable. The model assumes a causal relationship between student knowledge and student performance; i.e. the correctness of a question is probabilistically determined by student knowledge. There are four parameters estimated by the model: prior knowledge, which is the probability that a particular skill was known by the student before interacting with the tutoring systems; learning rate, which is the probability that student's knowledge transits from unlearned to learned state after each learning opportunity; guess, which is the probability that a student can answer correctly even if he/she does not know the skill required in the problem; slip, which is the probability that a student responds to a question incorrectly even if he/she knows the required skills. The classic Bayesian knowledge tracing has been used broadly and successfully across a range of academic domains and student populations, including elementary reading middle-school mathematics (Gong, Beck, & Heffernan, 2011), middle school science (Pardos, Baker & Gowda, 2013) and college-level genetics (Corbett et al., 2010). However, largely due to the simple model structure and the underlying assumptions the BKT model has, it seems to leave promising opportunities to improve. Another model (Grubišić et al., 2013) uses a combination of Bayesian knowledge tracing and stereotype models to predict student's learning performance. Also (Danaparamita & Lumban Gaol, 2014) is another model that uses BKT and fuzzy logic approaches to models and compare the prediction accuracies of both the Bayesian and fuzzy student models.

One issue with BKT is that it is skill oriented. The assumption is learning differs across skills, but students do not differ as individuals. The outcome is that for a skill, all students share the same BKT parameters, including prior knowledge, learning rate, guess rate and slip rate. Questioning this assumption, researchers have thought about individualization. The initial effort was done in the original work where BKT was introduced (Anderson, 1995). Apparently, the researchers acknowledged that solely skill-oriented knowledge estimation seems an incomplete assumption (Yudelson, Koedinger, & Gordon, 2013). Their solution is to estimate an individualized weight for each student and then adjust the model's generated parameters accordingly. However, the big drawback of this approach is that the optimization can only be conducted off-line, meaning only after all data is obtained a weight can be estimated, and this makes the approach a no run-time solution (Piech et al., 2015).

Another issue with BKT is that its estimated parameters are constant for a skill. This assumption applies that students learn, guess or slip at constant rates. They remain the same regardless of external factors, such as the time spent in learning, the problems practiced, or the mood the student is in, etc. The original intention of such design is made so as to reduce the number of parameters with the focus on refining a cognitive model rather than on evaluating students' knowledge growth (Chi, Koedinger, & VanLehn, 2011), which opposes the goal of student modeling. The third drawback with BKT is its lack of the ability to handle multiple skill problems. A classic BKT model is designed per skill. If a problem requires multiple skills to solve, it raises difficulty deciding to which skill this particular observation should belong.

2.2 Fuzzy Logic

Fuzzy logic approach has been applied successfully to a broad range of problems in different application domains. One such type of domain that is concerned with using fuzzy logic for system design and approximation is student modeling where a fuzzy inference mechanism is used to model students' knowledge states. However, existence of uncertainties and imprecision in student model design makes it difficult to model such problems using expert knowledge only (Almaraashi, 2012). Many researchers therefore, have used fuzzy logic techniques in student modeling. A fuzzy-based student model applied by (Jia, Zhong, & Liu, 2010) to the design an adaptive learning system in order to help students to memory the content and improve their comprehension. Another fuzzy logic representation for student modeling (Goel et al., 2012) that is aimed at facilitating student reasoning based on imprecise information coming from the student–computer interaction and performed the prediction of the degree of error a student makes in the next attempt to a problem. DEPTHS (Jeremić et al., 2012), which is an intelligent tutoring system for learning software design patterns, models the student's mastery and cognitive characteristics through a combination of stereotype and overlay modeling with fuzzy rules that are applied during the learning process to keep student model update. Another student model (Chrysafiadi & Virvou, 2012) integrates fuzzy logic to evaluate the effectiveness and accuracy of the student model of a web-based learning environment. Also (Voskoglou, 2013) uses fuzzy logic approach to model the assessment of students' knowledge and skills. A fuzzy logic based approach (Ajiboye et al., 2013) predicts risk of student's status based on predictive factors.

However, fuzzy logic systems also have their limitations as well. One major issue with the fuzzy logic approach is when designing a simple fuzzy logic system with few inputs, the experts may be able to provide efficient rules but as the complexity of the system grows, the rule base and membership functions become difficult to acquire (Goel, 2012). Moreover, unlike machine learning and neural networks, fuzzy systems lack the capability for pattern recognition which is a serious limitation to this approximate technique. Also, verification and validation of fuzzy knowledge base systems require extensive testing with hardware. And because of their subjective and context dependent nature, exact membership functions and fuzzy rules are very hard to determine and the issue of stability is causing lot of concern for fuzzy control mechanism (Voskoglou, 2013).

2.3 Overlay

The overlay model is a student model which contains the student's knowledge as a subset of the expert or domain knowledge (Figure 2.1) (Gong, 2014). It works on the basis that students will learn the domain and gain knowledge through aspiring to become experts. Knowledge is represented and structured in the same way for both the domain knowledge and the student model, the difference being in terms of completeness. Knowledge representation techniques include rule-based representations and semantic networks. During student modeling, diagnosis takes place by comparing the student's knowledge with the domain knowledge and the difference is explained as the student's lack of skill.

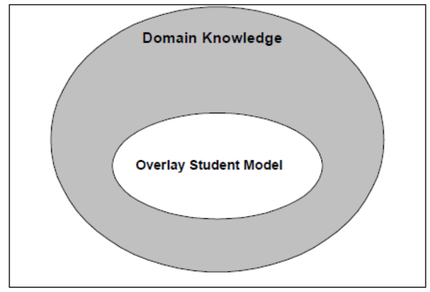

Fig. 1 Overlay Student Model

Examples of overlay models are LS-Plan (Limongelli & Sciarrone, 2011), which is a framework for personalization and adaptation in e-learning, uses a qualitative overlay model., IWT (Albano, 2012) models competence in mathematics in an E-learning environment through an overlay model, which applies an ontology-based representation of the domain knowledge. Also (Mahnane, Laskri, & Trigano, 2012) build an adaptive hypermedia system that integrates thinking style (AHS-TS) by applying an overlay model. Finally, PDinamet (Gaudioso, Montero, & Hernandez-Del-Olmo, 2012) is a web-based adaptive learning system for the teaching of physics in secondary education, which uses an overlay model in order to provide effective and personalized selection of the appropriate learning resources.

This method is incomplete because only the lack of knowledge can be modeled. The main problem with the overlay model is that it assumes that a student's knowledge can be merely a subset of that of an expert, which may not be the case. The domain model is usually represented in terms of atomic units, that is, a student either knows or does not know a certain unit. A student's partial knowledge of a unit cannot be represented. Also, it does not represent any knowledge or beliefs, such as misconceptions, that the student might have that differ from those of the expert. There is no possibility of allowing the student (novice) to have different conceptions of the domain from that of the expert. For example, when categorizing problems, novices tend to rely on surface analogies between problems while experts use deeper functional analogies (Chrysafiadi & Virvou, 2013).

2.4 Differential Model

The differential model is seen as an improvement to the overlay model. It does not assume that gaps in student knowledge are all undesirable. It divides the student's knowledge into two categories: knowledge that the student should know and knowledge the student could not be expected to know (Figure 2.2) (Gong, 2014). Examples of systems which use this approach to student modeling are (Khodeir, Wanas, & Darwish, 2010), an electronic board game to teach arithmetic, and (Guerra, Huang, & Brusilovsky, 2015).

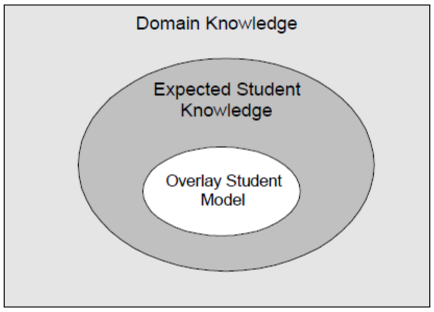

Fig. 2 Differential Student Model

The differential model still suffers from most of the same difficulties as the standard overlay model as it still assumes that the student model is essentially a subset of the expert and the student model remains incomplete.

2.5 Perturbation Model

The perturbation model approach, also called the buggy model, goes beyond inferring what the student knows and does not know about a domain but inferring any faulty knowledge or misconceptions that the student might possess as well. The perturbation student model, which represents the student's correct and faulty knowledge, is considered a subset of both the domain knowledge and buggy knowledge (Figure 2.3). This approach combines the standard overlay model with a representation of faulty or buggy knowledge (Gong, 2014). The domain or expert knowledge is first represented and then augmented with explicit knowledge of possible misconceptions of the student. This explicit knowledge is known as buggy knowledge and allows a more sophisticated diagnosis of the student's state of knowledge than can be accomplished with a simple overlay model. Subsequent remediation goes beyond filling in gaps in the student's knowledge where the tutor must identify and eliminate the student's misconceptions as well as adding the correct conceptions to the understanding of the student.

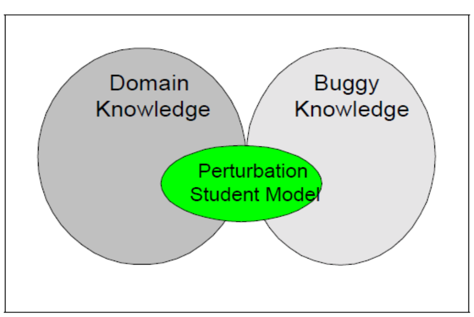

Fig. 3 Perturbation Student Model

Examples of this approach are (Baschera & Gross, 2010), a system that used perturbation student model for spelling training, which represented student's strength and weaknesses, in order to allow for appropriate remediation actions to adapt to students' needs. POMDP (Folsom-Kovarik, Sukthankar, & Schatz, 2013) modeled students' knowledge and misconceptions through an enumerative perturbation student model, which included both correct and incorrect knowledge propositions, in order to provide personalized feedback and support to the distant students in real time. An enumerative perturbation student model was also applied by (Somyürek, 2009) in an intelligent tutoring system that taught basic arithmetic to children (InfoMap). Their perturbation student model, which involved 31 types of addition errors and 51 types of subtraction errors, allowed the reasoning of students' errors and helped the system to expand the explanation during the feedback to the students.

There are many challenges to perturbation technique. For example, there is the problem of bug migration (Chi et al., 2011) which is caused by the change of a bug into a different but related one and this makes the diagnosis of student's actions even more difficult. Also, the construction of bug libraries often involves extensive empirical studies including protocol analysis. The high costs involved could be offset by its portability across student populations in a similar subject domain. However, (Loc Nguyen and Phung Do, 2008) questioned the generality of bug libraries when they conducted a study which showed that the bug library constructed by (Sleeman, 1984) and (Sleeman, 1985) were minimally relevant for the two new student populations. Some researchers have attempted to avoid collecting bugs through empirical observations by automating the generation of buggy knowledge (Martins, Faria, & Carrapatoso, 2008); another good example is the Repair Theory (VanLehn, 2013) which is a generative theory of bugs, that is, a method of deriving bug libraries directly from correct procedures. The usefulness of maintaining bug libraries was also questioned by (Baschera & Gross, 2010).

2.6 Constraint-Based Model

Constraint-Based Model (CBM) features as being computationally simple. It does not require large empirical studies for constructing a bug library, nor an executable expert model or an ideal student model. In CBM technique, no computationally expensive inference algorithm is required, simple pattern matching is used. The domain knowledge is elicited through task analysis and is represented as a set of constraints that capture the central concepts of the domain. The student model is the set of constraints which he violates. These violated constraints become candidates for concepts which the student does not know and is used to guide remediation or feedback. An example of this approach is the InvetionLab (Roll, Aleven, & Koedinger, 2010) which elicited from an expert around five hundred constraints. Another approach for modeling instructional domains, representing students' domain knowledge (Mitrovic & Martin, 2006) based on constraint base modeling was developed to serve as an authoring support for CBM tutors.

A serious limitation of the constraint-based modeling is that it does not prescribe any form of tutorial strategy. It ignores the student's problem solving strategy and is thus able to monitor free exploration and to recognize creative and novel solutions as correct.

2.7 Machine Learning

Student modeling involves a process of making inferences about the student's behavior taking into account the student's knowledge level, cognitive abilities, personal preferences, skills, aptitudes etc. The processes of observation of student's action and behavior in an adaptive or personalized tutoring system should be made automated by the system. A solution for this is machine learning, which is concerned with the formation of models from observations and has been extensively studied for automated induction (Chrysafiadi & Virvou, 2013). Observations of the user's behavior can provide training examples that a machine learning system can use to induce a model designed to predict future actions (Webb, Gallo, Gollwitzer, & Sheeran, 2012).

Another model (Inventado, Legaspi, Suarez, & Numao, 2011) used a combination of Bayesian networks and machine learning technique in order to observe students' reactions while using an intelligent tutoring system and adjust feedback automatically to each individual learner. Similarly, another student model (Baker et al., 2010) based on a combination of Bayesian networks and machine learning technique was also applied. The machine learning constitutes the student model able to assess the probability that a student learned skill at a specific problem step and thus the system can predict the student knowledge. The (Dorça, Lima, & Lopes, 2012) model is a machine learning approach that models student's learning style. Also, a machine learning based model (Centintas, 2010) used machine learning techniques for the performing of the automatic detection of off-task behaviors in intelligent tutoring systems. SimStudent (Matsuda, 2015) used a machine learning technique in order to construct student models automatically and improve the accuracy of prediction of real students learning performance. Finally, (Balakrishnan, 2011) build a student model upon ontology of machine learning strategies in order to model the effect of affect on learning and recognize for any learning task, what learning strategy, or combination thereof, is likely to be the most effective. Machine learning algorithms have become an increasingly important part of our lives. They are integral to all sorts of applications within various technologies.

But the machine learning approaches also have their shortcomings. First, machine learning algorithms are rarely set up to give a reason for a particular decision or output (Armstrong, 2015). This perception of machine learning as an opaque decision–making tool instills a level of mistrust in its outputs. It is important to have clear justifications for a decision; it is not good enough to rely on the supposed quality of an algorithms. This is particularly important because systems that are based on these types of algorithms may be prone to errors. Generally people understandably place more trust in humans than machines but this reluctance to trust these learning systems is a big challenge in realizing their full potential. A machine learning algorithm could fairly easily provide justifications for its decisions.

2.8 Stereotype Model

Another approach of student modeling is stereotyping. The main idea of stereotyping is to cluster all possible users of an adaptive system into several groups according to certain characteristics that they are typically shared. Such groups are called stereotypes. More specifically, a stereotype normally contains the common knowledge about a group of users. A new user will be assigned into a related stereotype if some of his/her characteristics match the ones contained in the stereotype.

The stereotype is a particularly important form of reasoning about users and also student modeling. The approach has been used for student modeling in many adaptive and/or personalized tutoring systems and often in combination with other methods of user modeling. In (Grubiši, Stankov, & Žitko, 2013a), a stereotype and Bayesian networks approach is used to design a system that classifies students in four dimensions according to their learning styles and select the more adequate objects for each student. An adaptive and collaborative learning environment (Conati & Kardan, 2013) that models three aspects of student; expertise level, performance type and personality through a hybrid model based on perturbation and the stereotype-based approaches. Another adaptive tutoring system that uses stereotypes in order to provide an individualized learning environment is CLT (Durrani & Durrani, 2010), which is a C++ tutor. Finally, a stereotype-like approach of student modeling is a software tutor that helps students learn to solve standardized-test type of questions, in particular for a math test called Scholastic

Aptitude Test, and other state-based exams taken at the end of high school in the USA, in order to discern factors that affect student behavior beyond cognition (Arroyo et al., 2014). The advantages of using the stereotype technique are that the knowledge about a particular user will be inferred from related stereotypes as much as possible, without explicitly going through the knowledge elicitation process with each individual user and the information about user group stereotypes can be maintained with low redundancy. However, stereotypes have shortcomings as well. The approach is quite inflexible due to the fact that stereotypes are constructed in a hand-crafted way before real users have interacted with the system and they are not updated until a human does so explicitly (Vieira, 2015). Moreover, (Grubiši, Stankov, & Žitko, 2013b) argues that stereotypes suffer from two problems. First, in order to use them, the set of system users must be divisible into classes; however, such classes may not exist. Second, even if it is possible to identify classes of system users, the system designer must build the stereotypes; this is a process that is both time-consuming and error-prone.

Table 2.1 Common Student Modeling Approaches used in some Researches

Model/Techniq	O/L	STE	PER	BKT	CBM	DM	ML	FL
Arroyo, 2010		✓						
Baker, 2010				✓			✓	
Baschera, 2010			✓					
Cetintas, 2010							✓	
Corbett, 2010				✓				
Durani, 2010								✓
Jia et al, 2010		✓						
Khodeir, 2010						✓		
Mitrovic,2010					✓			
Roll, 2010					✓			
Chi, 2011				✓				
Gong, 2011						✓		
Inventado, 2011				✓			✓	
VanLehn, 2011			✓					
Webb, 2011							✓	
LS-Plan	✓							
Chrysafiadi, 2012								✓
Goel, 2012								✓
IWT, 2012	✓							
Tourtoglou, 2012		✓						
Grubišić, 2013		✓		✓				
San Pedro, 2013				✓				
VanLehn, 2013			✓					
Voskoglou, 2013								✓
Danaparamita, 2014				✓				✓
DEPTH								✓
POMPD			✓					
PDinamet	✓							
Welsa		✓						
Guera						✓		
Danaparamita, 2014				✓				✓
DEPTH								✓
POMP			✓					
PDinamet	✓							
WELSA		✓						

Table 2.2 Existing Student Models

Model	Target Area/Problem Solving	Limitations
Arroyo, 2010	Models student's efforts and behaviors	Selection mechanism within topics made changes in a problem difficult
Baker, 2010	Assessing the probability that a student learned a knowledge concept at a specific problem step	There issue is with the quality of the model's training labels as there is only one way to infer the moment of learning. Also, equations used in this paper are currently based on an unmodified form of BKT
Baschera, 2010	Modeling error in classification and prediction of performance for local and global information	Inputs for parameter estimation are not suitable for long term applications with strongly changing student characteristics
Cetintas, 2010	Detecting students off-task behaviors	Does not explicitly model explicitly the difficult levels of available problems
Corbett, 2010	Evaluating student's learning gain	Doesn't accommodate multiple activities for advanced cognitive tutor curriculum
Durani, 2010	Providing individualized environment for teaching students according to their cognitive abilities	No thorough testing to prove the significance of the learning contents and algorithms. Moreover, learning contents are not designed to cater for other cognitive abilities of students such as working memory, attention, learning, visual thinking etc
Jia et al. 2010	Estimate the learner's knowledge level by test according to learner's target	Model is prone to too much learner's cognitive overload which resulted in difficulty to manage various levels of learning resources
Khodeir, 2010	Estimating differential student knowledge model in a probabilistic domain	Doesn't perform coarse update especially when the original match is limited which is essential for reflecting weak students.

Table 2.2 Contd.

Model	Target Area/Problem Solving	Limitations
Somyürek, 2010	Models and analyze students' information collection, construction and updating	Weak performance problem solving
Welsa, 2010	Adapting courses to the learning preferences of each student	Lack of advanced communication and collaboration. Also, the adaptation component doesn't integrate wide variety of actions
Chi, 2011	Modeling student's performance	Performance is heavily dependent upon the specific prediction task being performed and also the model lacks parameter fitting measure to determine the best fit
Gong, 2011	Comparing optimizing knowledge tracing and performance factor analysis	Doesn't effectively Handle items with multiple skills and the model ignores negative learning rates
Inventado, 2011	Offering adaptive support in relatively unconstrained learning environment	This model has limitations of inconclusive result as different methods always gives the same result even when applied on the same data
VanLehn, 2011	Modeling student's performance	Performance is heavily dependent upon the specific prediction task being performed and the model doesn't provides parameter fitting measures to determine the best fit
LS-Plan	Modeling the learner, and adapting their learning experience	The model lack any form of personalization which affect the automated sequencing of the course content for each student

Table 2.2 Contd.

Model	Target Area/Problem Solving	Limitations
IWT	Modeling competence in mathematics in an e-learning environment	Model supplements are based on multi-graphs that reflects mainly mathematics domain
PDinamet	Predictive models for teaching Physics in secondary education	Inadequate tools that provide a more seamless integration of the predictive models
Goel, 2012	Removing the arbitrary specification of precise numbers and facilitates modeling at a higher level of abstraction	Due to over-fitting issues and a lack of precision, the performance of the model in terms of accuracy was very low
DEPTH	Evaluation of a student model	of context-aware educational services to support collaborative learning contributes to low efficiency Lack
Chrysafiadi, 2012	Evaluating the effectiveness and accuracy of the student model	The occurrence of mean scores in the two groups of students is not real but by chance. Thus the results obtained do not represents a real difference between the two populations
Grubišić, 2013	Testing the successfulness of student knowledge prediction	Very low performance with 36% as the best prediction accuracy of the model.
POMPD	Model learner features such as mastery of individual skills or the presence of specific misconceptions	The model does not provide real-time evaluation of its efficacy with human trainees. And there is high performance degradation that resulted in information loss and state compression
San Pedro, 2013	Model student affect in a web-based tutoring platform and learning outcomes	No real time integration of affect detection into a teacher's tutor dashboard that can make an affective state constructive

Table 2.2 Contd.

Model	Target Area/Problem Solving	Limitations
VanLehn, 2013	Developing new learning activities involving model construction	Time required for students to become fluent in model construction is too high and consequently affects students' understanding of systems and domains
Voskoglou, 2013	Assessing student groups' knowledge and skills	The model lacks possible extension to various instructional domains
Danaparamita, 2014	Comparing the accuracy of student model developed with Bayesian Network and Fuzzy	The membership function dependency based inference process is independent of any

	Logic in predicting student knowledge level	form of fuzzy outputs from the fuzzifier which violets the real principle of a true fuzzy inference process. Moreover, the overreliance of the only two valued variables known or not known (true or false) as contain in this model's Bayesian distribution has created undefined intervals that consequently resulted in leaving any input space within the undefined interval untreated
Guerra, 2015	Exploring the open student model effects beyond comparison, particularly metacognitive support	System doesn't allow students to change their knowledge levels through selected assessment items

3. Conclusion

The results of the findings for the student modeling approaches and the various existing works are presented in tables 2.1 and 2.2 respectively. To be more specific, table 2.1 presents the student modeling approaches that have been used in a variety of adaptive and/or personalized tutoring systems. Table 2.2 presents a number of existing student models, the approaches that have been used in their modeling as well as the numerous limitations that characterized each model. From the result in table 2.1, it can be observed that the most common used student modeling techniques within the period of the review are the stereotype, Fuzzy logic and Bayesian approaches. The review of various student modeling approaches and the existing student models focuses mainly within a five year period (2010–2015) in order to arrive at getting the more recent trends in these directions. The year 2010 recorded the highest number of research works within the period under consideration. In addition, it can also be seen that many researchers have used a hybrid student model, which brings together various features of different techniques of student modeling, in order to combine various aspects of student's characteristics. For instance, there are hybrid student models that combine overlay with stereotype modeling techniques, or stereotypes with machine learning techniques, or an overlay student model with Bayesian networks techniques, or Bayesian networks with machine learning algorithms. The above combinations of student modeling techniques are just some examples.

References

[1] M. Almaraashi. "Learning of Type-2 Fuzzy Logic Systems using Simulated Annealing". 2012, PhD Theisis.

[2] A. T. Coerbett and J.R. Anderson "Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge", Kluwer Academic Publishers, Vol 4, 1995, pp. 253–278.

[3] H. Armstrong "Machine Learning Capabilities and Limitations" Nesta, 2015, www.nesta.org.uk.

[4] I. Arroyo, B. P. Woolf, W. Burelson, K. Muldner, D. Rai, and M. Tai. "A multimedia adaptive tutoring system for mathematics that addresses cognition, metacognition and affect". International Journal of Artificial Intelligence in Education, Vol 24, No. 4, 2014, pp. 387–426.

[5] Baker. R.S, Goldstein. A.B and Heffernan. N.T. "Detecting the Moment of Learning" International Journal of Artificial Intelligence in Education, IOS Press and the authors, 2010.

[6] Balakrishnan, A. "Multi-disciplinary Trends in Artificial Intelligence". In Conference: Multi-disciplinary Trends in Artificial Intelligence - *5th International Workshop*, MIWAI 2011, Hyderabad, India, December 7-9, 2011.

[7] Baschera, G. M and Gross, M. "Poisson-based inference for perturbation models in adaptive spelling training" International Journal of Artificial Intelligence in Education, 2010, Vol 20, No. 4, pp. 1–31.

[8] Carmona, C., Castillo, G., & Millan, E. "Designing a Dynamic Bayesian Network for Modeling Students' Learning Styles". *Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 346–350.

[9] Centintas, S. "Automatic Detection of Off-Task Behaviors in Intelligent Tutoring System with Machine Learning Techniques". IEEE Transaction on Learning Technologies, 2010, Vol. 7, pp. 1–9.

[10] Chi, M., Koedinger, K. R., Gordon, G., Jordan, P. W., & VanLehn, K. "Instructional Factors Analysis: A Cognitive Model For Multiple Instructional Interventions". *In Proceedings of the 4th International Conference on Educational Data Mining, Eindhoven, The Netherlands*, July 6-8, 201, pp. 61–70.

[11] Chrysafiadi, K and Virvou, M. "Evaluating the integration of fuzzy logic into the student model of a web-based learning environment". Expert Systems with Applications, 2012, Vol. 39,No.18, pp. 13127–13134.

[12] Chrysafiadi, K., & Virvou, M. "Student modeling approaches: A literature review for the last decade". Expert Systems with Applications, 2013, Vol. 40, No.11, pp. 4715–4729.

[13] Conati C, Gartener A, VanLehn K. "Using Bayesian Networks to Manage Uncertainty in Student Modeling" User Modeling and User-Adapted Interaction, Vol. 12, pp. 371- 417.

[14] Conati, C. "Bayesian student modeling". Studies in Computational Intelligence, 2009, Vol. 308, No.1, pp.281–299.

[15] Conati, C. and Kardan, S. “Student modeling: Supporting personalized instruction, from problem solving to exploratory open-ended activities”. AI Magazine, 2013, Vol.34, No.3, pp.13–26.

[16] Corbett, A, Kauffman, L, Maclaren, B, Wagner, A, and Jones, E. ”A Cognitive Tutor for Genetics Problem Solving: Learning Gains and Student Modeling” Journal of Educational Computing Research, 2011, Vol. 42, No.2, pp.219–239.

[17] Danaparamita, M and Gaol, L. F. “Comparing student model accuracy with bayesian network and fuzzy logic in predicting student knowledge level”. International Journal of Multimedia and Ubiquitous Engineering, 2014, Vol. 9, No. 4, pp.109–120.

[18] Dorça, F. A, Lima, L. V, Fernandes, M. A and Lopes, C. R. “A Stochastic Approach for Automatic and Dynamic Modeling of Students’ Learning Styles in Adaptive Educational Systems”. Informatics in Education, 2012, Vol. 11, No. 2, pp.191–212.

[19] Drigas, A. S., Argyri, K., & Vrettaros, J. “Decade Review (1999-2009): Artificial Intelligence Techniques in Student Modeling”. Knowledge, Learning, Development and Technology for All, 2009, Vol. 49, pp.552–564.

[20] Durrani, S, and Durrani, Q. S. “Intelligent Tutoring Systems and cognitive abilities”, 2010, Vol. 12, No. 3, pp.1–10.

[21] Folsom-Kovarik, J. T., Sukthankar, G., & Schatz, S. “Tractable POMDP representations for intelligent tutoring systems”. ACM Transactions on Intelligent Systems and Technology, 2013, Vol. 4, No. 2, pp.1–22.

[22] Goel, G., Lallé, S., & Luengo, V. “Fuzzy logic representation for student modelling: Case study on geometry”. In 11th International Conference on Intelligent Tutoring Systems, Greece,2012, Vol. 7315 LNCS, pp. 1–6.

[23] Gong, Y. ‘Student Modeling in Intelligent Tutoring Systems”. 2014, PhD Thesis.

[24] Gong, Y, Beck, J. E and Heffernan, N. T. “How to construct more accurate student models: Comparing and optimizing knowledge tracing and performance factor analysis” International Journal of Artificial Intelligence in Education, 2011,Vol. 21,No. 1-2, pp.27–46.

[25] Grubiši, A. “ADAPTIVE STUDENT’S KNOWLEDGE ACQUISITION MODEL IN E-LEARNING SYSTEMS”. 2012, PhD Thesis.

[26] Grubiši, A, Stankov, S and Žitko, B. “Stereotype Student Model for an Adaptive e- Learning System” International Journal of Computer, Electrical, Automation, Control and Information Engineering, 2013, Vol. 7, No. 4, pp.440–447.

[27] Inventado, P. S., Legaspi, R., Suarez, M., & Numao, M. “Predicting Student Emotions Resulting From Appraisal of Its Feedback” Research and Practice in Technology Enhanced Learning, 2011, Vol. 6, No. 2, pp.107–133.

[28] Jeremić, Z, Jovanović, J and Gašević, D. “Student modeling and assessment in intelligent tutoring of software patterns” Expert Systems with Applications, 2012,Vol. 39, No. 1, pp.210–222.

[29] Ajiboye, A. R, Arshah, R. A and Qin, H. “Risk Status Prediction and Modelling Of Students’ Academic Achievement - A Fuzzy Logic Approach”. International Journal Of Engineering And Science, 2013, Vol. 3, No. 11, pp.7–14.

[30] Albano, G. “A knowledge-skill-competencies e-learning model in mathematics “ En Un Entorno Virtual, 2012, Vol. 9, No. 1, pp.306–319.

[31] Jia, B., Zhong, S, Zheng, T and Liu, Z. “The Study and Design of Adaptive Learning System Based on Fuzzy Set Theory”. In Transactions on edutainment IV, 2010, pp. 1–11.

[32] Limongelli, C and Sciarrone, F, “Personalized e-learning in Moodle: the Moodle_LS System”. Journal of E-Learning, 2011, Vol. 7, No. 1, pp. 49–58.