Community Models to Enhance Adaptive Instruction

Robert Sottilare

U.S. Army Research Laboratory, Orlando, FL USA

robert.a.sottilare.civ@mail.mil

Abstract. This paper discusses the need and methods to develop community-based persona (learner models) to tie together key learner attributes and learning outcomes (e.g., knowledge acquisition) with the goal of facilitating the validation of adaptive instructional strategies and tactics. Adaptive instruction, sometimes referred to as differentiated instruction, is a learning experience tailored to the needs and preferences of each individual learner or team in which strategies (recommendations and plans for action) and tactics (actions by the tutor) are selected with the aim of optimizing learning, performance, retention, and the transfer of skills between the instructional environment (usually provided by an Intelligent Tutoring System or ITS) and the work or operational environment where the skills learned will be applied. Adaptive instructional systems (AISs) use human variability and other learner attributes along with instructional conditions to select appropriate strategies and tactics. This is usually accomplished through the use of machine learning techniques, but large amounts of data are needed to reinforce the learning of these algorithms over time. We propose a method to develop community models more quickly by enabling diverse groups to contribute the results of their experiments and training data in a common instructional domain to a cloud-based model.

Keywords: Adaptive Instructional Systems (AISs), Intelligent Tutoring Systems (ITSs), Learner Modeling, Reinforcement Learning

1 Introduction

Adaptive instructional systems (AISs) provide machine-based instruction through technologies like Intelligent Tutoring Systems (ITSs) which interact with learners and make decisions about interventions based on the needs and preferences of each individual learner [1]. These interventions are based on a model of that learner or team and the conditions in the instructional environment. A simple model of instruction includes instructional elements (Figure 1) to be considered during the AIS authoring process [2]. This model usually includes critical information about the learner and the instructional domain that informs a machine learning algorithm in the tutor and that algorithm is trained by consuming data involving both successful and unsuccessful decisions.

adfa, p. 1, 2011.

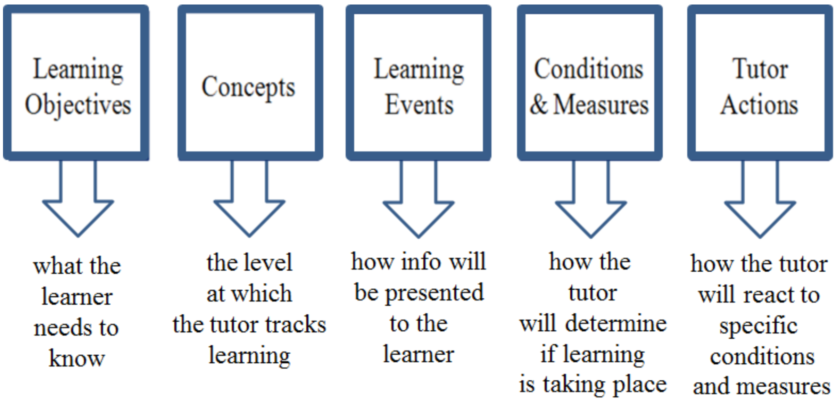

Figure 1. Elements of an Adaptive Instructional Model

A basic instructional model (Figure 2) involves learner actions and conditions, environmental conditions, instructional policies, and interactions (actions, observations, and assessments) capturing data between the tutor, the environment, and the learner.

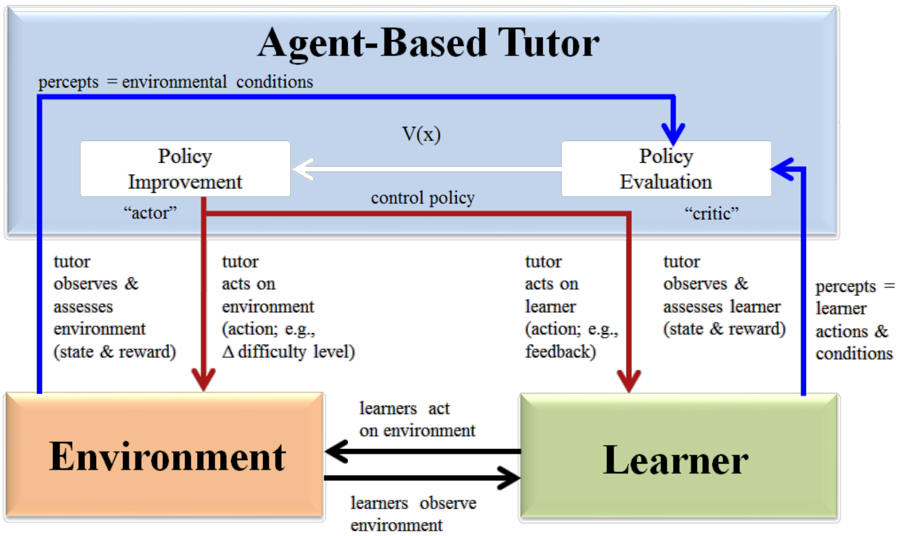

Figure 2. Interaction within an Instructional Model

The instructional model and its policies can be improved over time through a class of machine learning techniques called reinforcement learning (RL) algorithms. RL algorithms are often used to improve the accuracy and reliability of adaptive instructional decisions. However, this method requires usually large amounts of data to develop optimally effective instructional policies that drive tutor strategies and tactics. To reduce the development time of instructional policies, we advocate a mechanism to collaboratively develop instructional models for a variety of task domains. Before we discuss community models, it is appropriate to review how reinforcement learning works in practice.

2 Reinforcement Learning

Conditional systems, like AISs, are used to determine tutor behavior or more specifically their decisions and interactions with the learner and the environment. These interactions are in the form of instructional strategies and tactics in systems like the Generalized Intelligent Framework for Tutoring (GIFT) [3, 4]. AISs consider instructional strategies and tactics which are bounded by constraints posed by policies and these policies are often framed in terms of Markov Decision Processes (MDPs) [5] which seek to reinforce maximum outcomes or rewards over time [6, 7]. To select an optimal policy or modify it based on new information, the MDP considers the current state and the value of any successor states with respect to a desired outcome.

Per Mitchell [8], the optimal action, a, for a given state, s, is any action that maximizes an immediate reward, r (s, a) and the value, V, of the immediate successor state, s'. What does this means for AISs? First, based on our model in Figure 2, states are much more complex in AISs than in most systems. States must capture conditions of the learner (e.g., performance, affect, competence) and the instructional environment (e.g., concept under instruction, concept map (hierarchical relationships between learning objectives), and recent content presented) that affect the learning experience. Actions, referred to as tactics in GIFT, are the set of instructional options available in the current state. The reward is performance and the value is the anticipated performance in the next state. It is easy to see that the number of possible states can be very large in AISs and that the task of validating MDPs for all possible states could take a single organization a very long time. Hence the need for a process to divide the validation process into smaller discrete elements that can be processed by researchers in parallel, but to a similar standard.

3 An Approach to the Development of a Community Model

Based on our goal to reduce the time to validate a complex instructional model, let's simplify our model for adaptive instruction by dividing it into its three essential elements: the learner model, the instructional environment, and the instructor or tutor. The learner model consists of the attitudes and behaviors along with the cognitive states of the learner. The instructional environment consists of learning objectives (LOs; also known as concepts), a concept map (a hierarchical relationship of concepts to be learned), a set of learning activities which include content and directions on how the learner will interact with the content, a set of measures to determine learning and performance, and a set of available tutor strategies and tactics to respond to various learner attitudes, behaviors and cognitive states. The tutor consists of a set policies that drive its behavior and interaction with the learner. The goal is for the policies to be updated regularly as the tutor interacts with more and more learners and finds new highs to override previous best practices.

According to Chi and Wylie [9], learner activities vary from least effective to most effective are: passive (receiving), active (manipulating), constructive (generating), and interactive (dialoguing). As activities are selected and presented to the user, the tutor uses measures to assess progress toward learning and performance. For example, in a tutor that instructs learners to read, the tutor might engage the learner in a reflective dialogue (interactive activity) about a recently read passage to ascertain the learner's comprehension of the concepts presented in that specific reading.

To further our approach, we might consider generalizing terms and measures (Table 1) in lieu of using specific measures. Reducing the number of discrete states also reduces the matrix for selecting the best possible response by the tutor to existing conditions.

Table 1. Instructional Model Element Descriptors

Instructional Model Elements	Independent Variables	Dependent Variables	Variables of Interest	Nominal Descriptor
Learner Model				
Attitudes			X	Pos - Neg
Behaviors			X	Behavioral Marker
Cognitive States			X	H-M-L Workload
Instructional Environment				
Concepts (Learning Objectives)			X	LO Description
Concept Map (including prerequisites)				
Learning Activities (ICAP - interactive, constructive, active, and passive)			X	ICAP
Content & Interaction				
Directions & Support				
Measures of Learning & Performance (Desired Outcomes)		X		H-M-L
Instructional Policies	X			Policy Name
Tutor Strategies & Tactics				

4 Discussion

By coming to consensus on a common set of terms and defining their relationships in an ontology [10], we might realize the degree of interoperability needed to develop community-based models for AISs. However, we also realize that their complexity [11] and lack of interoperability [12] between various AISs may slow the progress of developing these models. The good news is that current events highlight significant opportunities to capture and share the learner data needed to grow community learner models.

Recently, the Institute of Electrical and Electronics Engineers (IEEE) Learning Technologies Steering Committee approved a study group to examine opportunities for standards to promote interoperability and reuse with this class of technologies known as AISs. If successful, this initiative will likely result in a high degree of sharing among AIS components, tools, methods, and data.

In 2017, under the auspices of the North Atlantic Treaty Organization's Human Factors and Medicine Panel (NATO-HFM), a research task group examining technologies and opportunities to exploit Intelligent Tutoring Systems for adaptive instruction completed its task and recommended "the development of standard learner model attributes which include both domain-independent (e.g., demographics) and domain-dependent (e.g., domain competency, past performance and achievements) fields which are populated from a learner record store (LRS) or long-term learner model. This will promote standard methods to populate real-time models during ITS-based learning experiences and allow for common open learner modeling approaches and transfer of competency models from one tutor to another" [13]. If adopted, this recommendation may be an impetus in creating a large, diverse community from which data for community learner models could be harvested.

While not a recent phenomenon, the advent of the educational data mining repositories DataShop [14] and its successor, LearnSphere [15] provide mechanisms for contributing and consuming experiment data related to learners interacting with instructional systems like AISs.

5 References

1. Sottilare, R. (*in press*). A Comprehensive Review of Design Goals and Emerging Solutions for Adaptive Instructional Systems. *Technology, Instruction, Cognition and Learning (TICL)*, Old City Publishing, Inc., Philadelphia, PA.
2. Sottilare, R.A. (2016). Adaptive Instruction for Individual Learners within the Generalized Intelligent Framework for Tutoring (GIFT). In *Foundations of Augmented Cognition* (pp. 90-96). Springer International Publishing.
3. Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf
4. Sottilare, R., Brawner, K., Sinatra, A. & Johnston, J. (2017). An Updated Concept for a Generalized Intelligent Framework for Tutoring (GIFT). Orlando, FL: US Army Research Laboratory. May 2017. DOI: 10.13140/RG.2.2.12941.54244.
5. Cassandra, A. R. (1998). Exact and approximate algorithms for partially observable Markov decision processes.
6. Nye, B., Sottilare, R., Ragusa, C., & Hoffman, M. (2014). Defining Instructional Challenges, Strategies, and Tactics for Adaptive Intelligent Tutoring Systems. In R. Sottilare, A. Graesser, X. Hu, & B. Goldberg (Eds.) *Design Recommendations for Intelligent Tutoring Systems: Volume 2 - Instructional Management*. Army Research Laboratory, Orlando, Florida. ISBN: 978-0-9893923-2-7.
7. Puterman, M. L. (2009). Markov decision processes: discrete stochastic dynamic programming (Vol. 414). Hoboken, NY: John Wiley and Sons.
8. Mitchell, T.M. (1997). Machine Learning. Boston, Massachusetts: McGraw-Hill. ISBN: 978-0-07-042807-2.
9. Chi, M.T.H. & Wylie, R. (2014) The ICAP Framework: Linking Cognitive Engagement to Active Learning Outcomes, Educational Psychologist, 49:4, 219-243, DOI: 10.1080/00461520.2014.965823
10. Bechhofer, S. (2009). OWL: Web ontology language. In *Encyclopedia of database systems* (pp. 2008-2009). Springer US.

11. Sottilare, R.A. & Ososky, S. (2017, July). Defining Complexity in the Authoring Process for Adaptive Instruction. In *Foundations of Augmented Cognition* (pp. 237-249). Springer International Publishing.
12. Wegner, P. (1996). Interoperability. ACM Computing Surveys (CSUR), 28(1), 285-287.
13. Sottilare, R.A. (2018). Chapter 8 – Summary and Recommendations for the Exploitation of ITS Technologies in NATO. NATO Final Report of the Human Factors & Medicine Research Task Group (HFM-RTG-237), Assessment of Intelligent Tutoring System Technologies and Opportunities. NATO Science & Technology Organization. DOI: 10.14339/STO-TR-HFM-237. ISBN 978-92-837-2091-1.
14. Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of educational data mining*, *43*.
15. Stamper, J., Koedinger, K., Pavlik Jr, P. I., Rose, C., Liu, R., Eagle, M., & Veeramachaneni, K. (2016). Educational Data Analysis using LearnSphere Workshop. In *Proceedings of the EDM 2016 Workshops and Tutorials co-located with the 9th International Conference on Educational Data Mining. Raleigh, NC. Workshop*.