Toward "Hyper-Personalized" Cognitive Tutors

Non-Cognitive Personalization in the Generalized Intelligent Framework for Tutoring

Stephen E. Fancsali[1], Steven Ritter[1], John Stamper[2], Tristan Nixon[1]

[1]*Carnegie Learning, Inc.*
437 Grant Street, Suite 918
Pittsburgh, PA 15219, USA
{sfancsali, sritter, tnixon}@carnegielearning.com

[2]*Human-Computer Interaction Institute*
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213, USA
john@stamper.org

Abstract. We are starting to integrate Carnegie Learning's Cognitive Tutor (CT) into the Army Research Laboratory's Generalized Intelligent Framework for Tutoring (GIFT), with the aim of extending the tutoring systems to understand the impact of integrating non-cognitive factors into our tutoring. As part of this integration, we focus on ways in which non-cognitive factors can be assessed, measured, and/or "detected." This research provides the groundwork for an Office of the Secretary of Defense (OSD) Advanced Distributed Learning (ADL)-funded project on developing a "Hyper-Personalized" Intelligent Tutor (HPIT). We discuss the integration of the HPIT project with GIFT, highlighting several important questions that such integration raises for the GIFT architecture and explore several possible resolutions.

Keywords: Cognitive Tutors, intelligent tutoring systems, student modeling, affect, personalization, non-cognitive factors, gaming the system, off-task behavior, Generalized Intelligent Framework for Tutoring, GIFT

1 Introduction

Our goal in developing a "Hyper-Personalized" Intelligent Tutor (HPIT) is to bring learning systems to the next level of user/student adaptation. In addition to traditional features of systems like Carnegie Learning's Cognitive Tutor (CT), HPIT includes non-cognitive factors to provide a more personalized experience for users of the system. In this paper, we discuss features of HPIT and situate the work in the context of the Generalized Intelligent Framework for Tutoring (GIFT) architecture.

1.1 Cognitive Tutors

Carnegie Learning's Cognitive Tutor (CT) [1] is an adaptive, computer-based tutoring system (CBTS) or intelligent tutoring system (ITS) based on the Adaptive Control of Thought—Rational (ACT-R) theory of cognition [2] used every year by hundreds of thousands of learners, from middle school students through college undergraduates. To date, Carnegie Learning's development of the CT has focused primarily on mathematics.

1.2 Generalized Intelligent Framework for Tutoring (GIFT)

The Army Research Laboratory (ARL) is working to develop the Generalized Intelligent Framework for Tutoring (GIFT). The GIFT project aims to provide a "modular CBTS framework and standards [that] could enhance reuse, support authoring and optimization of CBTS strategies for learning, and lower the cost and skillset needed for users to adopt CBTS solutions for military training and education" [3]. Given substantial efforts in both academia and industry to develop ITSs, integrating aspects of this work with ARL's GIFT is important for future development. We briefly provide an overview of GIFT before describing a particular project that will integrate architecture for "hyper-personalized" versions of ITSs, like Carnegie Learning's CT, with GIFT.

GIFT provides a modular framework to achieve and support three goals or "constructs" [3]: (1) affordable, easy authoring of CBTS components, (2) instructional management for integrating pedagogical best practices, and (3) experimental analysis of effectiveness.

GIFT's service-oriented architecture (SOA) currently provides four modules, among other functional elements, around which CBTSs can be implemented and into which existing ITSs can be integrated. Three modules are domain-independent: the *Sensor Module*, *User Module*, and *Pedagogical Module*. The *Domain Module* contains all domain-specific content, including problems sets, hints, misconceptions, etc.

One functional element outside of "local tutoring processes" in the GIFT architecture is important for the present discussion: *Persistent Learner Models*. These models are intended to "maintain a long term view of the learner's states, traits, demographics, preference, and historical data (e.g., survey results, performance, competencies)" [3]. As we review several key, non-cognitive factors upon which we seek to base a "hyper-personalized" CT, the importance of data intended to be tracked by *Persistent Learner Models* will be clear. However, the notion of "persistence" for this data becomes less clear.

2 Non-Cognitive Factors

While the CT and other ITSs adapt content presented to students based on cognitive factors such as skill mastery, there are many other (cognitive and non-cognitive) factors for which the student learning experience might be adapted and personalized. We present several examples of recent research focusing on the impact of non-cognitive factors on student learning in ITSs.

2.1 Gaming the System and Off-Task Behavior

A wide variety of behaviors in an ITS or CBTS like CT may be associated with learning. Two specific behaviors that have been widely studied in the recent literature include "gaming the system" behavior and off-task behavior [4] [5]. This research has not only studied the association of these behaviors with learning but has also led to the development of software "detectors" of such behavior from ITS log data.

Sometimes students attempt to advance through material in ITSs like the CT without actually learning the content and developing mastery of appropriate skills. Such behavior is generally referred to as "gaming the system." Examples of such behavior include rapid or systematic guessing and "hint-abuse." "Hint-abuse" refers to repeated student hint requests, sometimes until a final or "bottom-out" hint (essentially) provides the answer to a problem or problem step [10].

Software "detectors" of gaming the system behavior have been developed (e.g., [7]) and correlated with field-observations of student behavior. Such software detectors rely on various features that are "distilled" from CT log files [8]. Studies find an association [4] [9] [10] and evidence for a causal link [11] [12] between gaming the system behavior and decreased student learning. Similar software has also been developed, and validated via field-observations, to detect off-task behavior [5].

Other types of behavior, of course, may also be important for learning in CBTSs and ITSs. While some behaviors may be "sensed" via physical, tactile, and/or physiological sensors, we emphasize that state-of-the-art research attempts to detect different types of behavior from logs generated by CBTSs and ITSs.

2.2 Affect

Building on success in developing detectors of student behavior, current research seeks to detect student affect (e.g., boredom, engaged concentration, frustration, etc.) without sensors (i.e., without physical, tactile, and/or physiological sensors) [13]. Such detectors have also been validated by field-observations of students using ITS in the classroom. Further, these detectors have been successfully deployed to predict student learning via standardized test scores [14].

While student affect and behavior might also be physically "sensed", inferred, or measured via survey instruments (e.g., mood via survey [15]), data-driven detection of student affect and behavior is a promising approach to achieve the GIFT design goal of supporting "low-cost, unobtrusive (passive) methods… to inform models to classify (in near real-time) the learner's states (e.g., cognitive and affective)" [3].

2.3 Preferences

Carnegie Learning's middle school mathematics CT product, MATHia, allows students to set preferences for various interest areas (e.g., sports, art) and probabilistically tailors problem presentation based on those preferences. On-going research aims to determine if and how presenting students with problems related to their preference areas is associated with engagement and benefits student learning (e.g., [16-17]). Other student preferences might be ascertained via surveys, configuration settings, or inferred from data, at different levels of granularity and time scales.

2.4 Personality and Other Learner Characteristics

Other characteristics of learners may prove important for learning. We consider two prominent examples that are being considered as we develop HPIT. Investigating other learner characteristics is also a topic for future research.

Grit.
Grit [18-19] is defined as the tendency to persist in tasks over the long term, when reaching the goal is far off in the future. Duckworth et al. [18] found that grit, measured by a survey instrument [19], predicted retention among cadets at the United States Military Academy at West Point, educational attainment among adults, and advancement to the final round among contestants in the Scripps National Spelling Bee.

Educational environments like CT are able to adjust the rate at which difficulty of activities increases. Students high in grit may, for example, benefit more from rapid increases in the difficulty of course material compared to students low in grit, regardless of knowledge-levels.

Motivation and Goals.
Students' motivation and goals are likely to be important for learner adaptation. Recent research [20] considers fine-grained assessment, via frequent surveys (occurring every few minutes) embedded within CT, of student motivation and goal orientation to better understand models self-regulated learning. Elliot's framework for achievement goals provides for two dimensions, definition (mastery vs. performance) and valence (approach vs. avoidance), along which goals are oriented [20-22].

Particular problems or hints (or ways of providing hints) might, for example, be best suited to students with a mastery avoidance goal orientation that seek to avoid failure, and so on. In addition to ascertaining the influence of goals and motivation on learning, determining whether students' motivation and goals (at various levels of granularity) are relatively static or dynamic through a course, and possibly influenced by students' experience in a course, remains a topic of active research [20].

3 Hyper-Personalizing Cognitive Tutors

One particularly important aspect of CTs from an architectural perspective is that they are driven by user inputs (called "transactions" [23]). From a system perspective, an update to the learner model happens only when the student takes some action within the system (e.g., attempting to answer a question or asking for a hint). Other student-initiated inputs might include, for example, student ratings of whether particular problems are interesting (e.g., an ever-present 5-star ranking system attached to each problem). Student-initiated inputs range in time from the nearly continuous to being separated by significant amounts of time.

In a more general system like GIFT, updates to the student model happen, not only at different timescales, but can also be initiated by actors (or factors) other than the learner. Examples include: acquiring data to update the student model through surveys given to the student at times determined by the system (e.g., only at course-beginning and end vs. periodically between problems or units), through real-time sensors (e.g., an eye-tracker), through student-determined inputs, etc. Furthermore, in some learning environments, the student model might be updated by factors linked to the passage of time (e.g., inferring that a skill has been "forgotten" because the student does not use a tutor for a substantial amount of time or updating students' knowledge state after a chemical reaction occurs following some time-lapse in a simulation-based chemistry tutor). The mode and frequency of data collection, in part, determine the kinds of pedagogical moves that the ITS can take.

The ADL-funded Hyper-Personalized Intelligent Tutor (HPIT) project seeks to develop a modular, plug-in-like architecture using various data collection and processing elements to inform CT's provision of problems, feedback, hints, etc. Each factor (whether cognitive or non-cognitive) may contribute to varying degrees to the decision-making process, as data are collected and inferences drawn about learner "state." A plug-in architecture allows for "voting" schemes to drive the personalization process (e.g., perhaps two non-cognitive factors and one cognitive factor are all equally weighted, or not). Methods will be developed to resolve conflicts (i.e., break "ties") when multiple recommendations are appropriate given a student's "state."

While cognitive factors are crucial for adapting educational content for disparate users, HPIT's primary innovation is the creation of a platform and framework for adapting content based on non-cognitive factors. To do so, HPIT will draw on data from software detectors, surveys, and possibly physical sensors. Perhaps more important from an architectural perspective, however, is the fact that the measurement, inference, or assessment of various cognitive and non-cognitive factors may occur on different time scales and at different levels of granularity.

For example, if a student is both bored (as, for example, inferred from a software detector applied to real-time log data) and uninterested in material currently being presented (as inferred from survey results), material similar in difficulty, but providing examples better suited to student preferences, might be presented. However, a different strategy might be required if we lack data about their interests. Adapting pedagogical strategies based on data that is currently available is a virtue of the flexibility of the HPIT architecture we are developing.

4 Implications for GIFT Architecture

The GIFT architecture and recent research (e.g., [15]) focuses on using physical sensors and surveys to gather information about a learner's non-cognitive state. The HPIT framework builds on work to infer/measure student state with surveys and software detectors that use data from tutor logs. These software detectors rely on data generated by the ITS following student-initiated input to the ITS. We discuss several implications for the GIFT architecture and the integration of existing ITSs into GIFT.

4.1 Surveys

In GIFT, *Persistent Learner Models* store survey results and communicate with the *User/Learner Module* via the SOA. However, HPIT requires that surveys be deployable at nearly any point in the learning experience, rather than simply before and after a "chunk" (e.g., unit) of course material. Furthermore, surveys/polls might be conducted that assess momentary characteristics of the student experience, rather than the persistent state of a student[6].

Some survey-like elements may be deployed nearly continuously. Thus, it might be initially attractive to conceptualize surveys are as a particular type of sensor. However, the processing of the type of survey data we have in mind seems fundamentally different than processing sensor data (e.g., an eye-tracker). Consider, for example, the previously noted five-star rating system for problems. While the rating system may be deployed for near-continuous collection of data, frequently students may not choose to rate many problems. Perhaps we find that a student who rates problems infrequently assigns two particular problems a 1-star (low) rating. Given the lack of input from this student, these data may be especially salient and require special consideration compared to a student who frequently rates problems, and with high variability. Such possibilities seem to suggest that we treat discrete survey data (even with high-polling rates) differently than sensors that continuously provide data.

4.2 (Sensor-Free) Detectors in the GIFT Architecture

For purposes of software implementation, detectors are essentially sensors (i.e., both process, filter, and/or aggregate streams of data to make inferences about student state); "detector processing" would be nearly identical to "sensor processing" within the *Sensor Module*. However, the input characteristics of software detectors are much different than those of sensors in the GIFT architecture, as the notion of a sensor within GIFT, to date, focuses on physical sensors. Detectors generally rely on student/user-initiated input mediated by the learning environment, but detectors might also be developed that do not rely on user-initiated input (e.g., a detector of "forgetting" based on time-lapse in usage of the ITS).

One possible resolution would have the *Domain Module* (and/or the *Tutor-User Interface*) as input to the *Sensors* element, so that software-based detectors that rely on tutor log data are also conceptualized as *Sensors*. This proposal may stretch the notion of *Sensors* too far. In response, one might include a new type of *Detector/Analysis Module* that would take *Domain Module* (and possibly *Pedagogical Module* or *Tutor-User Interface*) data as input and provide information to the *User Module* about learners' affective and cognitive states via software detectors. This achieves the goal of keeping the relatively domain-independent detectors outside of the *Domain Module*. This requires that *Domain Module* output is sufficiently rich for use by detectors; as currently conceptualized, this is not clear.

---

[6] The HPIT architecture maintains such flexibility so that the investigator is free to make (or not make) distinctions about persistent versus non-persistent student characteristics (and concomitant timing decisions about assessment, measurement, or detection).

5 Discussion

Overall, we suggest that the GIFT architecture is well-served by considering the consequences of integrating a broader range of input and output relationships among its component modules (or possibly new types of modules) and other functional elements, including considerations of the presence, timing, granularity, and content of data passed between components.

Current research provides for data-driven means to use CT (and other CBTS) logs to classify and "detect" student behavior and affect without physical sensors, whether transactions and inputs are student-initiated or system-initiated. Integrating capabilities necessary for HPIT will be a fruitful extension of GIFT.

Furthermore, detectors rely on data from the ITS to determine whether students are off-task, gaming, bored, frustrated, etc. Such detectors require relatively rich log data and would not be served by the impoverished (i.e., abstract) assessment categories of "above standard," "below standard," etc., provided by the *Domain Module*. This suggests that detectors are a part of the *Domain Module*, but they are also (relatively) domain independent. Thus, it is not clear that they should be included in the *Domain Module*. Requiring detectors be a part of the *Domain Module* would also incur costs in terms of reusability and modularity. Alternatively, richer data might be provided to an enhanced *Learner Module* that subsumes (aspects of) the *Sensor Module* and our proposed detectors (i.e., the *Detector/Analysis Module*) to better infer characteristics of a learner's state. Further, other open questions remain as to the proper placement of other components of CTs within the GIFT architecture.

6 References

1. Ritter, S., Anderson, J.R., Koedinger, K.R., Corbett, A.T.: Cognitive Tutor: Applied Research in Mathematics Education. Psychonomic Bulleting & Review 14, 249-255 (2007)
2. Anderson, J.R.: Rules of the Mind. Erlbaum, Hillsdale, NJ (1993)
3. Sottilare, R.A., Brawner, K.W., Goldberg, B.S., Holden, H.K.: The Generalized Intelligent Framework for Tutoring (GIFT). (2012), http://www.gifttutoring.org/
4. Baker, R. S., Corbett, A. T., Koedinger, K .R., & Wagner, A. Z.: Off-Task Behavior in the Cognitive Tutor Classroom: When Students "Game the System". In: Proceedings of ACM CHI 2004: Computer-Human Interaction, pp. 383-390 (2004)
5. Baker, R.S.J.d.: Modeling and Understanding Students' Off- Task Behavior in Intelligent Tutoring Systems. In: Proceedings of the 2007 Conference on Human Factors in Computing Systems, pp. 1059-1068 (2007)
6. Aleven, V., & Koedinger, K. R.: Limitations of Student Control: Do Students Know When They Need Help? In: Proceedings of the 5th International Conference on Intelligent Tutoring Systems, pp. 292-303 (2000)
7. Baker, R.S.J.d., de Carvalho, A. M. J. A.: Labeling Student Behavior Faster and More Precisely with Text Replays. In: Proceedings of the 1st International Conference on Educational Data Mining, pp. 38-47 (2008)
8. Baker, R.S.J.d., Corbett, A.T., Roll, I., Koedinger, K.R.: Developing a Generalizable Detector of When Students Game the System. User Modeling & User-Adapted Interaction 18, 287-314 (2008)

9. Walonoski, J.A., Heffernan, N.T.: Detection and Analysis of Off-Task Behavior in Intelligent Tutoring Systems. In: Proceedings of the 8th International Conference on Intelligent Tutoring Systems, pp. 382-391 (2006)
10. Cocea, M., Hershkovitz, A., Baker, R.S.J.d.: The Impact of Off-Task and Gaming Behavior on Learning: Immediate or Aggregate? In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, pp. 507-514 (2009)
11. Fancsali, S.E.: Variable Construction and Causal Discovery for Cognitive Tutor Log Data: Initial Results. In: Proceedings of the Fifth International Conference on Educational Data Mining, pp. 238-239 (2012)
12. Fancsali, S.E.: Constructing Variables that Support Causal Inference. Ph.D. Thesis, Department of Philosophy, Carnegie Mellon University, Pittsburgh, PA, USA (2013)
13. Baker, R.S.J.d., Gowda, S.M., Wixon, M., Kalka, J., Wagner, A.Z., Salvi, A., Aleven, V., Kusbit, G., Ocumpaugh, J., Rossi, L.: Sensor-free Automated Detection of Affect in a Cognitive Tutor for Algebra. In: Proceedings of the 5th International Conference on Educational Data Mining, pp. 126-133 (2012)
14. Pardos, Z.A., Baker, R.S.J.d., San Pedro, M.O.C.Z., Gowda, S.M., Gowda, S.M.: Affective States and State Tests: Investigating How Affect Throughout the School Year Predicts End of Year Learning Outcomes. In: Proceedings of the 3rd International Conference on Learning Analytics and Knowledge (2013)
15. Sottilare, R., Proctor, M.: Passively Classifying Student Mood and Performance within Intelligent Tutors. Educational Technology & Society 15, 101-114 (2012)
16. Walkington, C., Sherman, M.: Using Adaptive Learning Technologies to Personalize Instruction: The Impact of Interest-Based Scenarios on Performance in Algebra. In: Proceedings of the 10th International Conference of the Learning Sciences, pp. 80-87 (2012)
17. Walkington, C.: Using Learning Technologies to Personalize Instruction to Student Interests: The Impact of Relevant Contexts on Performance and Learning Outcomes. Journal of Educational Psychology (forthcoming)
18. Duckworth, A.L., Peterson, C., Matthews, M.D., Kelly, D.R.: Grit: Perseverance and Passion for Long-Term Goals. Journal of Personality and Social Psychology 92, 1087-1101 (2007)
19. Duckworth, A.L., Quinn, P.D.: Development and Validation of the Short Grit Scales (Grit-S). Journal of Personality Assessment 91, 166-174 (2009)
20. Bernacki, M. L., Nokes-Malach, T.J., Aleven, V.: Fine-Grained Assessment of Motivation Over Long Periods of Learning with an Intelligent Tutoring System: Methodology, Advantages, and Preliminary Results. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies. Berlin: Springer (forthcoming)
21. Elliot, A. J., & McGregor, H. A.: A 2 X 2 Achievement Goal Framework. Journal of Personality and Social Psychology 80, 501-519 (2001)
22. Elliot, A. J., & Murayama, K.: On the Measurement of Achievement Goals: Critique, Illustration, and Application. Journal of Educational Psychology 100, 613-628 (2008)
23. Koedinger, K.R., Baker, R.S.J.d., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the EDM Community: The PSLC DataShop. In: Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.J.d. (eds.) Handbook of Educational Data Mining, pp. 43-55 Boca Raton, FL: CRC Press (2011)

Authors

Stephen E. Fancsali is a Research Scientist at Carnegie Learning, Inc. He received a Ph.D. in Logic, Computation, and Methodology from the Philosophy Department at Carnegie Mellon University in May 2013. His doctoral research centered on the construction of variables from fine-grained data (e.g., intelligent tutoring system log files) to support causal inference and discovery from observational data sets. At Carnegie Learning, he focuses on a variety of issues in educational data mining, including student modeling, providing interpretable ways to quantify improvements in cognitive models and other tutoring system components, and statistical and causal modeling of student affect, behavior, and other phenomena as they relate to learning and other education outcomes, especially in intelligent tutoring systems.

Steven Ritter is Co-Founder and Chief Scientist at Carnegie Learning, Inc. He received a Ph.D. in Psychology from Carnegie Mellon University.

John Stamper is Technical Director of the Pittsburgh Science of Learning Center (PSLC) and a faculty member at the Human-Computer Interaction Institute at Carnegie Mellon University. He received a Ph.D. in Information Technology from the University of North Carolina at Charlotte.

Tristan Nixon is a Research Programmer at Carnegie Learning, Inc. He earned a B.S. in Computer Science from the University of Toronto.