Toward Scalable Improvement of Large Content Portfolios for Adaptive Instruction

Stephen E. Fancsali
Hao Li
Steven Ritter
Carnegie Learning, Inc.
{sfancsali, hli, sritter}
@carnegielearning.com

ABSTRACT

Recent literature demonstrates data-driven improvements to content used in adaptive instructional systems like intelligent tutoring systems, following a multi-method approach to "design loop adaptivity." Examples from the literature are often relatively bespoke, focusing on a particular piece of content within a system and applying several, often time-consuming, methods to redesign important elements of content and deliver improved learning experiences. We draw attention to the problem of targeting and focusing design-loop adaptivity to make such data-driven improvement more scalable for large content portfolios. *Targeting* involves choosing the goal to be achieved by this improvement and the content that requires improvement. We build on our recent work on targeting by also considering what we call *focusing* design-loop adaptivity, which involves determining what aspects of the learning experience require (the most) improvement and the method(s) by which to achieve improvement. We present examples of how targeting may proceed and raise important questions about how to focus data-driven learning engineering improvement processes.

Keywords

Learning engineering; design-loop adaptivity; intelligent tutoring systems.

1. INTRODUCTION

Extensive literature in educational data science considers data-driven methods for improving existing instructional content in adaptive instructional systems like intelligent tutoring systems (ITSs). Nearly fifteen years ago, Cen et al. [6], for example, proposed Learning Factors Analysis (LFA), a semi-automated search technique to discover better cognitive skill, or knowledge component (KC) [16], models that are often used to drive adaptivity in ITSs.

ITSs like Carnegie Learning's MATHia (formerly Cognitive Tutor [18]), rely on KC models as a part of their knowledge tracing [8] approach to mastery learning [19]. While students work within a particular topical "workspace" in MATHia, complex, multi-step problems are selected for them based on the KCs associated with that workspace that the system has yet to judge as mastered by the student. Students make progress through sequences of instructional content (or MATHia workspaces) by demonstrating mastery of the KCs associated with each workspace. Starting from KC models specified by ITS developers and content authors, empirical, "close the loop" studies have demonstrated that using data-driven techniques to improve KC models (e.g., by "splitting" one or more existing KCs in a model into one or more new KCs) can drive improved learning outcomes in ITSs, for example, by enabling students to master content more efficiently (e.g., [15]). Nevertheless, there are a bevy of features of instructional content (e.g., KC model parameters, various elements of user-interface and task design) that might be reasonably improved, beyond underlying KC models, to drive better learning outcomes for students.

Going beyond KC model refinements, more recent literature proposes and demonstrates a data-driven, multi-method approach to systematically improving instructional content or "design-loop adaptivity" [13]. While these methods are promising to improve learning outcomes, examples in the literature, whether using methods like LFA, or more recent design-loop adaptivity efforts, tend to be relatively bespoke. Demonstrations start with a particular target piece of content (e.g., the *Algebraic Expressions* unit in a free, online ITS called Mathtutor [2, 13]), working through an improvement process, and demonstrating improved outcomes in an experimental or similar study. Since LFA and steps within the design-loop adaptivity process, detailed in the next section, can be relatively time-consuming and/or computationally expensive, we here seek to consider ways in which improvement processes might be both targeted and focused for developers and learning engineering working with large portfolios of instructional content (e.g., hundreds, or soon, thousands of MATHia workspaces) that might need improvement.

After briefly describing recent work on design-loop adaptivity, we detail our recent efforts at Carnegie Learning to target content for data-driven improvement based on several different goals [11], picking workspaces from amongst hundreds each academic year (or major software release) for iterative improvement, and point to important open questions for how we might use data to focus design-loop adaptivity, or more generally content improvement and/or redesign efforts. Once a target workspace has been identified, focusing improvement efforts will involve determining what aspects of the learning experience require (the most) improvement or which goals and methods of the design-loop

adaptivity process laid out in recent literature ought to be prioritized given finite learning engineering, software development, content authoring, and/or instructional design resources.

2. DESIGN-LOOP ADAPTIVITY

2.1 Three Timescales for Adaptivity

Aleven et al. [1] describe design-loop adaptivity as involving "data-driven decisions made by course designers before and between iterations of system design, in which a course or system is updated based on data about student learning" collected via the course or system. They contrast the relatively long timescale of design-loop adaptivity with adaptivity that occurs on a much shorter timescale like "task-loop" and "step-loop" adaptivity, or "outer-loop" and "inner-loop" adaptivity, respectively, as described in a popular taxonomy of ITS behaviors laid out by VanLehn [22].

Inner-loop or step-loop adaptivity in an ITS supports students within tasks or problems, providing affordances like just-in-time feedback to particular incorrect answers and hints that are sensitive to a student's chosen problem-solving strategy. Outer-loop or task-loop adaptivity drives learning activity or problem/task selection based on student performance; outer loop adaptivity, might, for example, involve problem selection that emphasizes problems associated with unmastered KCs, using a framework like Bayesian Knowledge Tracing (BKT; [8]) to monitor student progress toward KC mastery.

In addition to describing the timescale for different forms of adaptivity, the "Adaptivity Grid" framework due to Aleven et al. [1] also details a variety of goals at which any of these three types of adaptivity could be directed (i.e., what characteristics of learners adaptation is intended to address), including students' prior knowledge and knowledge growth, and the paths that students take through a problem (e.g., problem-solving strategies and the errors students make), among others.

This inherent goal orientation of adaptivity drives methods used in the systematic approach to design-loop adaptivity we describe in the next section and will also naturally apply to our discussion of targeting and focusing design-loop adaptivity.

2.2 A Systematic Approach to Design-Loop Adaptivity

In introducing a systematic approach to using data to drive iterative improvements to instructional content, Huang et al. [13] note that while many studies demonstrate how data mining methods can be used to improve prediction accuracy, "there is no good general guidance for how to convert data-mining outcomes into better tutor design" [13].

Huang et al. [13] describe three broad goals that can drive data-driven content improvement and redesign; they include two or more sub-goals for each broad goal and provide specific methods intended to achieve each sub-goal. We briefly review the three overall goals, sub-goals, and (necessarily, non-exhaustive) methods that Huang et al. [13] propose to achieve these goals before considering the targeting and focusing of this process.

2.2.1 Goal #1: Refine the KC model.

Huang et al. [13] propose two sub-goals to achieve KC model refinement. They first propose identifying difficulty factors to "split" KCs (i.e., decomposing an existing KC into one more new, hypothetical KCs) and then comparing hypothesized KC models. Difficulty factors are characteristics of learning tasks that may make them more difficult than similar tasks (e.g., properties of some problems or problem-steps in a workspace that might make them more difficult than other problems or problem-steps in the same workspace).

The semi-automated (but more computationally intensive) LFA method would also achieve similar goals, typically "seeded" with possible ways in which to split KCs by human experts (or perhaps other data-driven means). The "difficulty factor effect analysis" regression approach (to find associations between difficulty factors and student performance on KCs) proposed by Huang et al. [13] is explicitly noted as a potential "efficient simplification of LFA." They propose to use the additive factors models [6] and analyst inspection [21] to compare resulting, hypothesized KC models.

2.2.2 Goal #2: Redesign content.

Among a bevy of ways in which content and learning tasks can be redesigned (see §4.1 for two more, for example), Huang et al. [13] consider three redesign sub-goals. First, starting from a redesigned KC model achieved in Goal #1 (or possibly an existing KC model not subjected to Goal #1 refinements), they suggest estimating the number of opportunities to achieve mastery for KC in the model, as well as estimating the extent to which under-practice or over-practice may be occurring for each KC. Huang et al. [13] introduce a method they call "probability-propagation practice estimation" to accomplish this sub-goal. Other methods have been proposed in the literature (e.g., [14]). Second, they suggest creating focused practice tasks for difficult KCs (that eliminate steps in which students must practice easier KCs) seeking to reduce both over-practice of easier KCs and under-practice of more difficult KCs. Third, they suggest an analysis of student errors to creating feedback messages on frequent student errors.

2.2.3 Goal #3: Optimize individualized learning.

The last goal of the design-loop adaptivity methodology of Huang et al. [13] is to optimize individualized learning via optimizing the parameters of the student model (e.g., BKT parameters for each KC in the refined model) and task selection.[1] An important facet of the student model (and an ITS's implementation of such models) is whether it permits optimizing and/or individualizing parameters at a KC-level, student-level, or perhaps both.

The BKT framework [8] provides a two-state representation of student knowledge of each KC; a student is either in the "unknown" or "known" state for each KC at any given time. In its original formulation, BKT posits four parameters per KC that are used, along with student performance data, to track student progress to reaching the known state, or mastery, via an evolving estimated probability that a student is in the known (or mastered) state for each KC. Parameters for each KC include the probability that a student has prior knowledge of the KC (i.e., begins practice

[1] A reviewer noted that the scope of this goal as laid out by Huang et al. [13] may be incomplete in at least the sense that the chosen student model (e.g., BKT) is taken as given rather than being considered as a possible target for change and improvement. Whether a target learning platform or ITS is sufficiently flexible to allow for changes to the student model (e.g., adopting an alternative to BKT, perhaps for particular pieces of content) within the context of design-loop adaptivity content improvements raises an important design consideration (or future possibility) for such systems.

in the known or mastered state), the probability that a student transitions from the unknown to the known state at a particular practice opportunity, the probability that a student's performance at an opportunity represents "guessing" correctly despite being in the unknown state, and the probability that a student "slips" and produces an incorrect response despite knowledge of a KC. In the BKT implementation used by MATHia, a KC is considered mastered when the system's estimate of the probability that a student has reached the known or mastered state for the KC exceeds a conventional 0.95 threshold. Parameter optimization and individualization (e.g., within the BKT framework and variants thereof) are topics of extensive literature in educational data mining and related literature (e.g., [3, 7]).

Huang et al. [13] finally suggest simulating task selection [9] to optimize this facet of an intelligent tutor's presentation of learning activities to students based on the (now optimized) student model. Within the BKT framework, one factor to possibly consider in simulating task selection is varying the mastery threshold, perhaps considering values other than the conventional probability of 0.95, among other facets of variation and pedagogical rules.

2.2.4 Targeting & Focusing

Working through all three goals and their corresponding sub-goals can be a recipe for near complete redesign of particular instructional content, and there are cases in which near complete redesign is likely appropriate. However, given limited learning and software engineering resources, prioritizing which pieces of content ought to be targets for improvement or redesign as well as focusing improvement or redesign efforts on particular goals and sub-goals of design-loop adaptivity (or possibly other improvement) efforts would be beneficial to being able to improve instruction within large content portfolios.

We propose that developing methods to target and focus design-loop adaptivity could rely on data to determine:

1. *Targeting*: What content ought to be prioritized for data-driven improvement?
2. *Focusing*: Which of the overall goals (or sub-goals) of design-loop adaptivity are most important for delivering improved instructional content? Which methods should be applied to achieve this improvement?

3. TARGETING

Before getting down to the work of improving instructional content, learning engineers and technology developers must first identify which content is to be targeted for such efforts. Any of a variety of goals might inform what instructional content is targeted for data-driven improvement efforts. We briefly consider two goals and corresponding metrics for targeting MATHia workspaces for data-driven improvement.

The first targeting metric is considered in detail in our recent work [11], which serves as a companion piece to the present work. After briefly discussing our first targeting metric, we consider a second metric that was omitted for brevity from our recent work before moving on to consider goals, metrics, and open questions, concerning how to focus data-driven improvement or design-loop adaptivity work.

3.1 Target #1: Failures to Reach KC Mastery

Students working in MATHia make progress within an instructional sequence of topical workspaces by mastering all of the KCs associated with the workspace before reaching the maximum number of problems for that workspace. The maximum number of problems is set by instructional designers and is usually 25. If BKT has yet to judge the student as having mastered all KCs in a workspace when the student reaches the maximum number of problems, MATHia moves students on to the next workspace in their assigned curriculum sequence without mastery. The student's teacher is notified via MATHia's reporting analytics as well as within the LiveLab classroom orchestration companion app to MATHia, if the teacher is using it in their classroom.

The extent to which students fail to master KCs varies considerably across workspaces. Our recent work [11] considers data from 308 MATHia workspaces used during the 2018-19 academic year. Data included work in math content from Grades 6-8, Algebra I-II, and Geometry. The typical (median) workspace had 4.3% of students fail to reach mastery of at least one KC. Some workspaces have no such failures to reach mastery, and the workspace with the greatest proportion of failures to reach KC mastery had nearly 78% of student failing to reach mastery [11].

While especially high proportions of students failing to master pieces of instructional content are likely to be important factors in determining what learning content to improve with limited resources, there are other facets of the user experience that can raise obstacles to learning and practical factors that must be considered when managing and improving large portfolios of content. We turn now to a more practically-focused, composite metric developed by a curriculum developer to help target their work.

3.2 Target #2: A Composite (Design) "Attention Metric" or Index

A cross-functional team of instructional designers, cognitive scientists, and subject-matter expert content creators at Carnegie Learning is responsible for MATHia's content creation and continuous improvement. This team has collaborated with data scientists over nearly a decade to iteratively refine a composite index or "attention metric" to roughly prioritize workspaces requiring the most design and/or learning engineering "attention" for improvements to ensure satisfying, effective learning experiences for learners. User acceptance testing of iterative improvements to the attention metric has taken the form of identifying sets of workspaces for which the team largely agrees there are improvement needs through quality assurance testing, customer service reports, software bug reports, data analysis, and related means. Different ways of "weighting" particular measures within the index are then tested to see resulting lists of prioritized workspaces until rough consensus is reached that a reasonable list of priority workspaces has been identified.

In larger organizations, entire teams (or particular personnel on a team) might be responsible for targeting improvements based on any of the particular components of such an index, which effectively mixes (at least) goals of design-loop adaptation to student knowledge and motivation and affect. The current attention metric includes measures of the following factors, increases in any of which increase the extent to which developer attention ought to be drawn to a particular workspace:

- *Usage* helps to target developers to fix and improve content that is used broadly and is considered in two ways: (1) The rank over all workspaces of the total number of users of the workspace across MATHia's user-base; this provides a measure along which to "weight" the other factors in this index, and (2) the proportion of learners who abandon a workspace

after starting it (i.e., begin but do not complete a workspace). The abandonment measure captures issues like the extent to which teachers choose to move students beyond particular workspaces without completion and might indicate either teacher dissatisfaction or student frustration with the content. High rates of usage and/or abandonment increase the extent to which developers ought to seek to improve content.

- *Failure to Reach Mastery:* The proportion of students who reach the maximum number of problems in the workspace but fail to reach mastery of all KCs in the workspace (see §3.1 and [11]). High failure rates also increase the attention metric to direct attention to such content.
- *Completion Time*: Workspaces have "excessive" completion time to the extent that the average time to completion exceeds a target of 50 minutes.[2] If the average time is less than 50 minutes, this factor does not contribute to the workspace's attention metric value.
- *Problem-Level Usability Concerns:* The proportion of users who must be "skipped" over a problem within a workspace or have a problem "restarted" within a workspace by their teacher, which may indicate that there is a task design issue within a problem (or software bug) creating ineffective learning experiences. These issues are relatively rare, but when they occur in even an exceedingly low proportion of cases, instructional designers and software engineers quickly seek to rectify these issues.

Workspaces are ranked by their attention metric value (which is placed on a 0-100 scale, roughly corresponding to percentiles over all workspaces) within an internal learning engineering dashboard, which also provides its users with entrée to the various components of the index to better understand where and how particular workspaces may be failing to deliver effective learning experiences. We will return to considering particular facets of the attention metric when we discuss focusing design-loop adaptivity in §4.

The current iteration of the attention metric incorporates the relative frequency with which learners fail to master KCs in workspaces while also taking other practical aspects of the user experience into consideration. Over the workspaces included in our analysis (see [11]), the attention metric has a Pearson correlation of $r = 0.61$ ($p < .001$) with the relative frequency of failures to reach KC mastery.

Rather than focus on details of the more practically focused attention metric that is currently used by the system's developers, we merely seek to illustrate ways in which various goals might be addressed by design-loop adaptivity as well as the targeting metrics that learning engineers use to drive an improvement process for such adaptivity.

There is variation in the extent to which particular metrics for targeting design-loop adaptivity suggest ways in which such improvement or redesign work might be focused on particular methods. Several components of the attention metric, for example, naturally suggest ways to possibly focus design-loop adaptivity work (i.e., to single out or prioritize particular methods for data-driven redesign and improvement of a particular piece of content). Such focusing would provide new means by which to guide data-driven improvement of adaptive instructional content. We now consider focusing design-loop adaptivity in more detail.

4. FOCUSING

Faced with large content portfolios (e.g., hundreds of deployed MATHia workspaces used by hundreds of thousands of learners every year), learning engineers and developers need data-driven guidance on when a wholesale redesign versus more focused improvements and modifications may suffice for rapid and/or scalable (if sometimes incremental) improvements to outcomes. Bespoke, systematic approaches in the literature (e.g., working through methods addressing each of the three goals and associated sub-goals described in §2.1.1-§2.1.3) to improving particular pieces of content have been shown to drive improved learning, but there is also evidence that relatively simple improvements and partial redesigns may also drive improved outcomes. We consider two examples of how such improvement and redesign might be focused and raise questions for future work.

4.1 Focus #1: Problem-Step Engagement

For example, Fancsali et al. [12] recently described relatively modest, iterative task redesign (a la design-loop adaptivity Goal #2, §2.1.2) in a MATHia workspace on *Solving Quadratic Equations* that were associated with a 10.3 percentage point decrease in the proportion of students who failed to reach KC mastery in the second iteration of improvement. The workspace was targeted for improvements because 32.1% of students failed to reach KC mastery in the workspace in the 2018-19 school year.

In a more focused approach to improving this workspace rather than a wholesale redesign, learning engineers first developed more extensive (optional) scaffolding for the components of the quadratic formula to support students in using the formula to solve quadratic equations. The optional, enhanced scaffolding, however, in its initial deployment (in the 2019-20 school year release of MATHia), did not display to students by default. Rather, students had to expand the scaffolding to engage with it. A comparison of the proportion of students failing to reach mastery in 2019-20 to the previous school year did not reveal substantial improvements (32.1% in 2018-19 to 31.9% in 2019-20). Further, student usage data indicated that students weren't engaging with the enhanced, optional scaffolding's problem-solving steps. In the 2019-20 MATHia release, scaffolding was automatically displayed to students, while still remaining optional, after they chose to use the quadratic formula to solve a quadratic equation. With "displayed by default" optional scaffolding, only 24.1% of students failed to master all KCs in the workspace through March 1, 2021 (compared to 34.3% of students over the same period, through March 1, 2020, in the 2019-20 school year). This, still elevated, rate of failure to reach KC mastery may indicate that more comprehensive data-driven improvement is still appropriate, but substantial improvement appears likely driven by a relatively modest redesign that did not involve changes to the KC model or individualized learning parameters.

[2] MATHia workspaces that exceed approximately 50 minutes for the average learner to complete run the risk of disengaging and de-motivating students, as students are likely to work through an entire math class period without making progress to another workspace.

4.2 Focus #2: KC Model Deficiencies

Other modeling techniques might be used to determine that MATHia workspaces (or other instructional content) suffer from deficiencies in the underlying KC model that drives features like task-loop adaptivity or knowledge tracing. One such method considers patterns in KC learning curves that may suggest that an existing KC model omits KCs that, when included in student model for a particular piece of content, would better capture student learning. Such an omitted KC may represent a difficulty factor that serves as input to LFA search or difficulty factor effect analysis proposed by Huang et al. [13].

Fancsali et al. [10] considered the "segmented" learning curve [17] illustrated in Figure 1 as a way to emphasize how data can be used to inform instructional redesign to improve learning for all learners. The learning curve categorizes students into groups by the number of opportunities (≤ 5, 6-10 [≤ 10], etc.) before which they reached KC mastery by MATHia's implementation of BKT. The majority of students (in the top two learning curves) appear to have a unified conception of x- and y-intercept, yielding the top curve, which is monotonically increasing, and the relatively smoothly increasing second and third curves, generally indicative of learning of a single skill or KC over these practice opportunities.

The discussion of [10] focused on the relatively small number of students in the "lower" segments of the curve for which a "saw-tooth" pattern manifests. A saw-tooth pattern emerges at alternating opportunities to practice a KC related to plotting a linear function based on its x-intercept (at odd numbered opportunities) and y-intercept (at even numbered opportunities). Students appear to be having more difficulties specifying the x-intercept of a given function compared to the y-intercept.

The emergence of the saw-tooth pattern, even for a relatively small proportion of students, represents a clear difficulty factor that might serve as a place in which the current KC might be "split" into at least two KCs by methods like LFA or difficulty factor effect analysis. In this way, inspection of segmented learning curves (and perhaps semi-automated analysis of such curves) might serve as entry points to the more comprehensive process of design-loop adaptivity detailed above.

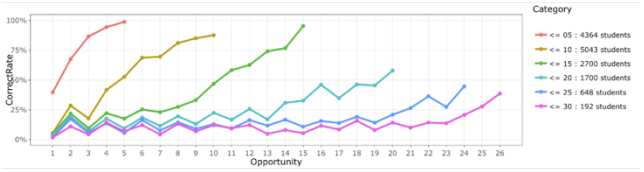

Figure 1. Segmented learning curve for a KC concerning plotting a given linear function by specifying its x- and y-intercepts (see [10]).

Nevertheless, future methodological work should consider ways in which guidance might be provided as to which of the design-loop adaptivity goals and methods are most important for a particular piece of content. Perhaps merely specifying the improved KC model and optimizing its parameters, with limited task redesign, would produce as much learning as a carefully crafted task redesign? Data-driven methods might help to illuminate places in which task redesign ought to be prioritized despite a lack of obvious areas for improvement in a KC model. This may be the case, for example, for content targeted on the basis of student motivation or meta-cognition. Students may, for example, display behaviors like gaming the system [4] or affective states like confusion [5] that could be addressed by task redesign, but without much need for modifications to the KC model or individualized learning parameters.

Empirical "close the loop" studies and A/B tests of variation of content improvement may illuminate insights into which facets of improvement and redesign deliver the best learning gains relative to the time-investment required to achieve such improvements. Such prioritization would be especially important and helpful to guide large-scale learning engineering efforts to efficiently improve large portfolios of content.

5. DISCUSSION

Our recent work [11] focused on ways in which content might be targeted for data-driven improvement processes, focusing on just a small subset of possible goals for such improvement. We here consider how to build on targeting by focusing improvement efforts in specific ways that may drive improved learning outcomes. Rather than provide definitive answers to questions about targeting or focusing, we seek to call attention to these issues.

Our efforts, as well as more comprehensive, multi-method design-loop adaptivity approaches advocated by Huang et al. [13], are centered on the idea that data-intensive modeling approaches ought to be developed in ways that provide guidance to researcher and developers about how adaptive instruction might be improved. These efforts are aligned with broader, recent calls for explanatory learner models (e.g., [20]), which emphasize the importance of going beyond mere improvements in student performance prediction accuracy to modeling that may lead to substantive improvements to instruction. We enthusiastically agree with such calls for explanatory learner models and emphasize the importance of practical and scalable data-driven methods to drive targeted and focused improvements in adaptive instruction.

6. ACKNOWLEDGMENTS

This research is funded by the National Science Foundation under the award The Learner Data Institute (Award #1934745). Opinions, findings, and results are solely those of the authors and do not reflect those of the National Science Foundation.

7. REFERENCES

[1] Aleven, V., McLaughlin, E.A., Glenn, R.A., and Koedinger, K.R. 2017. Instruction based on adaptive learning technologies. In *Handbook of Research on Learning and Instruction*, 2nd Ed., Routledge, New York, 522-560.

[2] Aleven, V., and Sewall, J. 2016. The frequency of tutor behaviors: a case study. In *Intelligent Tutoring Systems 2016*. ITS 2016. LNCS, vol. 9684. Springer, Cham, 396–401. https://doi.org/10.1007/978-3-319-39583-8_47

[3] Baker, R.S.J.d., Corbett, A.T., and Aleven, V. 2008. More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. In *Intelligent Tutoring Systems 2008.* ITS 2008. LNCS, vol. 5091. Springer-Verlag, Berlin, 406-415.

[4] Baker, R.S., Corbett, A.T., Koedinger, K.R., and Wagner, A.Z. 2004. Off-task behavior in the Cognitive Tutor classroom: When students "game the system." In

Proceedings of ACM CHI 2004: Computer-Human Interaction, 383-390.

[5] Baker, R.S.J.d., Gowda, S.M., Wixon, M., Kalka, J., Wagner, A.Z., Salvi, A., Aleven, V., Kusbit, G.W., Ocumpaugh, J., and Rossi, L. 2012. Towards sensor-free affect detection in Cognitive Tutor Algebra. In *Proceedings of the 5th International Conference on Educational Data Mining* (Chania, Greece, 2012). EDM 2012. IEDMS, 126-133.

[6] Cen, H., Koedinger, K.R., and Junker, B. 2006. Learning factors analysis: A general method for cognitive model evaluation and improvement. In *Intelligent Tutoring Systems 2006*. ITS 2006. Springer-Verlag, Berlin, 164-175.

[7] Cen, H., Koedinger, K.R., and Junker, B. 2007. Is over practice necessary? improving learning efficiency with the cognitive tutor through educational data mining. *Front. Artif. Intell. Appl.* 158, 511

[8] Corbett, A.T., and Anderson, J.R. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction* 4, 253-278.

[9] Doroudi, S., Aleven, V., and Brunskill, E. 2017. Robust evaluation matrix: towards a more principled offline exploration of instructional policies. In *Proceedings of the 4th (2017) ACM Conference on Learning @ Scale* (April 20-21, 2017, Cambridge, MA). L@S 2017. ACM, New York, NY, 3-12.

[10] Fancsali, S.E., and Ritter, S. 2020. Data-intensive learning engineering & applied education research with Carnegie Learning's MATHia Platform. In *Proceedings of the 1st Workshop of the Learner Data Institute at EDM 2020*.

[11] Fancsali, S.E., Li, H., Sandbothe, M., and Ritter, S. 2021. Targeting design-loop adaptivity. In *Proceedings of the 14th International Conference on Educational Data Mining 2021* (Paris, France, June 29 - July 2, 2021). EDM 2021. IEDMS.

[12] Fancsali, S.E., Pavelko, M., Fisher, J., Wheeler, L., and Ritter, S. 2021. Scaffolds and nudges: A case study in learning engineering design improvements. In *Artificial Intelligence in Education 2021.* AIED 2021. LNCS, vol. 12749. Springer, Cham, 441-445. https://doi.org/10.1007/978-3-030-78270-2_78

[13] Huang Y., Aleven V., McLaughlin E., and Koedinger K. 2020. A general multi-method approach to design-loop adaptivity in intelligent tutoring systems. In *Artificial Intelligence in Education 2020.* AIED 2020. LNCS, vol 12164. Springer, Cham, 124-129. https://doi.org/10.1007/978-3-030-52240-7_23

[14] Lee, J.I., and Brunskill, E. 2012. The impact of individualizing student models on necessary practice opportunities. In *Proceedings of the 5th International Conference on Educational Data Mining* (Chania, Greece, 2012). EDM 2012. IEDMS, 118-125.

[15] Liu, R., and Koedinger, K.R. 2017. Closing the loop: Automated data-driven cognitive model discoveries lead to improved instruction and learning gains. *Journal of Educational Data Mining* 9(1), 25-41.

[16] Koedinger, K.R., Corbett, A.T., and Perfetti, C. 2012. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science* 36(5), 757-798.

[17] Murray, R.C., et al. 2013. Revealing the learning in learning curves. In *Artificial Intelligence in Education 2013* (Memphis, TN, USA, Jul 9-13, 2013). LNCS, vol. 7926. AIED 2013. Springer-Verlag, Berlin, 473-482. https://doi.org/10.1007/978-3-642-39112-5_48

[18] Ritter, S., Anderson, J.R., Koedinger, K.R., and Corbett, A.T. 2007. Cognitive Tutor: applied research in mathematics education. *Psychon. B. Rev.* 14, 249-255.

[19] Ritter, S., Yudelson, M., Fancsali, S.E., and Berman, S.R. 2016. How mastery learning works at scale. In *Proceedings of the 3rd (2016) ACM Conference on Learning at Scale* (April 25 - 26, 2016, Edinburgh, UK). L@S 2016. ACM, New York, NY, 71-79.

[20] Rosé, C.P., McLaughlin, E.A., Liu, R., and Koedinger, K.R. 2019. Explanatory learner models: Why machine learning (alone) is not the answer. *Br J Educ Technol* 50, 2943-2958. https://doi.org/10.1111/bjet.12858

[21] Stamper, J.C., and Koedinger, K.R. 2011. Human-machine student model discovery and improvement using Datashop. In *Artificial Intelligence in Education 2011*. AIED 2011. LNCS (LNAI), vol. 6738. Springer, Heidelberg, 353–360. https://doi.org/10.1007/978-3-642-21869-9_46

[22] VanLehn, K. 2006. The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education* 16(3), 227-265.