A Learning Sciences Approach for Post-hoc Algorithm Explainables

First Author[1] and Second Author[2]

[1] Institution 1, City1 ST 12345, ABC email1@email.edu
[2] Institution 2, City1 ST 12345, ABC email2@email.edu

Abstract. As machine intelligence is increasingly incorporated into the world around us, it becomes imperative for users to understand the potential biases and flaws of the algorithms on which their systems rely. Explainables have been developed to provide post-hoc explanation of complex algorithms for this purpose. While explaining can be viewed as analogous to teaching and learning, little work has been done to systematically apply research from the learning sciences to the design and evaluation of these explainable systems. In this paper, we identify key principles from the learning science literature, and apply them as a framework for designing an explainable to explain Bayesian Knowledge Tracing, an artificial intelligence algorithm originating from educational technology.

1 Introduction

In order for users to make informed, ethical decisions with the assistance of the algorithmic systems on which they rely, users must understand the algorithm's processes and therefore its potential flaws and biases [14]. The same is true with educational technology, which is increasingly being used by teachers and students to make decisions in the classroom, such as who is at risk of dropping out and on which topics to focus on that day [3]. This concern focusing on machine models' ability to be interpreted is seated within the machine learning literature as transparency and interpretability, and the approach to provide users with an understanding of their models is referred to as post-hoc explanations or explainable AI (xAI) [9].

In this case study, we adopt and explore the application of a set of principles from the learning sciences [1] in the design process of an explainable for Bayesian Knowledge Tracing (BKT), a complex algorithm underlying the Open Learning Initiative [15] and Open Analytics Research Service [3], used in numerous universities across the United States. Alongside our learning principles, we used a qualitative user-centered design process to identify user needs and important factors to consider in building future post-hoc explanations for complex algorithmic processes.

BKT was introduced in 1995 by Corbett & Anderson as a means to model students' knowledge as a latent variable within technologically enhanced learning (TEL) environments [5]. The TEL maintains an estimate of the probability that the student has learned a particular set of skills based on their performance on problems, which is statistically equivalent to a 2-node dynamic Bayesian network. The development of BKT allowed for more accurate student modeling and more personalized learning opportunities for students. BKT predicts whether a student has mastered a skill, or not yet mastered it (either due to lack of data, or repeated failed attempts). This process requires a skill map connecting problems to skills. Mastery predictions require four parameters to calculate, with a probability of 0.9-0.95 typically being used as a cut-off to qualify as skill mastery:

1. $\mathbf{P(L_0)}$: the probability the student already knew the skill
2. $\mathbf{P(T)}$: the probability that the student learned after a learning opportunity
3. $\mathbf{P(G)}$: the probability the student guessed correctly on an unknown skill
4. $\mathbf{P(S)}$: the probability the student made a mistake and slipped on a known skill

Twenty three years after BKT was first introduced, [6] examined some of the potential flaws in this model. BKT is particularly prone to a semantic model degeneracy issue, as it is essentially a two-state hidden Markov model. When BKT tries to fit the best two state hidden Markov model it can to the data, the model does not understand that what it is predicting is intended to represent mastering a skill, and so it might fit a model that conflicts with human understanding of skill mastery [6]. Furthermore, from conversations with practitioners using BKT-enabled software, we know it is possible for flaws to be introduced in the skill-mapping process that can yield confusing results for practitioners. As an example, if all the problems mapped to the learning goal of interpreting bar charts use vertical bar charts as exemplars, a large proportion of students will answer incorrectly on a problem with horizontal bar charts. Even open source algorithmically-enhanced systems developed by well meaning researchers should not be indiscriminately followed. Interrogating the system is therefore a user trait that should be encouraged and explored.

Figure 1 is a single screen from our Interactive Alchemy explainable. In this tutorial, users interacted with buttons and watched the animations that accompanied the descriptions. Vials representing the four parameters would fill a beaker representing the probability the student had mastered a hypothetical skill. Much of the BKT algorithm was illustrated through the animations of the liquid rising and falling in the vials. This explainable was implemented in HTML and Javascript, resulting in more than 20 unique screens. Representing predicted mastery as liquid in a beaker, lead to the liquid rising whether the hypothetical student answered a question correctly or incorrectly. This animation illustrates the BKT concept that every problem is practice in a skill, leading to an increased likelihood of mastery within that skill, regardless of the correctness of the student response.

2 Related Work

Model interpretability is pointed to as a potential avenue for increasing transparency of complex computational models, but it is as yet unknown how to

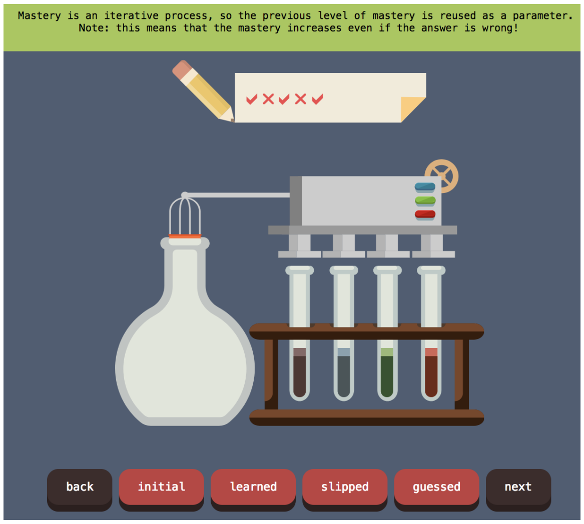

Fig. 1. A single screen from the Interactive Alchemy explainable for describing BKT interactively.

achieve the required level of interpretability to achieve the desired impact on user understanding and decision-making. Lipton (2018) introduces a desiderata of interpretability research, in which post-hoc interpretability is one property progressing toward the goals of trust, causality, transferability, and informativeness, as well as fair and ethical decision-making in machine learning models [14]. Of the four types of post-hoc interpretability detailed by Lipton (2018), explainables typically fall under the "Explanation by Example" category.

Based on this explanation by example category, the machine learning and visualization communities have begun creating what they call "explainables" to teach the concepts of particular algorithms. Explainables "that explain how AI techniques work using visualizations," to quote a recent workshop call for Visualization of AI[3], often take the form of interactive graphs or visualizations interspersed with paragraphs of explanatory text. However, this concept is not unique to machine learning nor the information visualization community, as very similar approaches can be found from within educational psychology under the name inquiry-based learning. In [8], inquiry-based learning is motivated as a requirement for understanding scientific inquiry and as a means to acquire, clarify, and apply an understanding of science concepts. The authors describe their collection of interactive learning technologies that provide interactive geosciences visualizations for novices to explore various atmospheric and meteorological sciences topics. These visualizations were integrated into a classroom curriculum and progressed from simple to complex activities and from specific instructions to open-ended tasks. Where a systemic application of inquiry-based learning of-

[3] http://visxai.io/

ten contains this scaffolding from the simple to the complex, explainables often mimic this process through a shorter process the community describes as going "up and down the ladder of abstraction" [4].

From within the machine learning community, discussion of how researchers might generate a rigorous science of interpretability focuses on methods with which to evaluate the interpretability of different models. In particular, Doshi-Velez & Kim point out scientific understanding, safety, ethics, mismatch objectives, and multi-objective trade-offs as incompleteness in models that produce unquantified bias [7]. While the authors introduce a sample approach for understanding user expertise via the basic units of the explanation or "cognitive chunks", they largely miss the modern literature on teaching and learning. Simply put, we can imagine the user as a learner and the post-hoc interpretation as the learning content, and this would allow xAI researchers to leverage the entire body of educational psychology and learning science research to achieve their goals of post-hoc explanations of complex algorithms.

While there is much research on measuring and evaluating what it means to know a concept from within the learning science field, this approach does not appear present in the work on explainables and algorithmic transparency via post-hoc explanation. A review of the example explainables listed in the IEEE VIS Workshop on Visualization for AI Explainability[3] shows a series of explanatory text interspersed with interactive visualizations, but no accompanying evaluations in the research literature. In short, the explainables community does not yet appear to empirically evaluate whether their explainables are successful in teaching the concepts the designers had intended. And in many cases, it is not clear how the explainable designers identified the concepts necessary to teach.

2.1 Effects of Interpretability on Actions

Assuming the desired level of interpretability is achieved, it is also as yet unknown how that interpretability will impact user behavior with algorithmically enhanced systems. Discussions within the ML community suggest that once a model increases its transparency, whether through post-hoc explanations or other techniques, that this will lead to more realistic trust in ML systems and eventually to fairer decisions made with the assistance of ML models [7].

However, research in the learning sciences suggests that increased transparency in grading can lead to student dissatisfaction and distrust [9, 11]. In particular, Kizilcec (2016) tested three levels of system transparency in a high-stakes essay peer assessment context within an online learning environment [11]. Individuals who received their expected grade reported that their system trust was unaffected by the three transparency levels. Individuals who received a lower grade on their essay than they expected reported reduced system trust, unless the grading algorithm was explained at the medium transparency level. High

[4] http://worrydream.com/LadderOfAbstraction

levels of system transparency, including a paragraph explaining how raw grading scores were algorithmically adjusted, yielded system trust indistinguishable from students experiencing the low transparency condition.

A behavioral measure for the effects of transparency on end users is examined in [18]. The authors performed a series of large-scale, randomized experiments in which participants were shown functionally identical models with two factors hypothesized to influence post-hoc interpretability varied: the number of input features, and model transparency. The algorithm model was a fairly straightforward series of coefficients for each of the model inputs. Understanding of the algorithm was measured by asking participants to predict housing prices, given an algorithmic model. Trust of the algorithm was measured by showing participants the model's prediction and asking for their own prediction of apartment cost. In cases where the conditions were unusual and the algorithm's prediction was inaccurate, such as with a one-bedroom three-bathroom apartment, the authors measured the users' detection of mistakes. Participants shown a transparent model with a small number of input features were better able to simulate the model?s predictions than other conditions. Trust did not significantly vary across conditions. However, the authors' results suggest that increased transparency actually decreased people's ability to detect when a model has made a sizeable mistake, possibly due to cognitive overload. This work shows that in designing explainables it is possible to overwhelm the user with distracting displays of information and thereby counteract the original intention of the explainable.

3 Learning Science Design Process for Explainables

While the design and implementation of explainables for AI may be new, the design and implementation of interactive tutorials to teach content to users is not new. For the foundation of our learning science approach to designing explainables, we began with foundational concepts from educational psychology and the learning sciences. *Backward Design* is a common pedagogical approach detailed in [21] that provides a suggested ordering of tasks when planning instructive content. In Backward Design, the curriculum designer: first, (1) identifies desired results, then (2) determines acceptable evidence, and finally (3) plans learning experiences. This may be considered results-oriented pedagogical design rather than content-oriented pedagogical design. The remainder of this section is decomposed into the steps of the Backward Design process.

3.1 Stage 1: Identifying Desired Results

Desired results imply skills students should be able to perform (i.e., procedural knowledge) or concepts that students should know (i.e., declarative knowledge). Practitioners commonly use a revised Bloom's Taxonomy [13] which suggests a taxonomy of cognitive process with verbs for describing these learning goals:

1. **Remember** (retrieving relevant knowledge from long-term memory): Recognizing, Recalling

2. **Understand** (determining the meaning of instructional messages, including oral, written, and graphic communication): Interpreting, Exemplifying, Classifying, Summarizing, Inferring, Comparing, Explaining
3. **Apply** (carrying out or using a procedure in a given situation): Executing, Implementing
4. **Analyze** (breaking material into its constituent parts and detecting how the parts relate to one another and to an overall structure or purpose): Differentiating, Organizing, Attributing
5. **Evaluate** (making judgments based on criteria and standards): Checking, Critiquing,
6. **Create** (putting elements together to form a novel, coherent whole or make an original product): Generating, Planning, Producing

The learning goals we selected for our BKT explainable were from levels 1, 4, and 5 of this hierarchical taxonomy: (1) *Recall* the definitions of the four parameters in BKT, (2) *Analyze* Predict the impact on mastery prediction when each of the four parameters are increased/decreased individually, and (3) *Critique* unexpected output from the BKT model. Goal 1 was selected as it is necessary to have a basic familiarity with terms and parameters before tackling more complex goals. Goal 2 was selected as a proxy for a basic understanding of the working of the BKT algorithm. Goal 3 was selected as a core skill representing an ability to question the model rather than indiscriminately agreeing with its output. Note that these goals do not require memorization of the BKT formula, nor knowing how the parameters are derived, nor reciting specific flaws of BKT. Our target user are instructors who use BKT-enhanced learning environments in their classrooms, and so reproduction of a complex artificial intelligence algorithm may be unnecessary for our time-constrained users with diverse math backgrounds.

These learning goals are often achieved through component tasks, and a task analysis is often performed to identify these necessary component tasks [10]. There are a variety of task analysis methods: job or performance analysis, learning analysis, cognitive task analysis, content or subject matter analysis, and activity analysis. Cognitive Task Analysis (CTA) is a popular approach used with technologically enhanced learning environments, such as the case study in [16]. For our task analysis, we performed a document analysis, which is one of the first steps of a CTA.

3.2 Stage 2: Determining Acceptable Evidence

Assessments are intended to be derived from the previous stage. That is, each "Desired Result" should have a matching assessment for measuring the successful learning of that result. For this step, one can start with work from the assessment and evaluation communities within pedagogical research. However, depending on the algorithm and the skillset, there may not be the necessary prior work for this assessment. In this case, researchers and instructors must develop their own assessments of the requisite knowledge and skills.

3.3 Stage 3: Planning Learning Experiences

At this stage of Backward Design, the instructional designer must consider what activities will equip students with the needed knowledge and skills to achieve performance goals determined in the previous stage. At this point, a much broader selection of learning science literature becomes relevant, although highly dependent on the learning goals. For this article, we introduce the book, *How Learning Works* [1], as it distills much modern learning science knowledge into seven domain-independent principles to consider when developing instructive materials:

1. Students' prior knowledge can help or hinder learning.
2. How students organize knowledge influences how they learn and apply what they know.
3. Students' motivation determines, directs, and sustains what they do to learn.
4. To develop mastery, students must acquire component skills, practice integrating them, and know when to apply what they have learned.
5. Goal-directed practice coupled with targeted feedback enhances the quality of students' learning.
6. Students' current level of development interacts with the social, emotional, and intellectual climate of the course to impact learning.
7. To become self-directed learners, students must learn to monitor and adjust their approaches to learning.

An additional tool in the learning scientist's toolbox is *Learning By Doing.* [12] showed that interactive activities with feedback in an online course lead to students learning one standard deviation more than students using just informational assets like videos and text. Therefore, if explainable designers wish to provide actual understanding of an algorithm, it is absolutely imperative that explainables provide interactive experiences. It is possible that simply moving a scroll wheel or typing in characters may not be comparable to the generative interactive learning experiences provided with corrective feedback from [12].

4 User-Centered Design Process

While learning science research is informative for the instructional design and content of the explainable, a user-centered design process is helpful for establishing the interaction design. Along this dimension, the main goals of our explainables were to render the artificial intelligence algorithm approachable to non computer scientists in the form of a playful experience. Instructors often do not have significant extra time to dedicate to system understanding, and so one of the main constraints on our prototype development was that the designs be engaging and brief. We are working toward connecting one of these explainables to an existing BKT learning analytics system, in order to provide algorithmic understanding to instructors and students of that system.

4.1 User-Centered Design Method

Our research team applied a user-centered design process to develop our BKT explainable through the below steps. At this stage in the design process, we had three different designs that we evaluated, but the user-centered design process was essentially the same for each design:

1. **Brainstorming & Sketching.** The design was first initialized through individual brainstorming where the goal was to sketch at least nine ideas. Next, individual ideas were explained, refined, and edited in a group brainstorming process.
2. **Low-fidelity Prototyping.** Ideas from the brainstorming sessions were narrowed down to one and construction of low-fidelity paper prototypes began. After several iterations of prototyping within the research team, the paper prototype was pilot-tested on fellow research assistants outside of the research team.
3. **User Pilot Testing.** The prototype was tested with external researchers, and on two different users for each iteration following the process detailed in [19]. Users were instructed to think aloud and explain any difficulty in use or understanding that they encountered while interacting with the prototypes.
4. **Revisions & High-fidelity Prototypes.** The prototype was iteratively adjusted to address user feedback. After the third iteration, the paper prototype was converted into a high-fidelity prototype using click-through prototyping software such as InVision[5].
5. **Implementing the Designs.** Once pilot testing was completed with the high-fidelity prototype, the design was implemented. We then proceeded to run user tests with participants external to our department.
6. **Participant Recruitment.** Upon receiving approval from our Institutional Review Board, we recruited six participants from a small rural town in the northeastern United States via Craigslist, paper flyers, and the local college's online message board. We recruited a range of 6 participants, 3 in the 25-59 age range and 3 older than 25. 4 fo the participants were male. With the consent of each participant, we audio recorded each 30-minute session.
7. **Providing Task Context.** After the first user study, participants were first presented with a brief lesson in a Technologically Enhanced Learning environment, followed by an interactive quiz. The quiz provided immediate feedback on the participants' responses to questions about boxplots, and the final screen displayed a sample dashboard using BKT to predict the participants' mastery of the two boxplot skills. The purpose of this TEL was to demonstrate a context in which BKT is used so participants could better understand the purpose of the explainable.
8. **Usability Testing.** A user test for our explainable typically lasted 10-15 minutes. Once context for the BKT system was completed, participants were shown the explainable and prompted to think aloud [20]. As participants stepped through the explainable, if silence passed, participants were politely asked to continue thinking their thoughts aloud.

[5] https://www.invisionapp.com/

9. **Semi-Structured Interview.** After completing the explainable, we performed a semi-structured interview, asking for the participant's opinions on the format and content before a series of questions evaluating their knowledge retention. These knowledge retention questions were inspired by Bloom's Taxonomy [13] to ensure investigation into varying depths of understanding, with the following example questions:
   (a) *Remember.* What does BKT stand for? What is the purpose of BKT? Can you list and describe the meaning of the 4 parameters used in BKT? What is skill mastery?
   (b) *Understand.* Explain the relationship between the parameters and the skills prediction.
   (c) *Apply.* Explain how BKT works with a new skill of your choosing. How might one increase the probability of mastering a skill in this new context?
   (d) *Analyze.* How do the probabilities of slipping and guessing differ? What is the difference between the probability of already knowing and the probability of learning?
   (e) *Analyze.* Predict what will happen to the mastery prediction if... ...a student answers a problem in/correctly...the probability of initial mastery is increased...the probability of having learned is decreased...the probability of slipping is increased...the probability of forgetting is decreased.
   (f) *Evaluate.* What potential problems or concerns might be brought up by using BKT for predicting when someone has learned a skill?
10. **Affinity Diagramming.** When user testing was complete, a researcher listened to the audio recordings of the interviews and transcribed comments about the explainables onto small pieces of paper resulting in approximately 90 of these raw comments (this includes comments from usability tests on all three of our initial designs). When all user comments were transcribed in this manner, an affinity diagramming process was followed as described in [4]. In traditional user experience design contexts, the affinity diagram represents the scope of the user problem in performing a particular task. This method can be used throughout the user-centered design process to identify user needs or issues in system designs.

4.2 User-Centered Design Results

The results of the affinity diagram formed from the usability tests on our three initial design prototypes suggest common issues for explainable designers to look out for. We identified three overarching categories through the affinity diagramming process: "Consider Individual differences", "Refine the Level of Detail", and "Usability Design Principles".

Consider Individual Differences. This section of our affinity diagram contained often-conflicting comments, which we believe stem from each individual participant's background and personal preferences. These comments covered topics such as comprehension of the tutorial materials. The themes in this category were as follows: “The tutorial taught me something”, “I don’t understand why we’re doing this”, “I don’t understand the metaphor”, “The metaphor worked”, “The second tutorial confused me”, “The second tutorial enlightened me”, and “I don’t know where my knowledge came from”. This category will be the most difficult to address because preferences and prior knowledge vary widely. Our aim is to find a metaphor that works for most people in our target user population. In future testing, our user body will be focused on instructors and students, so that we can ensure the explainable chosen works best for our target populations.

Refine the Level of Detail. This section contained comments surrounding satisfaction and dissatisfaction with the level of detail in our tutorials. The themes in this category were: “The details were helpful”, “There was noticeable cognitive load”, “Too many details make the tutorial overwhelming”, and “Too few details leave more questions”. While the ultimate test of identifying the right level of detail can be identified through measuring participant learning from a pre- to post-test, it is still important to consider user frustration and engagement levels. Creating explainables that are flexible in the amount of detail offered may help test hypotheses about pedagogical content depth for explainables.

Usability Design Principles. This category of our affinity diagram contained comments about the appearance and function of our tutorials, which are typical details unearthed in usability tests. The groups in the Usability Design Principles category were as follows: “It looks pretty and understandable”, “The graphics were helpful”, “The graphics were not helpful”, “There are a few design changes I would make”, “I expected change but didn’t get any”, “Too many buttons makes it easy to mis-click”, “Make the text readable”, “Brief text is not intimidating”, and “Technical terms are intimidating.” These usability issues are the most straight-forward to address. A rational evaluation, such as applying Nielsen’s Heuristics [17] could provide a better initial screening for the usability issues caught in this category going forward.

5 Conclusion

At this stage in the design process we have built an explainable based on a foundational understanding of learning sciences research, and then iteratively evaluated and redesigned the interaction design according to usability testing feedback. However, we have not yet evaluated the explainable for conceptual learning. Our user studies assessed learning during the semi-structured interview stage, but that evaluation should be formalized in a quantitative assessment.

In this article we presented results from our first investigation of designing explanations by example for BKT, a complex algorithm that predicts student knowledge within technologically enhanced learning environments. Through applying user-centered design practices, we found a wide variety of user opinions on our three explainables. By interpreting these results in light of learning science literature, we have produced the following main take-aways for designers of algorithmic explanations by example:

References

1. Ambrose, S.A., Bridges, M.W., DiPietro, M., Lovett, M.C., Norman, M.K.: How learning works: Seven research-based principles for smart teaching. John Wiley & Sons (2010)
2. d Baker, R.S., Corbett, A.T., Aleven, V.: More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In: Intelligent Tutoring Systems. pp. 406–415. Springer (2008)
3. Bassen, J., Howley, I., Fast, E., Mitchell, J., Thille, C.: Oars: exploring instructor analytics for online learning. In: Proceedings of the Fifth Annual ACM Conference on Learning at Scale. p. 55. ACM (2018)
4. Beyer, H., Holtzblatt, K.: Contextual design: defining customer-centered systems. Elsevier (1997)
5. Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User modeling and user-adapted interaction **4**(4), 253–278 (1994)
6. Doroudi, S., Brunskill, E.: The misidentified identifiability problem of bayesian knowledge tracing. In: Proceedings of the 10th International Conference on Educational Data Mining (2017)
7. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning.(feb. 2017). arXiv preprint stat.ML/1702.08608 (2017)
8. Edelson, D.C., Gordin, D.N., Pea, R.D.: Addressing the challenges of inquiry-based learning through technology and curriculum design. Journal of the learning sciences **8**(3-4), 391–450 (1999)
9. Hosanagar, K., Jair, V.: We need transparency in algorithms, but too much can backfire. Harvard Business Review (Jul 2018), https://hbr.org/2018/07/we-need-transparency-in-algorithms-but-too-much-can-backfire
10. Jonassen, D.H., Tessmer, M., Hannum, W.H.: Task analysis methods for instructional design. Routledge (1998)
11. Kizilcec, R.F.: How much information?: Effects of transparency on trust in an algorithmic interface. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. pp. 2390–2395. ACM (2016)
12. Koedinger, K.R., Kim, J., Jia, J.Z., McLaughlin, E.A., Bier, N.L.: Learning is not a spectator sport: Doing is better than watching for learning from a mooc. In: Proceedings of the second (2015) ACM conference on learning@ scale. pp. 111–120. ACM (2015)
13. Krathwohl, D.R.: A revision of bloom's taxonomy: An overview. Theory into practice **41**(4), 212–218 (2002)
14. Lipton, Z.C.: The mythos of model interpretability. Queue **16**(3), 30 (2018)
15. Lovett, M., Meyer, O., Thille, C.: The open learning initiative: Measuring the effectiveness of the oli statistics course in accelerating student learning. Journal of Interactive Media in Education (2008)
16. Lovett, M.C.: Cognitive task analysis in service of intelligent tutoring system design: A case study in statistics. In: International Conference on Intelligent Tutoring Systems. pp. 234–243. Springer (1998)
17. Nielsen, J.: Usability engineering. Elsevier (1994)
18. Poursabzi-Sangdeh, F., Goldstein, D.G., Hofman, J.M., Vaughan, J.W., Wallach, H.: Manipulating and measuring model interpretability. arXiv preprint arXiv:1802.07810 (2018)

19. Snyder, C.: Paper prototyping: The fast and easy way to design and refine user interfaces. Morgan Kaufmann (2003)
20. Van Someren, M.W., Barnard, Y.F., Sandberg, J.A.: The think aloud method: a practical approach to modelling cognitive processes. Londen: Academic Press (1994)
21. Wiggins, G.P., Wiggins, G., McTighe, J.: Understanding by design. Ascd (2005)
22. Yudelson, M.V., Koedinger, K.R., Gordon, G.J.: Individualized bayesian knowledge tracing models. In: International Conference on Artificial Intelligence in Education. pp. 171–180. Springer (2013)