AXIS: Generating Explanations at Scale with Learnersourcing and Machine Learning

Joseph Jay Williams[1] **Juho Kim**[2] **Anna Rafferty**[3] **Samuel Maldonado**[4]
Krzysztof Z. Gajos[1] **Walter S. Lasecki**[5] **Neil Heffernan**[4]

[1]Harvard University
Cambridge, MA
joseph_jay_williams@harvard.edu, kgajos@eecs.harvard.edu

[2]Stanford University & KAIST
Stanford, CA
juhokim@cs.kaist.ac.kr

[3]Carleton College
Northfield, MN
arafferty@carleton.edu

[4]WPI
Worcester, PA
{sjmaldonado,nth}@wpi.edu

[5]Computer Science & Engineering
University of Michigan, Ann Arbor
wlasecki@umich.edu

ABSTRACT

While explanations may help people learn by providing information about why an answer is correct, many problems on online platforms lack high-quality explanations. This paper presents AXIS (Adaptive eXplanation Improvement System), a system for obtaining explanations. AXIS asks learners to generate, revise, and evaluate explanations as they solve a problem, and then uses machine learning to dynamically determine which explanation to present to a future learner, based on previous learners' collective input. Results from a case study deployment and a randomized experiment demonstrate that AXIS elicits and identifies explanations that learners find helpful. Providing explanations from AXIS also objectively enhanced learning, when compared to the default practice where learners solved problems and received answers without explanations. The rated quality and learning benefit of AXIS explanations did not differ from explanations generated by an experienced instructor.

CCS Concepts

•Human-centered computing → Human computer interaction (HCI); •Applied computing → Education; Computer-assisted instruction; Interactive learning environments; Collaborative learning; •Computing methodologies → Sequential decision making;

Author Keywords

Explanation; learning at scale; crowdsourcing; learnersourcing; machine learning; adaptive learning.

INTRODUCTION

Explanations go beyond facts to provide understanding and help people identify principles that generalize to new problems [18, 21]. For example, students learning math frequently memorize how to apply rote procedures to solve problems [8]. With only superficial changes to how problems are described (e.g., which side of the equation the variable x appears on) students may misapply procedures and make mistakes. They cannot draw on conceptual descriptions (explanations) of *why* a procedure works in order to generalize to a broader suite of problems. This problem is exacerbated in online learning and MOOCs, where rapid feedback on answers allows people to game the system [3] and to try many answers until they get it right, without understanding why it is right. Some existing platforms such as intelligent tutoring systems present explanations to learners, which has been shown to enhance learning [1].

However, generating high quality explanations for why those answers are correct is significantly more difficult than providing answers [10]. Instructors have limited time and resources to generate quality explanations for all the problems they create. This means that online learners typically attempt problems and get answers without additional explanations. Even when instructors handcraft explanations — like explanations of how to solve math problems on Khan Academy [khanacademy.org] or ASSISTments [assistments.org] — it is rare for these to be revised over time. This is problematic if an initial explanation suffers from an *expert blind spot* which limits recognition of how students will misunderstand explanations [20]. Online platforms run the risk of scaling the negative effect of poor explanations to thousands of learners.

The challenge we address is how to develop a scalable mechanism to generate and improve explanations for online learning materials. To offload the explanation generation effort from busy instructors or amateur content creators, we turn to learners. Learners are a viable crowd for generating explanations, because they are *experts* in typical misconceptions, directly experiencing the effect of gaps in their knowledge. However, asking individual learners to generate explanations for others is unlikely to be reliable unless one can determine which explanations are most helpful, as many learner-generated explanations may be superficial or even incorrect.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

L@S 2016, April 25 - 26, 2016, Edinburgh, Scotland Uk.

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-3726-7/16/04...$15.00.
DOI: http://dx.doi.org/10.1145/2876034.2876042

In this paper, we present AXIS (Adaptive eXplanation Improvement System), a system that dynamically improves explanations over time as a byproduct of learners' collective interactions with the content. AXIS does this by adding and iteratively refining explanations via a combination of learner prompts to crowdsource explanations and machine learning to choose effective ones. Learners contribute by reporting their level of knowledge, evaluating the quality of explanations generated by others, and adding or refining explanations. Upon analyzing the learner-provided information, machine learning algorithms sift through the changing pool of explanations to identify those that are consistently judged to have highest quality, as rated by learners. The system seamlessly introduces improved versions of explanations to future learners as more learner contributions become available, without requiring manual revision or republication cycles by the instructor. All of these system components are not only designed to improve explanations, but are designed to integrate into learners' interactions with the problems by prompting them to reflect on the information being presented [8, 25].

To evaluate our approach, we recruited 150 participants online from Mechanical Turk and asked them to solve math problems using AXIS. We discuss the design of AXIS and how it was deployed to collect and evaluate explanations. To evaluate the quality of the explanations AXIS collects and the selection policy it discovers, we also report results from a randomized controlled experiment with an independent group of 524 participants. The experiment is designed to measure how the AXIS explanations and policy are judged by learners, and whether they impact learning.

While there were poor explanations generated by learners, the evaluation showed that AXIS was able to identify explanations that many learners rated as helpful. These explanations were also demonstrated to improve learning over the default practice, in which learners simply solve problems and receive unexplained answers. The explanations learnersourced by AXIS even approached those by experienced instructors in terms of perceived benefit and objective learning gains.

Our approach shifts the effort required to create useful learning materials from instructors to the community of learners. It helps both expert instructors, whose time is scarce, and non-expert instructors not trained in formulating effective explanations. Both can use systems like AXIS to generate explanations for instructional materials by leveraging crowds of learners. Additionally, AXIS leverages learners' collective insight into the problem-solving process to help future learners, even when instructor resources are not available, such as in settings where explanations are generated on-the-fly by end users in live sessions [15].

Specifically, this work contributes:

- AXIS, a prototype semi-automated system that instructional designers can add to online problems for which no explanation exists. AXIS engages learners to generate, revise, and evaluate explanations, and provides quality explanations to future learners.
- Results from a study showing that AXIS elicits quality explanations and discovers effective policies for deciding which explanations to provide to learners. We report evidence that people who are solving problems learn more when explanations from AXIS are provided, and that AXIS-curated explanations are as effective for learning as explanations written by an instructor.
- An approach that combines crowdsourcing and machine learning to leverage learners' organic interactions with content, which in turn enhance future learners' experience.

RELATED WORK

AXIS uses crowdsourcing from learners (*learnersourcing*, [11]) to elicit explanations, and machine learning to differentiate among these explanations and determine which are most helpful. We briefly overview other educational systems that have used crowdsourcing and provide background for our machine learning approach.

Crowdsourcing Systems for Education

Prior learnersourcing systems build on successes in designing systems for human computation that satisfy the dual objectives of helping users learn while simultaneously getting useful work done. For example, to provide real-time captions for deaf and hard of hearing users, Lasecki *et al.* ask learners in a classroom to collaboratively caption what they hear using Scribe [16]. Recent work in Massive Open Online Courses (MOOCs) mines traces of MOOC learners' interactions with video to adaptively alter the video interface to highlight sections other students have paid attention to [12]. More active learnersourcing is observed in Crowdy, which embeds prompts for learners to summarize subgoals in sections of an instructional video. While giving learners a useful learning exercise, the system converts the learnersourced summary labels into a browsable text outline for the video [23]. AXIS contributes a novel application to this growing body of research by focusing on learnersourcing explanations that are applicable to a variety of online learning contexts.

Machine Learning for Exploration & Exploitation

Reinforcement learning is a common machine learning technique for situations in which a system must determine which of several actions is best and information about the actions' effectiveness is gathered only by trying an action and observing the results. It has been used successfully in educational applications, including modeling student knowledge [5] and automatically generating hints [4]. Most relevant to AXIS is a subset of reinforcement learning problems known as multi-armed bandit problems, which have been examined in education for choosing sequences of teaching actions [9] and automated experimentation in educational games [17]. Multi-armed bandits are increasingly used in large-scale randomized A/B experimentation by technology companies [13]. This paper frames explanation generation as a multi-armed bandit problem in a large-scale experimentation setting.

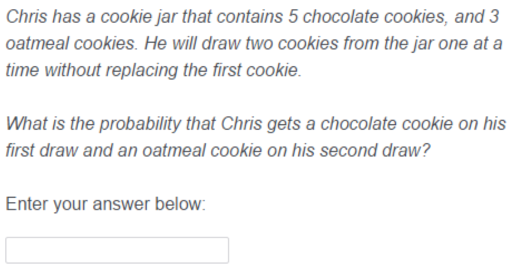
In multi-armed bandit problems, the system repeatedly faces a choice of which action to take and seeks to maximize the total cumulative reward over many repetitions of the choice. In such a problem, there is generally a fixed set of actions (arms), and typically, the system maintains an estimate of the expected reward from taking each action. At each timestep, the system chooses one action and observes the reward from taking that action; over time, the system learns which actions are more effective and thus can earn larger rewards. The key challenge in this type of problem is to balance exploiting the information that has already been gained about the effectiveness of each action and exploring actions where the estimates about their value are still relatively uncertain. For example, imagine a bandit problem with three actions. If each action has only been selected once, with observed rewards of 4, 5, and 6, it probably does not make sense to then only choose the third action at all remaining timesteps. The rewards for each action may be variable, and it could be the case that exploring the first or second actions by choosing them several more times would reveal that they actually produce higher rewards than the third action, on average.

A number of approaches have been proposed for how to select an action at a given timestep based on the evidence observed in the previous timesteps (e.g. [2], [7]). Most approaches combine information about the current estimated expected value of an action and the uncertainty of that estimate, as measured by the variability in observed rewards and the number of times that the action has been selected. Existing methods have been evaluated both theoretically and empirically, with theoretical results [2] making guarantees about asymptotic performance and empirical results helping to illustrate performance given real-world scenarios [7]. In AXIS, we formulate the selection of an explanation as a multi-armed bandit problem where the actions to choose from are explanations generated by learners, and the reward for taking the action of presenting an explanation is the learner's rating of its helpfulness.

AXIS OVERVIEW

Design Goals. While sites like Khan Academy hire hundreds of teachers to produce explanations for their problems, many instructors create online learning materials with far fewer resources. AXIS is aimed at helping these instructors, who often lack the time or experience to create high quality explanations for all of their content but do have access to a large pool of learners. The goal of AXIS is to take a problem and its answer, and then to construct explanations for how to solve this problem, by leveraging the interactions of many learners who solve this problem. AXIS crowdsources production and evaluation of explanations to learners and uses machine learning to analyze this data to identify and deploy the effective explanations.

Core System Components. The two key AXIS components are (1) the learnersourcing interface and (2) the explanation selection policy. The learnersourcing interface collects learning data from learners and their evaluations of explanations, and elicits the generation of new explanations from future learners. The explanation selection policy is used to decide which candidate explanation to present to a new learner. This policy is continually updated based on learners' interactions with the system. The system chooses explanations to present to learners, while the learnersourcing interface prompts them to rate the explanations. A multi-armed bandit algorithm is used to statistically analyze these ratings and update the explanation selection policy. This allows the system to perpetually add new explanations, while dynamically learning which explanations to present, without needing human intervention.

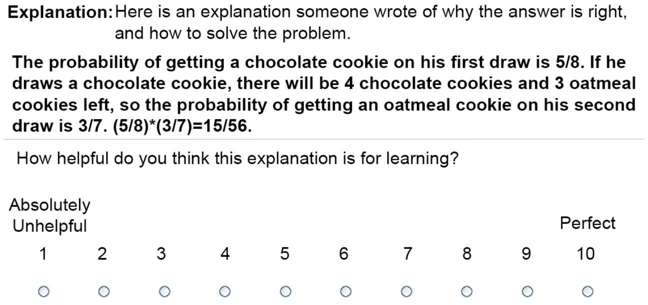


Chris has a cookie jar that contains 5 chocolate cookies, and 3 oatmeal cookies. He will draw two cookies from the jar one at a time without replacing the first cookie.

What is the probability that Chris gets a chocolate cookie on his first draw and an oatmeal cookie on his second draw?

Enter your answer below:

Figure 1. Example of a math problem users might be solving.



Explanation: Here is an explanation someone wrote of why the answer is right, and how to solve the problem.

The probability of getting a chocolate cookie on his first draw is 5/8. If he draws a chocolate cookie, there will be 4 chocolate cookies and 3 oatmeal cookies left, so the probability of getting an oatmeal cookie on his second draw is 3/7. (5/8)*(3/7)=15/56.

How helpful do you think this explanation is for learning?

Absolutely Unhelpful									Perfect
1	2	3	4	5	6	7	8	9	10
○	○	○	○	○	○	○	○	○	○

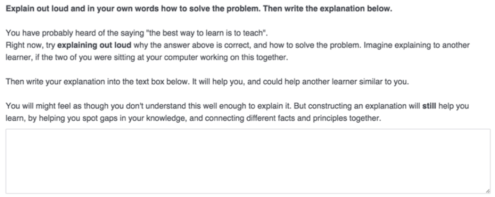
Figure 2. Presentation of explanation to user for learning & rating.

Learnersourcing Interface

A dual goal guides the design of the learnersourcing interface: supporting learners through behavioral science and instructional wisdom, while simultaneously acquiring useful input for computational processes that continually improve the system. Figure 2 shows an example of how the Learnersourcing Interface presents an explanation to a learner of how to solve a problem, *and* prompts them to rate how helpful the explanation is for learning. This data is provided to algorithms in the AXIS backend and used to change which explanations are delivered to future learners. The learnersourcing interface also displays questions that prompt learners to write self-explanations, which existing cognitive and learning sciences research has shown is beneficial for constructing knowledge [1, 8, 25]. At the same time, learners' explanations can be useful to other learners, if they are added to the system pool.

Explanation Selection Policy

AXIS provides learners with explanations of how to solve problems. Soliciting explanations from learners addresses the problem of scalable creation of explanations for a large and potentially growing database of activities. But it introduces a new challenge: how can we reliably determine which explanations are effective for helping new users, without instructional designers expending significant time vetting contributions? We address this challenge by formulating the problem of selecting explanations as a multi-armed bandit problem.



Explain out loud and in your own words how to solve the problem. Then write the explanation below.

You have probably heard of the saying "the best way to learn is to teach".
Right now, try **explaining out loud** why the answer above is correct, and how to solve the problem. Imagine explaining to another learner, if the two of you were sitting at your computer working on this together.

Then write your explanation into the text box below. It will help you, and could help another learner similar to you.

You will might feel as though you don't understand this well enough to explain it. But constructing an explanation will **still** help you learn, by helping you spot gaps in your knowledge, and connecting different facts and principles together.

Figure 3. Self-explanation prompt for learner to write an explanation for why the given answer to a math problem is correct.

Multi-armed bandit problems require a system to repeatedly select an action, and to learn which action is most effective, based on observing the non-deterministic results. This is exactly the scenario that AXIS faces: each problem can be viewed as a different multi-armed bandit. When a new user is introduced to a problem, the system must choose which explanation to show to the user. The explanations are thus different action choices. After the explanation has been given to the user, we must measure how effective it was; this is the observed reward in the bandit formulation. In the case of an educational system, this might correspond to having users provide feedback on how much the explanation helped them learn. Other reward signals can be used, and in our future directions we consider accuracy on subsequent problems. In the current system deployment, the algorithm aimed to optimize for learners' ratings of the helpfulness of an explanation because it is a direct function of the actions AXIS is deciding between: which learnersourced explanation to present. Although we seek explanations that teach well enough that the learner gets the next problem correct, this variable is noisy and influenced by many variables outside system control.

By framing explanation selection as a multi-armed bandit problem, we can draw on the existing literature for an algorithm that addresses the problem of exploitation (presenting explanations that have been observed to be relatively effective) versus exploration (experimenting with different explanations to gain more evidence about their effectiveness). We use Thompson sampling, a Bayesian algorithm that has been shown to have near-optimal regret bounds and performs well on practical problems [7]. Other bandit algorithms may also have been effective, but Thompson sampling has advantages for future work with instructors, because it facilitates interpretable representations of the system's beliefs at any point in time. We can intuitively capture both estimates about explanations' effectiveness and the algorithm's uncertainty about those estimates.

Like most bandit algorithms, Thompson sampling provides a dynamic policy for choosing which explanation to give a new user, and an algorithm for incorporating new information to update this policy based on observing the reward after an explanation has been selected. Thompson sampling stores an estimated distribution for the reward for each explanation. This distribution indicates both the expected reward from choosing a particular explanation, and how variable the reward is. Both of these aspects can impact what action we wish to select. The parameters of each distribution are initially set based on a prior, which intuitively indicates our beliefs about the effectiveness of explanations that have not yet been presented to any users, and then are updated based on the likelihood of the observed evidence.

AXIS's beliefs about the value of each explanation are represented using a Beta distribution. The prior for this distribution is also a Beta distribution, and the likelihood is a Bernoulli distribution. The posterior is then proportional to the product of the prior and the likelihood, with the likelihood updated after each reward observation; this update is easy to implement because the Beta and Bernoulli distributions are conjugate. In AXIS, explanations are added by learnersourcing. AXIS uses a filtering mechanism, only adding explanations to the system pool when: the explanation is above a minimum character length, the explainer displays above average knowledge about how to solve this type of problem, and the explainer rates her explanation as likely to be helpful to other learners. Explanations that meet these criteria are added as new arms to the bandit for the problem. The prior distribution for their reward or expected rating follows a Beta(19, 1) distribution, which expresses beliefs analogous to having seen the explanation get rated a 9 and a 10. Intuitively, this distribution reflects a great deal of optimism about how helpful new explanations will be – the expected rating is 9.5 out of 10. But at the same time the prior will be rapidly updated, as these highly optimistic beliefs are based on the equivalent of just two observations. This prior encourages the algorithms to collect data about new explanations, as discussed below.

After an explanation has been chosen and displayed to the user, we use the user's rating of its effectiveness (shown in Figure 2 as the observed reward. In order to allow the same infrastructure to be used for a binary reward signal as for the rating reward signal, we treat each action as adding 10 total observations of a Bernoulli variable. The number of successes is the user's rating of the explanation's effectiveness (on a 10-point scale where 10 is maximally helpful), and the number of failures is ten minus the number of successes. The update to a posterior that is $\text{Beta}(x,y)$ is simply $\text{Beta}(x + \text{number of new successes}, y + \text{number of new failures})$.

As an example, consider a learnersourced explanation that has been rated as five out of ten by each of the first two subsequent learners who viewed the explanation. While the initial expected rating for this explanation was high, due to the $\text{Beta}(19,1)$ prior, this distribution also has significant uncertainty. This means that ratings by even a few learners have a large influence on this expected rating. To incorporate these two ratings of five out of ten, the Beta distribution is updated as described above, resulting in $\text{Beta}(29,11)$ as the posterior distribution. The expected rating indicated by this distribution is only 7.25. Thus, the prior indicates a high expected rating early on, encouraging the algorithm to use the explanation, but the uncertainty in the prior means that collected explanations quickly dominate the expected value of the posterior.

So far, we have described how Thompson sampling represents the observed data about each explanation and how this representation is updated based on new observations. The final component of Thompson sampling is its policy: how to select an appropriate explanation for a new user. Thompson sampling selects the explanation that satisfies $\text{argmax}_{e \in \text{explanations}} E[reward|\theta_e]p(\theta_e|D)$, where D is the set of observed data and θ_e is the parameters of the Beta distribution for this explanation. That is, it chooses the explanation that has highest expected value, taking into account the uncertainty we have about the distribution of rewards from this explanation. This corresponds to selecting each explanation in proportion to the probability that it is the best explanation, given the priors and observed data. Implemented via highly efficient sampling, such a policy balances exploration and exploitation by incorporating uncertainty about the underlying distribution.

The probabilistic policies for multi-armed bandits have several advantages over more obvious methods, like presenting the highest rated explanation. Apparently simpler methods raise many questions. For example, if AXIS used ranking, how many good ratings would a new learnersourced explanation have to receive for AXIS to identify it as the current best? Instead of choosing an arbitrary heuristic (5? 10?), this question can be answered in a principled way by capturing uncertainty in the probability distributions used in Thompson Sampling. These allow AXIS to encode beliefs about how noisy learner ratings are, by defining the likelihood of broad versus narrow ranges of ratings. Does the risk of showing students a poor explanation outweigh the value of getting an explanation that is 10% better than the best? Multi-armed bandits provide an extensively studied formal model for answering questions about balancing exploitation–giving explanations known to help– against exploration– trying out new explanations that may turn out to be bad or good.

Implementation

Our goal in designing AXIS was for intelligent web apps to be easily duplicated and shared to enable end-user programing [19] for online educational resources like websites, lessons, problems, and quizzes. User groups like instructors rarely manage servers and write code, value support in automating some features of instruction, and wish to maintain discretion and control over learning materials. Our mashup integration for implementing systems like AXIS combines (more or less) freely available web resources that bridge easy-to-use features like WYSIWYG with underlying programming languages and flexible APIs. The interface for presenting and collecting information was created using Qualtrics, an advanced survey software that most universities have an unlimited license to. The machine learning algorithm was written, hosted and deployed using the Apps Script functionality in Google Spreadsheets. Code using Javascript libraries received data from the Qualtrics API every time a learner interacted with the AXIS front-end, made this data available for display and manipulation in a Google Spreadsheet, implemented Thompson Sampling to analyze the data and update the policy after each user, and sent instructions via the Qualtrics API as to which explanations to present. A key consideration in the choice of these resources, despite their many technical limitations, was availability to end-users. The combination of Qualtrics and Google Spreadsheets/Apps Script allows those without programming knowledge to obtain, host, modify, deploy, and share customized intelligent educational agents that run machine learning algorithms on request. Access to the resources we've created can be requested via `http://tiny.cc/useaxis`.

AXIS CASE STUDY: GENERATING EXPLANATIONS FOR SOLVING MATH PROBLEMS

We deployed and tested AXIS in the context of providing explanations to learners solving math problems. The target user in our case study was an online instructional designer overseeing online math problems for ASSISTments [`assistments.org`], a math platform similar to Khan Academy. This platform has a content library of over 500 math problems, hundreds of which do not have explanations. The instructional designer wanted a way to generate explanations to present to learners, but she had not had much success in relying on work-study undergraduate students to do so. We identified four math problems that she had already written explanations for, as it would allow us all to see how close the output of AXIS could get to explanations she had already created. The problems covered algebra, expressions, and probability, at a level appropriate to both middle schoolers and adults. Before implementing AXIS with students in classrooms, she wanted to see evidence that AXIS could successfully cull explanations from untrained people.

The next section explains how we implemented and deployed AXIS with 150 study participants solving the four math problems. Our evaluation was done in two stages: in the first stage, we describe the explanations AXIS collects and how the policy changes over time; in the second stage, we report results from a randomized experiment. This experiment investigates recruits an independent group of participants to investigate how their perceptions and success in learning are influenced by different components of the AXIS explanation pool and policy.

Methods: AXIS Implementation & Deployment

Participants

The deployment case study was conducted with 150 people residing in the US who were recruited online to participate in an education research study, via Amazon Mechanical Turk. Each task paid $3.50 for the 40 minute study. 150 participants roughly matches the number of students learning a math topic at a typical middle school, and the size of a large introductory university course.

Understanding these 150 participants' baseline level of knowledge is useful for interpreting results from AXIS. Participants gave a subjective rating of their relevant school and work experience for solving each problem, as a percentile of the general population. 25.0% of participants rated themselves as being in the bottom quartile (0th to 25th percentile), 41.7% in the second quartile, 28.6% in the third quartile, and only 4.7% rated themselves in the top quartile (75th-100th percentile).

An objective measure of learning was also available from whether their answers to the problems were correct or incorrect. 13.3% of participants had accuracy between 0 and 0.25, 20.0% accuracy of 0.25–0.50, 19.1% accuracy of 0.50–0.75, and 47.5% accuracy of 0.75–1.00.

Additional demographic information was not collected, although it should be in future research. We anticipate that the trends will match typical distributions on Amazon Mechanical Turk. For example, [6] found that the population of workers on MTurk is similar to the general US population, albeit

	Explanation	Explanation Rating
Learner Explanation AXIS Discarded via Filtering Rule	It is three over seven because after the chocolate cookie has been removed there are 7 cookies in the jar, leaving 3 oatmeal cookies remaining.	5.2
Early Stage AXIS	go based on the amount of cookies that are available and run a trial until the chocolate cookie is picked out, then do the same for oatmeal	4.2
Later Stage AXIS	When you have 8 cookies in the jar and 5 are chocolate you have a 5/8 chance of the cookie you draw being chocolate. When there are 7 cookies in the jar and 3 are oatmeal you have a 3/7 chance of drawing the oatmeal cookie. To get the overall probability you need to multiply 5/8 by 3/7 which results in overall probability of 15/56	6.8
Written by Instructional Designer	The total number of cookies in the jar is 8. Since there are 5 chocolate cookies the probability that Chris gets an chocolate cookie is 5/8 Since Chris removed 1 cookie from the jar and did not replace it or put it back there are now 7 cookies in the jar. So, the probability that Chris gets an oatmeal cookie from the jar is 3/7 5/8 x 3/7 = 15/56 So, the probability of Chris getting a chocolate cookie on the first draw, and an oatmeal cookie on the second draw is 15/56 Type in 15/56	7.7

Figure 4. Examples of explanations for one of the problems that AXIS was deployed for. After deployment, we conducted an independent evaluation study with new users to evaluate explanations from AXIS and other sources. The explanations were included in the evaluation study, and the mean helpfulness ratings are shown in the second row.

slightly younger (M = 32.3), more educated (M = 14.9 years of education), and more female (60.1%).

Procedure and System Configuration

All participants worked on the four math problems in a random order. For each problem, after entering an answer, they were told the correct answer. AXIS would then displayed an explanation for why the answer was right (chosen by the explanation selection policy) and/or a prompt for learners to explain to themselves why the answer was right. Figure 3 shows this prompt, which emphasized the value of explaining as a way to help the learner to understand more deeply. At first the explanation pool was empty, so learners would instead see only the self-explanation prompt. We defined an *AXIS Filtering Rule* to automatically discard explanations that were unlikely to be helpful to others. Specifically, AXIS added a learner's explanation to the explanation pool only if it was longer than 60 characters, the learner rated herself as having above average knowledge of how to solve problems like the current one, and the learner rated the likelihood of the explanation helping another learner as higher than 6, on a scale from 1 (Zero Chance) to 10 (Absolutely Likely). Once added to a problem's explanation pool, the explanation would be probabilistically selected for presentation to future learners working on the problem, based on how highly it had been rated whenever presented. A separate explanation pool and policy was maintained for the explanations in each of the four problems.

Results: Description of AXIS Explanation and Policy

Adding Learnersourced Explanations to the Pool

By interacting with AXIS, 150 learners generated between 60 and 72 explanations for each of the four problems. The AXIS Filtering Rule added 12, 9, 12, and 12 of the learner-sourced explanations to the pools for the 4 problems. Figure 4 illustrates the explanations that learners generated, and how AXIS processed them. The *Discarded using Filtering Rule* explanation generated by a learner was discarded because it did not meet the AXIS Filtering Rule. The *Early Stage AXIS* explanation was added to the pool via the filtering rule, but analysis of its ratings by the selection policy resulted in its probabilistic phasing out – continually decreasing probability of being sampled for a new learner. In contrast, the selection policy has identified the *Later Stage AXIS* explanation as one of the highest rated, with a higher probability of being sampled for users. For comparison is the explanation our ASSISTments instructional designer wrote for this problem. The Explanation Rating column provides mean helpfulness ratings that were collected in the experiment we conducted to evaluate AXIS.

Dynamic Evolution of AXIS Policy

Once explanations are collected from the learnersourcing interface, AXIS automatically analyzes each new learner's ratings of how helpful an explanation was for learning, and immediately updates the probabilistic policy for which explanations should be presented to the next learner. As an illustration, we examine the policy for one of the four problems, the Compound-Probability problem. Figure 5 shows two snapshots of how this policy dynamically varied as more learners used the system. The policy for this problem can be represented as a probability distribution over the ten explanations AXIS selected for presentation. Figure 5 shows the AXIS probabilistic policy for determining which explanation would be seen by the 76th learner (after the first 75 learners) and the policy for the 151st learner (after all 150 AXIS learners).

This illustrates a challenge with evaluating the explanations while the AXIS system is changing dynamically. To evaluate the AXIS explanations and policies we randomized their presentation to a new group of 524 people, collecting ratings of explanations, along with subjective and objective measures of whether these explanations influenced learning.

EVALUATION OF AXIS EXPLANATIONS AND POLICY

To evaluate whether the AXIS system was able to collect and identify *useful* explanations from learners, it is necessary to determine if AXIS successfully picks explanations that are helpful to future learners and discards ones that are not. Our experiment compared the quality of the explanations selected by AXIS to explanations AXIS filtered out and discarded, and to the original explanations that the ASSISTments instructional designer had written for the problems. An ambitious secondary goal of the experiment was to investigate whether these learnersourced explanations could impact learning.

Methods

Participants

The randomized experiment recruited 524 new people to participate in a HIT posted on Amazon Mechanical Turk. Each HIT paid $3.50 for the 40 minute study.

Procedure

The study consisted of a learning phase in which participants solved the four problems and provided ratings for explanations,

	AXIS Index for Explanation	1	2	3	4	5	6	7	8	9	10
AXIS Policy for Presenting Explanations: Probability of each explanation being presented.	Policy for 76th learner (after 75 learners)	0.36	0.34	0.31	0	0	0	0	0	0	0
	Policy for learner 151 (after 150 learners)	0.1	0.1	0.09	0.1	0.09	0.1	0.1	0.09	0.11	0.11
Outcomes of AXIS Policy	Explanation Rating	3.97	3.85	5.36	7.97	5.5	6.97	8.53	8.16	3.21	7.29
	Increase in Perceived Skill at Solving Problems (on a 1 to 10 scale)	-0.63	-0.91	-0.28	0.55	-0.06	0.47	0.81	1.71	-1.71	0.29
	Learning Gains	-0.01	0.06	0.16	0.02	-0.10	-0.01	0.03	0.27	-0.04	-0.04

Figure 5. AXIS Policy for the explanation pool for one of the four problems. The policy's probability distribution over the ten explanations that were added to the pool during deployment is shown after 75 learners (row 2) and after 150 learners (row 3).

followed by an assessment phase where they had to solve twelve problems without being given any feedback.

Learning Phase. In this phase, participants were randomly assigned to a number of different conditions, in order to evaluate a wide range of explanations. One condition was solving *Problems with answers only*, standard practice for many online problems without explanations. The other conditions all included explanations that were displayed after seeing the correct answer to a problem. For any one of the four problems, participants were randomly assigned to one of many conditions. They could see one of the explanations from the AXIS pool for that problem, an explanation that was *Filtered out by AXIS* and not presented, or an original explanation *Written by Instructional Designer* at ASSISTments. This provided two important comparisons to AXIS explanations and policy; a lower bound in the form of explanations AXIS filtered out and did not present, and an upper bound in the form of the high quality explanations written by the original instructor.

Participants were prompted to rate how helpful these explanations were for learning, on a scale from 1 (Not Helpful At All) to 10 (Extremely Helpful). Moreover, participants were asked to indicate how likely they were to solve future problems like the one they were working with. They made this rating on a scale from 1 (Zero Chance) to 10 (Absolutely Certain). By comparing these ratings before and after learners received different explanations, we had a more direct measure of the impact of different explanations on people's learning.

Assessment Phase. The ideal outcome for the learnersourced explanations is impact on objective behavioral measures of learning, especially *transfer* of knowledge to novel problems. The learning phase was followed by problems designed to assess whether participants learned from particular explanations. For each of the four original problems, there was an *isomorphic* problem where only the numbers and surface details (e.g., names in an expression) were changed. To measure *transfer* of the knowledge gained from explanations, participants were provided with two problems that were novel but tested the same topic as the original problem (e.g., compound probability, using variables in algebraic expressions).

Results: Usefulness of Explanations for Learners

Figure 6 shows data about the effects of providing explanations while people solved math problems. These diverse measures ranged over rating explanations, subjective judgments about solving future problems, and objective measures of accuracy on novel problems. To investigate the benefits of explanations from AXIS, we used randomized comparisons to explanations (or lack thereof) from a range of sources. Participants were randomly assigned to see: No explanation (original problems), AXIS explanations, learner explanations discarded by AXIS filtering rule, or explanations written by the instructional designer. All our reported analyses used linear mixed–effect models, including a fixed factor representing which explanations were provided. This factor had a significant effect in all analyses conducted, all ps < 0.05. Problem type (since four different probability and algebra problems were used) was a within-subjects variable that was incorporated as a random effect. All statistics reported in this section concern pairwise comparisons conducted within the mixed-effects model, unless otherwise stated.

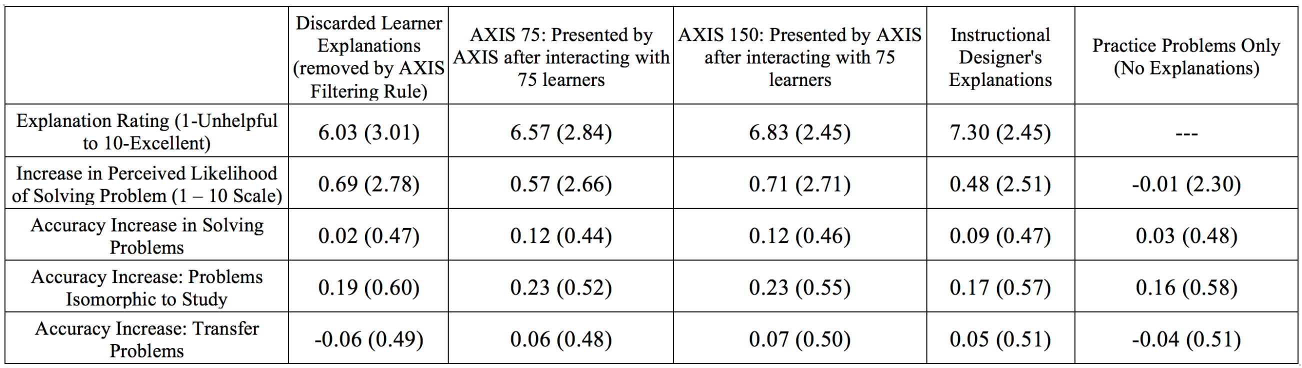
To evaluate the effectiveness of the AXIS policy at a particular point in time, we must consider both how good each explanation is, and how likely it the explanation is to be shown under the current policy. The evaluation experiment randomized presentation of every explanation in the AXIS pool to learners, to assess its impact on their behavior and their perceptions of its helpfulness. For example, to quantify the overall *helpfulness* of AXIS at timestep 150, we compute a weighted average of the helpfulness of all the explanations in the pool at that time, where the weights were determined by the probabilistic policy. This builds on the approach in [7] for assessing the quality of a bandit's policy. We also computed measures of the benefits of the AXIS explanation pool and policy after 75 learners (*AXIS-75*), a subset of the AXIS explanation pool and policy after 150 learners (*AXIS-150*). This data is shown in Figure 6 to provide a qualitative snapshot of how AXIS was changing over time.

Rated Quality of AXIS Explanations

The learnersourced explanations AXIS presented were rated as significantly more helpful for learning than the explanations removed by the filtering rule ($M = 6.83$ vs. 6.03, $SE = 0.28$, $p < 0.01$).This provides evidence for a reliable improvement over learnersourced explanations that were not screened and optimized by AXIS.

Increase in Perceived Skill at Solving Problems

For each problem in the learning phase, participants were asked to rate how likely it was that they could solve problems like it without any help. They responded on a scale from 1 (Zero Chance) to 10 (Absolutely Certain). After attempting the problem, seeing the answer, and (depending on condition)



	Discarded Learner Explanations (removed by AXIS Filtering Rule)	AXIS 75: Presented by AXIS after interacting with 75 learners	AXIS 150: Presented by AXIS after interacting with 75 learners	Instructional Designer's Explanations	Practice Problems Only (No Explanations)
Explanation Rating (1-Unhelpful to 10-Excellent)	6.03 (3.01)	6.57 (2.84)	6.83 (2.45)	7.30 (2.45)	---
Increase in Perceived Likelihood of Solving Problem (1 – 10 Scale)	0.69 (2.78)	0.57 (2.66)	0.71 (2.71)	0.48 (2.51)	-0.01 (2.30)
Accuracy Increase in Solving Problems	0.02 (0.47)	0.12 (0.44)	0.12 (0.46)	0.09 (0.47)	0.03 (0.48)
Accuracy Increase: Problems Isomorphic to Study	0.19 (0.60)	0.23 (0.52)	0.23 (0.55)	0.17 (0.57)	0.16 (0.58)
Accuracy Increase: Transfer Problems	-0.06 (0.49)	0.06 (0.48)	0.07 (0.50)	0.05 (0.51)	-0.04 (0.51)

Figure 6. Data from evaluation experiment about effects of different (or no) explanations, reflected in means for: Subjective Rating of Explanation helpfulness for learning; increase in Self-Reported Skill at solving problems; and increase in objective Accuracy in solving problems.

interacting with explanations, the next page showed them the problem and asked them to make the judgment again about their likelihood of solving it. We analyzed the increase from before to after the problem as a measure of how much different explanations resulted in learners perceiving that they would be better able to solve future problems. Figure 6 shows these values in the third row.

Learners who received the AXIS-150 explanations were more likely to experience increases in their expectation that they could solve future problems, when compared to those learners simply practicing problems without explanations. ($M = 0.71$ vs. -0.01, $SE = 0.13$, $p < 0.001$). There was no significant difference in learners' beliefs about being better able to solve problems, whether they received the AXIS explanations or those written by the ASSISTments instructional designer (M =0.71 vs 0.48, $SE = 0.23$, p = 0.14).

Learning Gains in Accurately Solving Problems

The most ambitious test of AXIS is whether it provides explanations to learners that measurably increase their success in solving problems. Participants might report that explanations from other learners were helpful, and even feel a sense of understanding and capacity for solving problems. But these explanations could still fail to produce any actual learning or lasting acquisition of knowledge. Participants' accuracy in solving the four problems in the learning phase was used as a baseline of knowledge, and Figure 6 shows the overall increase in accuracy from the learning to assessment phase, as a result of receiving AXIS explanations, explanations from other sources, or no explanations.

In fact, AXIS explanations did not merely have subjective benefits. Participants were significantly more likely to solve future problems after receiving AXIS explanations, when compared to practicing of problems. A pairwise comparison within the mixed-effect model revealed a significant increase in accuracy from the initial problems to the assessment problems, M = 12% versus just 2.7%, $SE = 0.027$, $p < 0.05$.

Of course, it might seem obvious in hindsight that providing *any* explanation will increase learning and success on future problems. However, this was not the case. The learnersourced explanations AXIS discarded did not provide any learning benefits beyond normal practice of math problems (M = 2% vs 3%, $p = 0.86$). Simply providing explanations was not sufficient to support learning, and these explanations were significantly less beneficial for learning than explanations delivered by the AXIS policy (M = 12% vs. 2%, $SE = 0.04$, $p < 0.029$).

The AXIS explanations also increased success in solving novel transfer problems that required going beyond the explicit information in the explanation (differences of 9-12%, $SE = 0.03$, 0.04, $p < 0.01$). Overall, it was encouraging that there were no significant differences between learnersourced explanations curated by AXIS, and the explanations written by the ASSISTments instructional designer herself (all ps > 0.30).

QUALITATIVE RESULTS

Instructional Designer's Perspective

Our hope is that AXIS makes it easy for instructors to add a plugin to problems and educational content that will build a pool of explanations, and automatically learn which ones to present. We conducted a 30 minute semi-structured interview with the instructional designer at ASSISTments to show her several of the AXIS explanations. She said that the top-rated AXIS learnersourced explanations were comparable to the explanations she had written, and were of sufficient quality to deploy to the middle school students currently using ASSISTments. She admitted a natural preference for the explanations she herself had written, but believed the quality was sufficiently similar to the AXIS learnersourced explanations, that the best test to discriminate them would be actually comparing their effect on student learning. She was surprised but pleased that the learning benefits were comparable in our evaluation experiment. She also commented on how the AXIS plugin could be used more generally than explanations for math problems, since textual explanations can fit on any webpage in a course, or take the form of motivational messages and tips for learning.

Currently we are building a Learning Technologies Interoperability (LTI)–compliant connector that would allow AXIS to be embedded within ASSISTments and all LTI–compliant MOOC platforms and on-campus Learning Management Systems. This currently includes Coursera, EdX, Moodle, and Canvas.

Moreover, while AXIS does not require manual intervention by instructors for the system to run, it is designed to enable an instructor to interact with the explanation pool and algorithm at any point, through examining data or looking at explanations in the Google Spreadsheet. By typing into cells of the Google Spreadsheet, instructors can freeze and override AXIS's policy changes, or set the policy manually. For future research we will explore using interactive machine learning to allow instructors to work cooperatively with AXIS by adding their own explanations, and adjusting weights for explanations according to their opinion, by typing different prior probabilities. AXIS could also be used in conjunction with systems for automatic generation of educational content beyond explanations, such as hints [4].

Learner's Perspective

The reciprocity of help might encourage learners to participate in learnersourcing explanations, even when individual benefits are not apparent at the time of contribution. By providing explanations that help future learners, learners know they will sometimes benefit from explanations previous learners have provided for them. In addition, we took a step further to qualitatively explore direct benefits to learners in our deployment. After using the system they answered an online form with open-ended questions about what they thought of the learning activities, how they felt about writing explanations, and what they found helpful for learning. Some learners candidly stated that explaining wasn't helpful, or that they did not bother to explain: "I didn't write explanations because I don't think I could get it down on paper."

A substantial number acknowledged the challenge in writing explanations ("While it can sometimes be a bit frustrating") but were pleasantly surprised by the value: "It lets you really understand the logic behind it so you are more able to solve similar problems," and "Talking it out really helps. I will try and use that strategy for other problems besides math."

DISCUSSION AND LIMITATIONS

While AXIS and its evaluation showed some initial promise in generating explanations, there are limitations to the current system and the evaluation methods. AXIS focuses on presenting a single best explanation to all learners, agnostic to their level of knowledge and preference. Personalization of different explanations to different profiles of learners was not explored. Future versions of AXIS could use the learnersourcing interface to elicit information that could be used to personalize delivery of explanations across learners– like study preferences or current state of confusion. AXIS could then implement Thompson Sampling for *contextual* multi-armed bandits, in which the reward of an action depends on a context vector of side information [14]. The reward of an explanation would therefore depend on a set of variables about the user. A second limitation was that our participants were paid crowd workers on Mechanical Turk. While these workers share more demographic features with online learners than convenience samples in typical laboratory studies, future deployments will embed AXIS within platforms like ASSISTments and edX to help authentic students, who may be less (or more) motivated.

There are clear limitations to having AXIS optimize for a reward signal like learners' subjective ratings of explanation quality. We chose to have AXIS select explanations based on their ratings rather than accuracy on subsequent problems because: it was continuous rather than binary, immediately available, and arguably less influenced by factors extraneous to the explanation. However, an extensive literature reveals learners' failures in metacognitive awareness of what they do and do not know, such as the *illusion of explanatory depth* [22] reveal people's great surprise at their erroneous assumptions about being able to provide detailed explanations. While this version is limited to explanation rating, a general strength of AXIS is that it allows instructors to set variables that the multi-armed bandit takes actions to optimize. Future work can explore reward variables like performance on quizzes, continued persistence, or even attitudes towards learning, by varying which versions of explanations and other educational content are presented. The underlying approach AXIS takes in using machine learning to do automatic and real-time optimization of educational content should also be generalized beyond its current application to explanations for solving problems.

The current results do not shed light on many design choices, like knowing when sufficiently many explanations have been collected. Future research can investigate these and other issues, like which filtering rules should be used to add explanations to the pool. This paper had the narrower aim of describing the first implementation of the AXIS system, and evaluating whether it was even effective with the small samples of 75–150 that are typical in larger university courses, and K12 classes.

We chose to separate the evaluation from the deployment phase to provide a rigorous assessment of AXIS performance even when limited to a crowd of 75–150 learners. This allowed for greater statistical power, increasing from 150 participants in deployment to 524 in evaluation. Even with 524 participants, any individual explanation from AXIS was only seen by an average of 30 people. The evaluation study also included additional extensive measures of learning that would have been onerous in the system deployment. However, when thousands of participants are easily available, future system deployments can build on the current work to integrate deployment for practical use with *in vivo* evaluation.

CONCLUSION AND FUTURE WORK

Generating explanations for a large number of online learning materials requires significant time and effort from instructors. In this paper, we present an alternative model that engages learners to help generate and refine explanations. AXIS combines techniques from crowdsourcing and machine learning to achieve this goal. In an experiment with math problems, AXIS successfully led learners to produce quality explanations that helped improve the learning of future users.

While we focused on explanations to math problems in this paper, our approach can generalize to producing and improving explanations for other types of online learning content: adding *why* information to how-to instructions that teach procedural skills, adding more illustrative examples in a learning material, or clarifying task instructions on online workplaces (e.g., Mechanical Turk) to improve worker understanding and success. Any instructor or researcher can register interest in collaboration or access to AXIS via `http://tiny.cc/useaxis`, or build adaptive online resources using the *MOOClet* formalism [24] we used to implement AXIS.

REFERENCES

1. Vincent AWMM Aleven and Kenneth R Koedinger. 2002. An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive science* 26, 2 (2002), 147–179.

2. Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47, 2-3 (2002), 235–256.

3. Ryan Shaun Baker, Albert T Corbett, Kenneth R Koedinger, and Angela Z Wagner. 2004. Off-task behavior in the cognitive tutor classroom: when students game the system. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 383–390.

4. Tiffany Barnes and John Stamper. 2008. Toward automatic hint generation for logic proof tutoring using historical student data. In *Intelligent Tutoring Systems*. Springer, 373–382.

5. Joseph E Beck and Beverly Park Woolf. 2000. High-level student modeling with machine learning. In *Intelligent tutoring systems*. Springer, 584–593.

6. Adam J Berinsky, Gregory A Huber, and Gabriel S Lenz. 2012. Evaluating online labor markets for experimental research: Amazon. com's Mechanical Turk. *Political Analysis* 20, 3 (2012), 351–368.

7. Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*. 2249–2257.

8. Michelene TH Chi, Miriam Bassok, Matthew W Lewis, Peter Reimann, and Robert Glaser. 1989. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive science* 13, 2 (1989), 145–182.

9. Benjamin Clement, Pierre-Yves Oudeyer, Didier Roy, and Manuel Lopes. 2014. Online optimization of teaching sequences with multi-armed bandits. In *Educational Data Mining 2014*.

10. Arthur C Graesser, Patrick Chipman, Brian C Haynes, and Andrew Olney. 2005. AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. *Education, IEEE Transactions on* 48, 4 (2005), 612–618.

11. Juho Kim. 2015. *Learnersourcing: Improving Learning with Collective Learner Activity*. Ph.D. Dissertation. Massachusetts Institute of Technology.

12. Juho Kim, Philip J. Guo, Carrie J. Cai, Shang-Wen (Daniel) Li, Krzysztof Z. Gajos, and Robert C. Miller. 2014. Data-driven Interaction Techniques for Improving Navigation of Educational Videos. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. 563–572.

13. Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M Henne. 2009. Controlled experiments on the web: survey and practical guide. *Data mining and knowledge discovery* 18, 1 (2009), 140–181.

14. John Langford and Tong Zhang. 2008. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*. 817–824.

15. Walter S. Lasecki, Juho Kim, Nick Rafter, Onkur Sen, Jeffrey P. Bigham, and Michael S. Bernstein. 2015. Apparition: Crowdsourced User Interfaces That Come to Life As You Sketch Them. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1925–1934.

16. Walter S. Lasecki, Christopher Miller, Adam Sadilek, Andrew Abumoussa, Donato Borrello, Raja Kushalnagar, and Jeffrey Bigham. 2012. Real-time Captioning by Groups of Non-experts. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 23–34.

17. Yun-En Liu, Travis Mandel, Emma Brunskill, and Zoran Popovic. 2014. Trading Off Scientific Knowledge and User Learning with Multi-Armed Bandits. In *Educational Data Mining 2014*.

18. Tania Lombrozo. 2006. The structure and function of explanations. *Trends in cognitive sciences* 10, 10 (2006), 464–470.

19. Brad A Myers. 1995. User interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)* 2, 1 (1995), 64–103.

20. Mitchell J Nathan, Kenneth R Koedinger, and Martha W Alibali. 2001. Expert blind spot: When content knowledge eclipses pedagogical content knowledge. In *Proceedings of the Third International Conference on Cognitive Science*. Citeseer, 644–648.

21. Alexander Renkl. 1997. Learning from worked-out examples: A study on individual differences. *Cognitive science* 21, 1 (1997), 1–29.

22. Leonid Rozenblit and Frank Keil. 2002. The misunderstood limits of folk science: An illusion of explanatory depth. *Cognitive Science* 26, 5 (2002), 521–562.

23. Sarah Weir, Juho Kim, Krzysztof Z. Gajos, and Robert C. Miller. 2015. Learnersourcing Subgoal Labels for How-to Videos. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW '15)*. 405–416.

24. Joseph J. Williams, Na Li, Juho Kim, Jacob Whitehill, Samuel Maldonado, Mykola Pechenizkiy, Larry Chu, and Neil Heffernan. 2014. The MOOClet Framework: Improving Online Education through Experimentation and Personalization of Modules. *Social Science Research Network Working Paper Series* (Nov. 2014). http://ssrn.com/abstract=2523265

25. Joseph J Williams and Tania Lombrozo. 2010. The role of explanation in discovery and generalization: evidence from category learning. *Cognitive Science* 34, 5 (2010), 776–806.