UNIVERSITY OF PITTSBURGH

INFSCI 2970 INDEPENDENT STUDY COURSE - SPRING 2019

Explaining Need-based Educational RecommendationsUsing Interactive Open Learner Models

---

Author:
Jordan BARRIA-PINEDA
Kamil AKHUSEYINOGLU

Supervisor:
Dr. Peter BRUSILOVKSY

Contents

Chapter 1

Abstract

Students might pursue different goals throughout their learning process. For example, they might be seeking new material to expand their current level of knowledge, repeating content of prior classes to prepare for an exam, or working on addressing their most recent misconceptions. Multiple potential goals require an adaptive e-learning system to recommend learning content appropriate for students' intent and to explain this recommendation in the context of this goal. In our prior work, we explored explainable recommendations for the most typical "knowledge expansion goal". In this paper, we focus on students' immediate needs to remedy misunderstandings when they solve programming problems. We generate learning content recommendations to target the concepts with which students have struggled more recently. At the same time, we produce explanations for this recommendation goal in order to support students' understanding of why certain learning activities are recommended. The paper provides an overview of the design of this explainable educational recommender system and describes its ongoing evaluation.

Chapter 2

Introduction

Over the last few years, the field of recommender systems paid increasingattention to explaining recommendations as well as making the recommendation process more transparent to the end users. It has been argued that among other benefits, providing these explanations can boost the system's transparency and increase users' trust and satisfaction in the recommended items Tintarev and Masthoff, 2011. A number of approaches to generating explanations were explored and reported for several domains Musto et al., 2016; Sato et al., 2018; Kouki et al., 2019, yet explanations for educational recommendations received little attention so far. Meanwhile, educational recommendations have become increasingly important Manouselis et al., 2013 and the need for explanations in this domain is relatively high since learners with insufficient domain knowledge often are not able to assess the quality and relevance of recommended content Hosseini and Brusilovsky, 2017.

To the best of our knowledge, only a few attempts have been made to explore explanations in educational recommender systems. The *Knowledge Maximizer* recommender system Hosseini, Brusilovsky, and Guerra, 2013 attempted to justify the value of recommending a specific learning content by visually representing the accumulation of knowledge associated with each recommended item. Putnam et. al. Putnam and Conati, 2019 explored the attitude of students towards having explanations for hints generated by an Intelligent Tutoring System (ITS). They found that students exhibited interest in hint explanations, although this interest varied over time, and it depended on students' goals while interacting with the ITS. In our previous work Barria-Pineda and Brusilovsky, 2019, we used a combination of visual and textual explanations, and found that learners with access to textual explanations were more

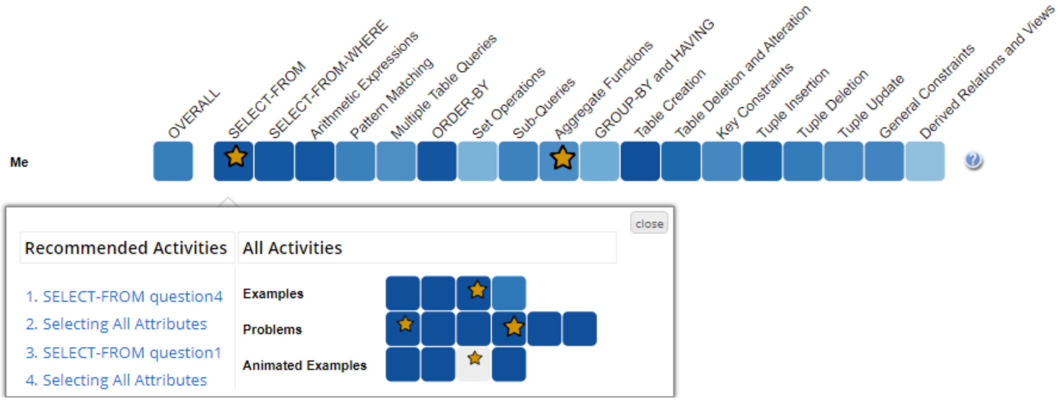

FIGURE 2.1: Activities provided by Mastery Grids for a specific topic (recommended ones are depicted with stars).

eager to attempt the recommended activities than students with no access.

Educational recommender systems have several other interesting properties which make the problem of explanations of this domain especially interesting and challenging. First, student-focused recommendations of educational content should seriously address the current level of learner knowledge rather than learner's interests. Second, recommendations should address current student goals as well as instructors' approach to teach a course. In turn, learner goals could differ for different courses, for different students taking the same course, and even for the same student at different points of study. Most educational recommender systems implicitly assume that the learner goal is "knowledge expansion", i.e., master more and more domain concepts until the whole set of educational objectives is reached Hosseini et al., 2015. However, in some contexts such as exam preparation, learners might want to review already learned knowledge rather than advancing to new. In other contexts, learners might have to focus on immediate needs, i.e. already attempted concepts which have caused problems and misunderstanding. In each of these contexts, a properly designed recommendation approach could help.

The presence of multiple possible learning goals makes the process of explaining educational recommendations especially challenging since good explanation should explain how a specific item is contributing to the current learning goal. In this paper, we focus on a less-explored case of "need-based" or "remedial" educational recommendations, i.e., recommendations that support users in addressing their problems and misconceptions. We present a design of an explainable content recommender system, which uses an open learner model (OLM) as an approach to explain need-based recommendations and make them more transparent. We also review our on-going study of the need-based recommender and report results of our survey of user goals in a college-level database course supported by the system.

Chapter 3

Recommendations in Mastery Grids System

Mastery Grids is an open social learner modeling interface which allows students to access different types of smart learning content to practice Loboda et al., 2014. In Mastery Grids topic-level OLM interface, the course content is grouped into a set of topics (see columns in Figure 2.1) and the level of knowledge for each topic is visualized using color density. By clicking on a topic cell, students can see the practice content associated with the topic. Similar to the topic-based visualization, for each topic, Mastery Grids shows the progress level for each type of content. Figure 2.1 shows practice content for the topic *SELECT-FROM* and the progress level for each content available. Moreover, the list of recommended activities highlighted with stars and explicitly as a ranked list.

In this study, we have designed Mastery Grids for a database management course with three different smart learning content concentrated on Structured Query Language (SQL): examples, problems and animated examples. In problems content, students are asked to write SQL statements to solve a given problem. The problems are important for the study because the student knowledge-level is updated based on the evidence gathered from the solutions attempts to the problems. Details are explained in Section 4.1.

Presenting current progress level provides navigational support to the students. In our previous study Hosseini et al., 2015, we introduced personalized recommendation approaches to improve existing navigational support. Top three recommended content items were highlighted using *red stars* on colored cells for topics and content. This way of representing recommended items does not force students to follow the recommendations but rather help them to combine both progress information and recommendation to decide their next action step. Originally, the system does not provide any hint or explanation for a given recommendation. However, in Barria-Pineda and Brusilovsky, 2019 the interface was redesigned to connect recommended activity with a finer-grained concept-level OLM. Thus, we already explored an approach to explain the recommendations. Different from our previous work, in this paper, we focused on producing remedial recommendations to support struggling students. Moreover, we used a simpler recommendation approach and reduced the complexity of the student modeling service. The details of the recommendation approach and the student modeling explained in the next section.

Chapter 4

Remedial Recommendations

Our remedial recommendations approach focuses on domain concepts that students have *struggled* with in their recent problem-solving attempts. Personalized remedial content recommendations should be adjusted to the current knowledge level and the immediate need of the learner. Thus, the system should model both the learner knowledge (to what extent each domain concept is know) and the learner needs (which domain concepts students are struggling with). Moreover, both modeling approaches should be sufficiently transparent to make recommendations easy to explain.

In our previous work, we used a Bayesian network based student model and an expert-defined rule-based recommendation algorithm Barria-Pineda and Brusilovsky, 2019. Due to the complexity of the underlying rules and the partial use of the concept levels in the recommendation process, the visual/textual explanations for recommendations presented do not match perfectly in all cases. The lack of match between recommendations and explanations makes it hard for students to build a clear mental model to decide what to practice next. It also might affect students' trust in the system and lead them to ignore the recommendations provided Muir, 1994. To overcome this problem, we decided to use to a less complex concept-based student modeling approach implemented in the CUMULATE user modeling server Yudelson, Brusilovsky, and Zadorozhny, 2007, and designed a simpler recommendation approach. These approaches are described in the following sections.

4.1 Student Learning Modeling

CUMULATE combines evidence generated from problem-solving attempts using an asymptotic function. This function is used to calculate the probability of a learner mastering a concept. The probability of mastery increases with each successful attempt. Due to the nature of the asymptotic function, first attempts on a concept rapidly increases the probability, however, in later successes, the rate of growth diminishes. Thus, as the learner approaches to master a concept (approaching to 1), the change in probability asymptotically decrease. It is also worth to note that, CUMULATE does not take into consideration wrong attempts. Therefore, there is no decrease in knowledge level even if a student fails (i.e. no penalty). We hypothesize that these features make CUMULATE student modeling less-complex to visualize, explain and fit more to novice users' belief of how knowledge grows compared to the Bayesian student model we used in our previous study Barria-Pineda and Brusilovsky, 2019. As a result, the less-complex student model helps us to generate more coherent visual and textual explanations.

CUMULATE needs a map between each activity (i.e. examples, problems, etc.) and domain concepts that are practiced when working with it. In this way, the evidence of a successful problem-solving attempt is uniformly applied to all concepts related to a problem, i.e., the knowledge level increases for all concepts in the same way. In this paper, we focus on SQL programming domain (rather than Java programming explored in previous work Barria-Pineda and Brusilovsky, 2019) and use SQL ontology (`http://www.pitt.edu/~paws/ont/SQL.owl`).

4.2 Recommendation Approach

In this study, we followed a simpler personalized recommendation approach compared to Barria-Pineda and Brusilovsky, 2019 and focused on remedial recommendations as it could be one of the different specific goals that students can set during their learning process. The recommendation approach consists of the following steps: (1) Calculate/update the probability of mastering each concept using CUMULATE. (2) Calculate a difficulty score of each activity. (3) Identify struggling concepts. (4) Eliminate activities without any struggling concept. (5) Rank activities based on the difficulty score. (6) Recommend activities distributed around median difficulty score.

Except from *detecting struggling concepts* and *calculating the difficulty score*, other steps are straight forward and self-explanatory. Here in this section, we will explain these steps in detail. However, it is important to note that in the last step of the recommendation approach, we used median difficulty score to specify suitable activities to recommend as remediation, i.e. learning content that is not too hard activities but at the same time not too easy. We hypothesized that activities which reside at median level difficulty for a student would not be so hard or so easy to lead any further hardship or discouragement.

4.2.1 Calculating Difficulty Score

This step can be broken into further smaller steps as follows:

1. *Calculate the importance of a concept within a topic*: As described earlier, each topic in Mastery Grids contains a set of activities and each activity is associated with a set of concepts. Thus, each topic can be represented as a collection of concepts. The weight of a concept within a topic is calculated using tf-idf approach, i.e., the more uniquely a concept is covered by one specific topic, the higher its importance will be for that topic (in contrast to concepts that are covered in several other additional topics).

2. *Calculate the concept difficulty*: Concept difficulty is calculated based on a student's current knowledge level and the average success rate on the concept. The knowledge level is determined by CUMULATE student model. We have calculated the concept-based success rate by treating each problem-solving attempt as an opportunity for the concepts associated with it. Thus, if a student succeeds in an attempt, s/he succeed on all concepts related to it. Aggregating correct/incorrect attempts on the concepts, we can calculate the average success rate per concept in last t attempts. t is set to 10 for this study.

Using the calculated weight, knowledge level, and success rate, the difficulty score $diff_{ij}$ of an activity i for student j is calculated by equation 4.1:

$$diff_{ij} = \frac{1}{\sum_k w_k} \sum_k w_k \left(\alpha * Q_{kj} + (1 - \alpha) * s_{kj} \right) \tag{4.1}$$

where k is a concept associated with activity j; Q_{kj} is knowledge level and and s_{kj} is the success rate of student j on concept k. For this study, α is set to 0.5 to put equal importance on knowledge level and the success rate. After conducting a real classroom study, we are planning to tune these manually set parameters.

4.2.2 Identifying Struggling Concepts

Remedial recommendations should focus on concepts with which a student struggled recently and have not learned properly. Using the concept-based success rate (s_{kj}) and knowledge level (Q_{kj}) calculated in the previous step, we defined a concept as struggling if $s_{kj} < 0.5$. Please note that s_{kj} is calculated by using the last t attempts which reflects the recent performance of a student. Therefore, the system does not put emphasis on historical success rate as opposed to knowledge growth and will not label a concept as struggling if the student starts to perform well (assuming the success rate goes above 0.5).

Chapter 5

Explaining Recommendations

5.1 Visualizing Knowledge Level

The concept-level knowledge estimation is visualized as a bar chart (see Figure 5.1), where the bar length represents the actual knowledge level. Initially, all bar-lengths are set to 0 and start to increase based on the evidence collected by CUMULATE. After a successful problem-solving attempt, the knowledge estimates are updated as described in 4.1 and the corresponding concept bars rise. As mentioned earlier, wrong problem-solving attempts do not change knowledge estimates thus the concepts' bar chart keeps the same.

5.2 Visualizing Struggling Concepts

As each individual concept held by the student model is represented as a bar, we used a second visual encoding variable for representing the level of struggle of a specific concept. This variable is color, and we defined a color scale going from red to green. The bar color gets greener with higher success rates, and they are gray if the concept has not been practiced in the last t attempts. As explained in section 4.2.2, concepts are identified as struggling if the success rate is below 0.5. To make it apparent to students, we visually labeled struggling concepts with a *warning sign* shown on top of the concept bars. This way students can easily recognize if they are struggling with any particular concept by checking the concept bar chart as shown in Figure 5.1. The interface features are explained in a start-up tutorial, and can get a reminder through help buttons.

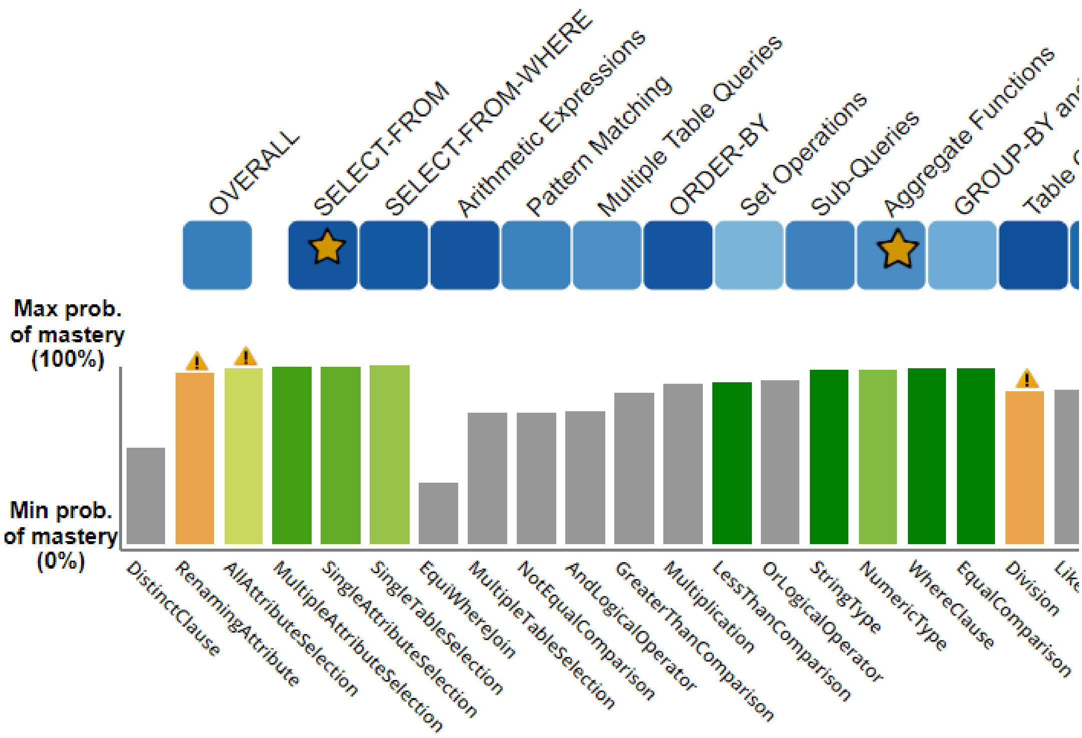

FIGURE 5.1: Rich OLM showing topic-level progress (grid) and concept-level knowledge estimation (bar chart)

5.3 Explanation Generation

Explanations are shown to the students when a recommended activity (i.e. an activity cell with a star) is *mouseovered*. We categorized explanations into two groups, visual and textual:

5.3.1 Visual Explanations:

To make learners understand which concepts are required to understand an example/animated example or to solve a problem, we highlighted concept-bars as shown in Figure 5.2. Highlighted concepts help learners to catch a glimpse of their knowledge levels and whether or not they are struggling with any of the related concepts (i.e. warning sign). This would help them to understand better why an activity is recommended.

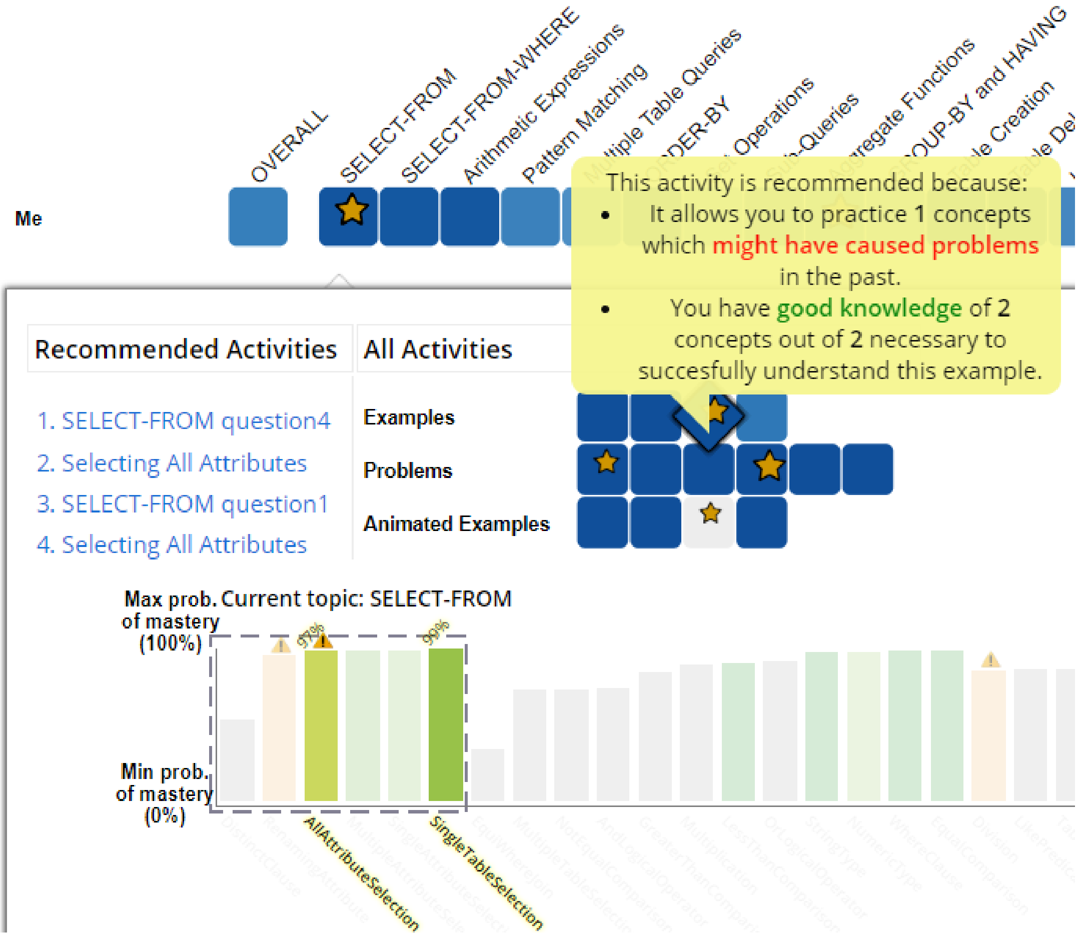

FIGURE 5.2: Visual and textual explanations triggered when mouseovering a recommended activity.

5.3.2 Textual Explanations:

A textual explanation is showed as a *tooltip* which summarizes: (1) Knowledge-level (2) Struggling concepts. As shown in Figure 5.2, the explanation states that the student should practice the recommended activity since s/he has one struggling concept (i.e. a concept which could be causing problems) and s/he has *good knowledge* of both concepts to understand the example. We used colored text to put emphasis on struggling concepts (red) and knowledge level (green). The text related to knowledge level is presented if the student demonstrated enough proficiency in any related concept. For now, the proficiency threshold is set to 0.66 as it is both conservative and understandable by the student (upper third of the knowledge range).

Chapter 6

In-progress Evaluation

Currently, we are running an experiment to evaluate the effect of the recommendation approach and the generated explanations (visual and textual) on students' behavior in a college-level database management course. We conducted a survey at the beginning of the term to collect the enrollment goals of 377 students. The survey was co-designed by the course instructors, who defined the most common students goals for taking the course given their broad teaching experience and institutional context. The survey results revealed that students have different goals in selecting this course offering. 251(67%) students reported that the course is either mandatory or highly recommended in their program. On the other hand, 126(33%) students reported that they are taking the course voluntarily or as optional. In order to have more insight about the students taking the course as non-mandatory, we asked them a follow-up question. 59(16%) students said that they are interested in databases in general. On the other hand, 48(13%) students thought that completing this course would increase the probability of getting a job. Finally, 216(57%) students stated that they could be a database manager in the future, while 161(43%) students stated the opposite.

In short, students have different plans/goals to take the database management course. To be able to fulfill different expectations, the system should adjust the recommendations given to the students based on those different goals, e.g., if they are planning to become database administrators they need to master specific core concepts in contrast to someone that only wants to learn at a surface level.

Chapter 7

Conclusion

In this paper, we acknowledge that students come with diverse goals when starting a new learning process and that we should take them into account when generating knowledge-based learning content recommendations. We presented the design of an explainable educational recommender system, which uses an interactive OLM as an approach to explain these need-based recommendations and make them more transparent. For this initial exploratory process, we focused on one specific goal, which is remediating knowledge about problematic concepts, and we adapted the generated explanations based on that. This effort enables us to check if explanations have a stronger influence on students when the recommendation algorithm and the explanations are aligned. We reviewed an ongoing study to understand the effect of this explainable need-based recommender in a college-level database course. Finally, we reported preliminary results of our survey about learners' goals, which confirmed the idea that the goals of students taking a course are diverse. Collecting these individual differences through different instruments is vital to enable more holistic recommendations that adapt their mechanisms and potential explanations.

Chapter 8

Future Work

At the end of the experiment, we will analyze the students' activity logs in the system to check the effectiveness of the need-based recommender and the explanations. We also plan to conduct a post-survey to understand learners' thoughts about the recommendations and the explanations. We are planning to check usage differences based on student goals and design new classroom experiments based on the insight we will get. Additionally, we want to tune manually set parameters based on the usage data and the student goals. Finally, we will investigate how to include more scenarios to the need-based recommendation approach in order to define various adaptation strategies - and corresponding explanations - for students with different course goals or short-term goals that could change during the term.

Bibliography

Barria-Pineda, Jordan and Peter Brusilovsky (2019). "Making Educational Recommendations Transparent through a Fine-Grained Open Learner Model". In: *Joint Proceedings of the ACM IUI 2019 Workshops co-located with the 24th ACM Conference on Intelligent User Interfaces (ACM IUI 2019), Los Angeles, USA, March 20, 2019.*

Hosseini, Roya and Peter Brusilovsky (2017). "A Study of Concept-Based Similarity Approaches for Recommending Program Examples". In: *New Review of Hypermedia and Multimedia* 23.3, pp. 161–188.

Hosseini, Roya, Peter Brusilovsky, and Julio Guerra (2013). "Knowledge Maximizer: Concept-based Adaptive Problem Sequencing for Exam Preparation". In: *the 16th International Conference on Artificial Intelligence in Education (AIED 2013)*. Vol. 7926. LNAI, pp. 848–851.

Hosseini, Roya et al. (2015). "What Should I Do Next? Adaptive Sequencing in the Context of Open Social Student Modeling". In: *Design for Teaching and Learning in a Networked World*. Ed. by Gráinne Conole et al. Cham: Springer International Publishing, pp. 155–168. ISBN: 978-3-319-24258-3.

Kouki, Pigi et al. (2019). "Personalized explanations for hybrid recommender systems". In: *the 24th International Conference on Intelligent User Interfaces (IUI '19)*. ACM, pp. 379–390.

Loboda, Tomasz D et al. (2014). "Mastery grids: An open source social educational progress visualization". In: *European conference on technology enhanced learning*. Springer, pp. 235–248.

Manouselis, Nikos et al. (2013). *Recommender Systems for Learning*. Berlin: Springer. URL: http://www.springer.com/us/book/9781461443605.

Muir, Bonnie M (1994). "Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems". In: *Ergonomics* 37.11, pp. 1905–1922.

Musto, Cataldo et al. (2016). "ExpLOD: A Framework for Explaining Recommendations Based on the Linked Open Data Cloud". In: *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, pp. 151–154. ISBN: 978-1-4503-4035-9.

Putnam, Vanessa and Cristina Conati (2019). "Exploring the Need for Explainable Artificial Intelligence (XAI) in Intelligent Tutoring Systems (ITS)". In: *Joint Proceedings of the ACM IUI 2019 Workshops co-located with the 24th ACM Conference on Intelligent User Interfaces (ACM IUI 2019), Los Angeles, USA, March 20, 2019.*

Sato, Masahiro et al. (2018). "Explaining Recommendations Using Contexts". In: *23rd International Conference on Intelligent User Interfaces*. ACM, pp. 659–664.

Tintarev, Nava and Judith Masthoff (2011). "Designing and Evaluating Explanations for Recommender Systems". In: *Recommender Systems Handbook*. Ed. by Francesco Ricci et al. Boston, MA: Springer US, pp. 479–510. ISBN: 978-0-387-85820-3.

Yudelson, Michael, Peter Brusilovsky, and Vladimir Zadorozhny (2007). "A User Modeling Server for Contemporary Adaptive Hypermedia: An Evaluation of the Push Approach to Evidence Propagation". In: *User Modeling 2007*. Ed. by Cristina Conati, Kathleen McCoy, and Georgios Paliouras. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 27–36. ISBN: 978-3-540-73078-1.