Personalized Remedial Recommendations for SQL Programming Practice System

Jordan Barria-Pineda*
Kamil Akhuseyinoglu*
jab464@pitt.edu
kaa108@pitt.edu
University of Pittsburgh
Pittsburgh, USA

Peter Brusilovsky
University of Pittsburgh
Pittsburgh, USA
peterb@pitt.edu

Kerttu Pollari-Malmi
Aalto University
Espoo, Finland
kerttu@cs.hut.fi

Teemu Sirkiä
Aalto University
Espoo, Finland
teemu.sirkia@aalto.fi

Lauri Malmi
Aalto University
Espoo, Finland
lauri.malmi@aalto.fi

ABSTRACT

Personalized recommendation of learning content is one of the most frequently cited benefits of personalized online learning. It is expected that with personalized content recommendation students will be able to build their own unique and optimal learning paths and to achieve course goals in the most optimal way. However, in many practical cases students search for learning content not to expand their knowledge, but to address problems encountered in the learning process, such as failures to solve a problem. In these cases, students could be better assisted by *remedial recommendations* focused on content that could help in resolving current problems. This paper presents a transparent and explainable interface for remedial recommendations in an online programming practice system. The interface was implemented to support SQL programming practice and evaluated in the context of a large database course. The paper summarizes the insights obtained from the study and discusses future work on remedial recommendations.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Social and professional topics → Computer science education**; • **Applied computing → Interactive learning environments**.

KEYWORDS

educational recommender systems, explainability, transparency

ACM Reference Format:
Jordan Barria-Pineda, Kamil Akhuseyinoglu, Peter Brusilovsky, Kerttu Pollari-Malmi, Teemu Sirkiä, and Lauri Malmi. 2020. Personalized Remedial Recommendations for SQL Programming Practice System. In *Adjunct Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '20 Adjunct), July 14–17, 2020, Genoa, Italy*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3386392.3399312

*Both authors contributed equally to this research.

UMAP '20 Adjunct, July 14–17, 2020, Genoa, Italy

ACM ISBN 978-1-4503-7950-2/20/07…$15.00
https://doi.org/10.1145/3386392.3399312

1 INTRODUCTION

Over the last decade, the issues of transparency and control in recommender systems (*RecSys*) have emerged as an important stream of research. One technology that has been studied in this context is the explanation of recommendations. Research has shown that explanations can increase persuasiveness of the recommended items as well as users' trust and satisfaction with the recommender system [17]. Based on these results, guidelines have been developed for designing and evaluating the benefits of explanations [16]. Despite of the increasing volume of research on explaining recommendations, this work has been predominantly focused on traditional taste-based and interest-based recommendation in e-commerce and media consumption systems and such items ad products, movies or songs [11]. In this paper, we explore the problem of explaining recommendations in a considerably different domain, e-learning, where recommendations usually focus on user's knowledge rather than interests. Here, we explore a relatively new class of remedial recommendations focused on helping to address problems encountered during the learning process. Following a brief review of related work, we introduce a novel interface for explaining remedial recommendations. The remaining part of the paper reviews the results of our study of this technology in a target educational context. We conclude by discussing lessons learning and planning future work.

2 RELATED WORK

A field that has been understudied in the *RecSys* context is the educational domain, where the main goal of a recommender system is to support students' learning by filtering educational content for the different learning settings that differ from one individual student to another [18]. Although there is a large body of research on Educational Recommender Systems (*EdRecSys*) [6], to the best of our knowledge, there is no research work that has tried to investigate the effects of explanations for students in *EdRecSys* contexts. Thus, it is not clear how feasible it is to directly transfer the lessons learned in other recommendation domains into this context. The closest

attempt we have identified is the work of Putnam and Conati [13], which studied students' perceptions and attitudes toward explanations for automatically generated hints in an Intelligent Tutoring System scenario.

The aforementioned gap is important to address because *EdRecSys* are different from conventional *RecSys* as their main goal is supporting students' learning [6]. Thus, not only the interests/preferences of the end-users (students) are important to generate the recommendations, the level of domain knowledge at each stage of their learning is crucial for suggesting appropriate learning activities to each student [18]. Hence, this difference makes it necessary to (1) include students and instructors in the design of the recommendation approach since its conception [14], and (2) define proper evaluation metrics to assess the recommendations effectiveness which must include students' learning [7].

The implementation of *EdRecSys* stresses how critical it is to consider that each student is unique in terms of her "knowledge readiness" for attempting learning activities. Likewise, we hypothesize that it is also important to ponder the students' ability to process recommendations and understand potential explanations. In other scenarios, researchers have discovered that the explanations' level of detail can affect users' mental models with both positive and negative effects [11]. Additionally, they have found that explanations' complexity (e.g., depth and visual format) could either help or burden users' understanding [15].

Altogether, there is an evident lack of research on the effects of transparency in the context of educational recommendations. Exploring the potential benefits/drawbacks of recommendations and their explanations will contribute to improving the future of adaptive and personalized online learning. In this paper, we aim to fill this research gap through an empirical classroom study.

3 SQL PROGRAMMING PRACTICE SYSTEM WITH REMEDIAL RECOMMENDATIONS

3.1 Interface

The core of the SQL Programming Practice System is the *Mastery Grids* interface which offers open learner modeling (OLM) [5] and provides access to several types of learning content [12]. The version of Mastery Grids interface for SQL is presented in Figure 1-A. Mastery Grids uses a topic-level approach to OLM where the course content is grouped into a set of topics (see Fig. 1-A). The level of progress for each topic is visualized using a color opacity. The higher the opacity, the higher the progress of the learner in that topic. In addition to the topic-level progress visualization, Mastery Grids shows the progress level for each type of content for each topic. In Fig. 1-A, the available practice content for the topic *SELECT-FROM* is shown, as well as the associated progress level for each content. In addition, the interface recommends most appropriate learning activities to each learner in two ways: (1) highlighting recommended activities with stars on the grid of activities and (2) offering them as a ranked list on the left from the grid. Students can access the learning content by clicking on an activity cell or a line of the ranked list.

3.2 Learning Content

In this study, Mastery Grids provided access to three types of interactive practice content for learning SQL programming: annotated examples (labeled as *Examples* in Fig. 1-A), animated examples, and query problems. Annotated examples provide step by step text explanations to SQL query statements, which are delivered by the *WebEx* system [4]. Query animations visualize the execution of a query. The aim of these examples is visually demonstrating how various query clauses are executed (step-by-step) to help students understand the semantics of the query. Finally, query problems require students to write an SQL query to solve the given problem prompt using the associated database schema. The correctness of the query is evaluated against a model solution using the sample database and immediate feedback is provided. These problems are served by the *SQL-KnoT* (Knowledge Tester) system [3]. *SQL-KnoT* leverages *template*-based problem-generation. Every time a student accesses a problem, the actual problem is randomly selected from a problem set associated with the template. *SQL-KnoT* problems are critical for the study because the knowledge-level of a student is updated based on her attempts on these problems.

3.3 Student Modeling and Knowledge Level Visualization

Mastery Grids can depict the concept-level knowledge estimation as a bar chart (see Fig. 1-C&D). Each bar represents the actual student's knowledge level estimated by the system for a specific concept. Initially, all bar lengths are set to 0 and start to increase based on the successful problem-solving attempts. The knowledge estimates are calculated by the CUMULATE user modeling server [19]. CUMULATE combines evidence generated from problem-solving attempts using an asymptotic function. This asymptotic function is used to calculate the probability of a learner mastering a concept. The probability of mastery increases with each successful attempt. CUMULATE does not take into consideration wrong attempts. Therefore, there is no decrease in knowledge level (i.e., there's no penalty in the student model) – even if a student fails. The asymptotic nature of the CUMULATE student modeling function implies that when a student starts studying a new concept, learning gains are high; however, these learning gains rapidly become smaller as the student becomes more proficient on the subject.

The concepts are grouped and arranged along the x-axis by topic, according to the order in which topics are covered in the course (see Fig. 1-C&D). When students mouse over a grid cell that represents a topic, the interface highlights the concepts in the bar chart that the topic covers. Learners can check their estimated knowledge of the related concepts in a specific topic to the presence or absence of bars in the chart. The concepts that are associated with each content are also highlighted similarly. The height of each bar indicates the estimated level of knowledge of the student. We also used a second visual encoding variable to represent 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 a higher success rate, and it gets gray if the concept has not been practiced recently. If a concept is labeled as *struggling* (see section 3.4.1), the system depicts it with a warning sign shown on top of the concept bar, as shown in Fig. 1-C&D.

Presenting the current progress level provides navigational support to the students. In our previous study [8], we introduced personalized recommendation approaches to improve existing navigational support. The 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, Mastery Grids does not provide any hint or explanation for a given recommendation. In our previous work [1], the interface was redesigned to connect recommendations with a finer-grained concept-level OLM and explored an approach to explain the learning content recommendations. Following that study, we shared our new system design and the study plan in [2] where we focused on producing remedial recommendations to support struggling students. Moreover, we introduced a simpler recommendation approach and reduced the complexity of the student modeling service. The details of the recommendation approach and student modeling are explained in the next section. Following [2], in the current paper, we share the results of that study.

3.4 Educational Recommender System

3.4.1 Recommendation Approach. In this study, we used a remedial recommendation approach, which focuses on suggesting learning activities that cover some of the concepts where students have exhibited some level of struggle [1, 2]. In other words, remedial recommendations should target the concepts with which a student struggled recently.

To generate remedial recommendations, we used the concept-level knowledge estimated by CUMULATE student model and concept-level success rate to calculate a *difficulty score* for each learning activity. The difficulty score diff_{ij} of an activity i for student j is calculated by equation 1:

$$\text{diff}_{ij} = \frac{1}{\sum_k w_k} \sum_k w_k \Big(\alpha * Q_{kj} + (1 - \alpha) * s_{kj} \Big) \qquad (1)$$

where k is a concept associated with activity i, Q_{kj} is knowledge level estimate, s_{kj} is the success rate of student j on concept k, and w_k is the topic-level importance of the concept calculated by using tf-idf approach (i.e., the more unique a concept in a topic, the higher its importance). We considered each problem-solving attempt as an opportunity for the concepts associated with it and calculated the average success rate per concept in last t attempts. For this study, t is set to 10 and α is set to 0.5 to put equal importance on knowledge level and the success rate.

To focus on struggled concepts, we eliminate learning activities that do not have any struggled concepts from the recommendation process. We defined a concept as *struggling* if students started to fail on problems that cover the concept. Specifically, using the concept-based success rate (s_{kj}), we defined a concept as *struggling* if $s_{kj} < 0.5$. As s_{kj} is calculated by using the last t attempts, the system will not label a concept as *struggling* if the student starts to perform well (assuming the success rate goes above 0.5). We further calculated the median difficulty score after each attempt to specify suitable activities to recommend as remediation, i.e. learning content that is not too hard 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. The details of the recommendation process can be found in [2].

3.4.2 Explanations for Learning Content Recommendations. Given our past work on communicating learners the reasons behind suggesting them learning activities [1], we decided to define different experimental treatments by combining textual and visual explanatory elements. Our goal here is to examine the different effects that different explanation formats can have on students working with recommended content in an online learning environment. Thus, we defined 4 treatment groups that are explained here:

(1) *NoExp* group: In this group, no explanation is provided to the students when mousing over a recommended activity (see A in Fig. 1). Thus, learners do not know why a specific learning material was suggested to them.
(2) *TextualExp* group: Only an explanation based on natural language is provided to these students when mousing over recommended activities (see B in Fig. 1). This explanation format textually describes (a) how many *struggling* concepts the learning content is covering and, at the same time, (b) how many concepts in that specific activity the student has shown a good proficiency-level (which makes it more approachable to solve the ongoing misconceptions).
(3) *VisualExp* group: Here, a concept-based OLM is used as a visual explanatory component when mousing over recommended activities (see C in Fig. 1). By examining their own OLMs, students are able to know how many *struggling* concepts they have (as they are highlighted with warning signs) and their respective level of proficiency on each of the concepts covered by the activity. Each concept bar visualizes the learner's knowledge-state by means of two graphic variables:
    - length: shows the cumulative estimation of knowledge of the student, which is calculated based in the historical performance on problems involving that concept.
    - color: shows the success rate on the most recent attempts on problems that include that concept, by using a scale that ranges from red (0%) to green (100%) and using yellow as an intermediate point (50 %).

    It is important to mention that for this group, the visual OLM highlights the information about the associated concepts for the moused over activities, but not only for the recommended ones (i.e., when mousing over non-recommended activities they can see their knowledge state on associated concepts).
(4) *DualExp* group: In this version of the interface both textual and visual explanatory components explained above are shown to the students when mousing over recommended activities (see D in Fig. 1). We hypothesize that by having these explanatory components together, learners could get a clearer picture about why the recommendation algorithm selected those specific activities to remediate their misconceptions (balance between *struggling* concepts and concepts where they are proficient).

4 STUDY DESCRIPTION

We conducted a classroom study in Spring 2019 term at Aalto University, major research university in Finland, from February to May.

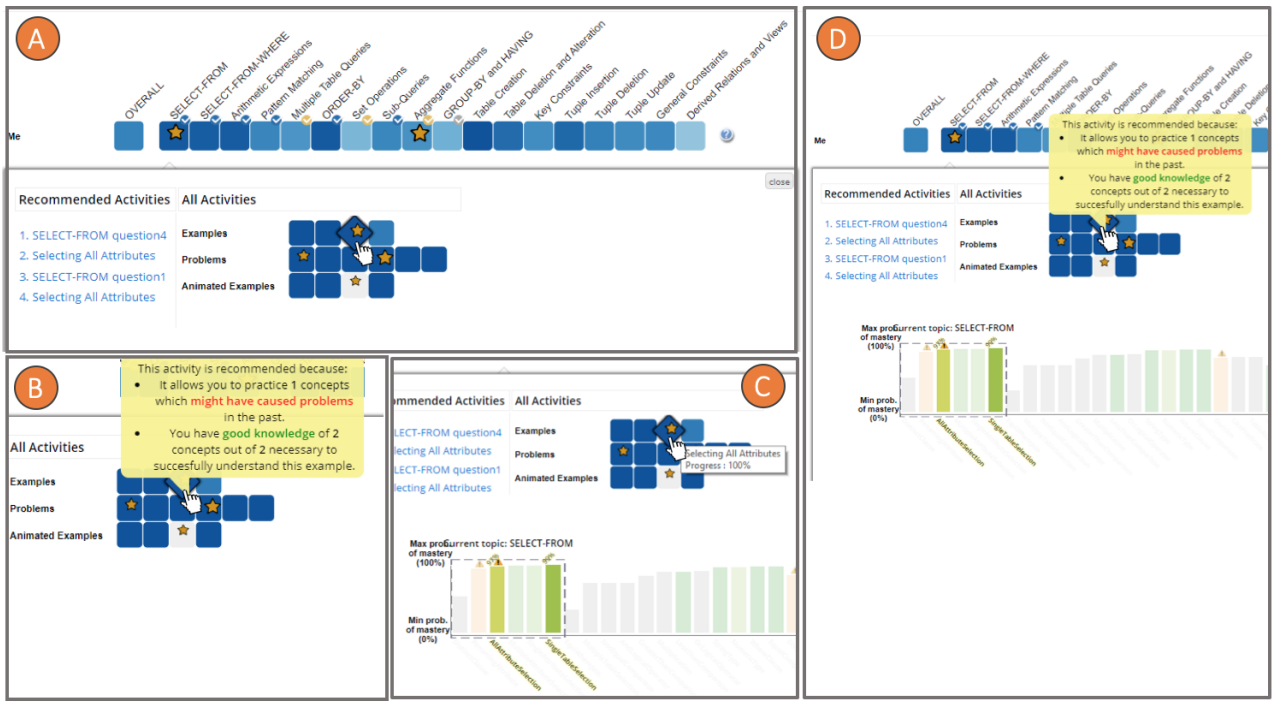

Figure 1: Four different experimental treatments which combine textual and visual elements for explaining recommendations in Mastery Grids: (A) No explanation (B) Textual explanation only (C) Visual explanation only (D) Visual and textual explanation combined

Mastery Grids was offered as a practice system to students who were enrolled in an undergraduate database management course named "CS-A1150 - Databases". The course is a database management course covering topics such as relational modeling, relational algebra, UML modeling, SQL, transaction management, etc. The course is compulsory for Computer Science and Industrial Engineering and Management majors and highly recommended for Computer Science minors. The course is also taken by many other students in various bachelor's and master's programs. In total over 550 students enrolled the course. In the beginning of the course, we informed students about this research and asked them to give their consent to participate in it. The data used in this research is from those students who gave their consent and were engaged with Mastery Grids in some way. Not all students used the system, because it was a voluntary additional learning resource.

Mastery Grids content is designed to help students to practice on SQL topics and it was accessed through a direct link from A+ course management system [9], which was used to deliver other contents and exercises of the course. The use of Mastery Grids was not-mandatory but to encourage participation 20 extra exercise points, which were about 7.5 % of all exercise points available, were given to students if they solved 2 SQL problems per topic.

In this study, we followed a pre/post test design to examine the learning gain throughout the semester. Before the SQL topics were introduced, the pretest was administered. At the end of the semester, the post-test was administered. Both tests include 10 problems, 5 multiple-choice and 5 SQL *fill in the blank* problems covering data definition, data query and data manipulation SQL commands related to a given database schema. Post-test problems were isomorphic to the pretest. However, in our analysis, we realized that students do not spend enough time on post-test problems and decided to use the final exam grades instead of post-test scores. Moreover, students were asked to complete a questionnaire related to Mastery Grids usage at the end of the semester. The questionnaire consisted of questions related to system satisfaction, interface features and recommendation quality. To encourage the students to take the pre/post-tests and complete the questionnaire, 4 exercise points and 1 exam bonus point (out of total 40 exam points) were given to the students who completed them.

5 RESULTS

At the end of the experiment, we noticed that most of the students' activity in the Mastery Grids was registered during the last weeks of the course. At that point of the term, the provided educational tools served more like a knowledge-confirmation instrument rather than one for accurately measuring their knowledge acquisition process. Given this, for further analysis we only consider the subset of students who clearly exhibited needs for remediation at some point of their work with the system, reflected by: (a) not having a high success rate on their submissions, and (b) accessing to at least a couple of recommended remedial problems. Thus, in the subsequent analysis, we only considered students with an average success rate lower than 75% and with at least more than 2 attempts on recommended problems. After filtering out students who did

not exhibit troubles when solving problems, we ended up having 18 students in the *TextualExp*, 25 in the *VisualExp*, 20 in the *NoExp* and 20 in the *DualExp* group. Table 1 shows the summary statistics for important usage variables after the filtering process mentioned above.

	Mean	Sd	Median
Learning indicators			
Pretest	11.7	10.7	8
Posttest	38.8	12.5	43.3
Norm. Learn. Gain	.776	.211	.824
Learning actions			
Problems attempted	41.35	19.1	37
Rec. problems attempted	6.82	8.1	5
Correct problems	33.7	13.7	31

Table 1: Student behavior summary

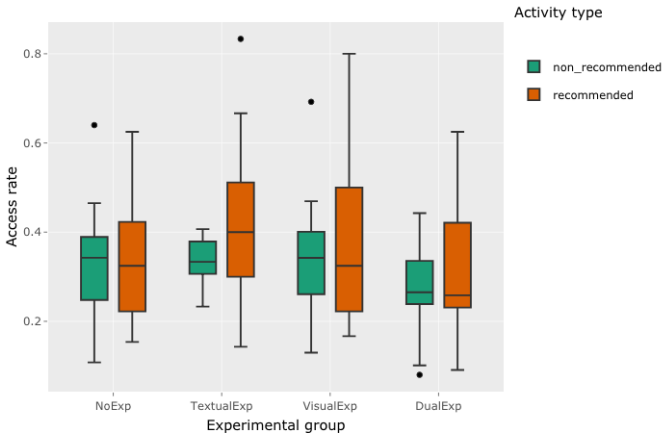

Figure 2: Average access rate per treatment group

5.1 Persuasiveness of Recommendations

We defined three levels of engagement with the activities: (1) the probability to access them when they are moused over (*access rate*), (2) the probability of attempting them when these are opened (*conversion rate*), and (3) the probability of keep working on the activities until solving it correctly (*persistence rate*). These three levels go from a low level of persuasion, where the reflected engagement for following the recommendation is low (i.e., it just takes a click), to a deeper level of persuasion, where the commitment with working in the recommended activity is higher (i.e., keep trying until the problem is solved).

A mixed-design ANOVA analysis on these different levels of persuasion was performed by having: (a) one within-subjects factor which is the type of activity (i.e., recommended or non-recommended) and (b) two between-subjects factors corresponding to the two explanation formats (i.e., visual and textual).

The goal of this analysis is to detect differences between the experimental groups on their interaction behavior with recommendations, by comparing this with their work with non-recommended activities:

(1) *Access rate*: We found that the overall access rate was higher for recommended than non-recommended activities, across all the four explanatory interface treatments F(1,73)=4.624, p=.035 (see Fig. 2). This reflects that regardless of showing or not showing explanations students tended to click more on the suggested activities highlighted by stars (*Mean access rate*=.36) when comparing this with the non-recommended ones (*Mean access rate*=.32). In this context, we also found that there was a significant interaction effect between the two explanation formats on the overall access rate, F(1,73)=4.132, p=.046 (see Fig. 4). This result suggests that students tended to inspect problems in Mastery Grids more frequently when they had access to partial explanations (either visual or textual), while they were more careful in clicking the activities that they moused over (regardless of their recommendation status) when they received either none or both type of explanations (see Fig. 4).

(2) *Conversion rate*: A marginally significant interaction effect between the access/non-access to a visual explanation and the activity type (recommended/non-recommended) was found, F(1,74)=3.738, p=.057 (see left side of Fig. 5). This result suggests that in average, when students had access to visual explanations for the recommended content, their willingness to work on already-opened recommended activities (*Mean conversion rate*=.811) was lower than on non-recommended activities (*Mean conversion rate*=.841). The same marginally significant interaction effect was found between textual explanations and the activity type (recommended/non-recommended), F(1,74)=3.377, p=0.07. Again, this indicates that in general, when learners had the option to receive a textual explanation for recommended problems learners are controversially more likely to start work on opened non-recommended problems (*Mean conversion rate*=.833) rather than on opened recommended ones (*Mean conversion rate*=.801). These results, combined, make us hypothesize that the inclusion of individual explanatory elements can lead students to think a little bit more about the appropriateness of the recommendations, especially when they have the meta-cognitive ability to do this (more likely to happen at the end of the course where students have a higher degree of self-awareness about their own knowledge). For a complete view of the conversion rates distribution across treatments, please refer to Fig. 3.

(3) *Persistence rate*: A marginally significant difference on the persistence rate between recommended and non-recommended activities was found, regardless of the recommendations' explainability treatment applied on the Mastery Grids interface, F(1,74)=3.32, p=.072 (see Fig. 6). Here, average persistence in non - recommended activities (*Mean persistence rate*=.873) was slightly higher than persistence in recommended ones (*Mean persistence rate*=.83). Additionally, a marginal interaction effect was found between the visual and textual explanation treatments, F(1,74)=3.868, p=.053 (see Fig. 7). This shows that students overall persistence on problems, once they start working on them, is lower when they only had access to partial explanations (opposite phenomenon discovered for access rate).

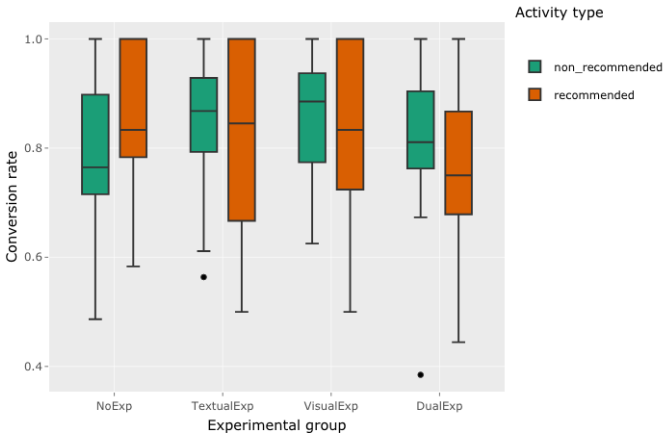

Figure 3: Average conversion rate per treatment group

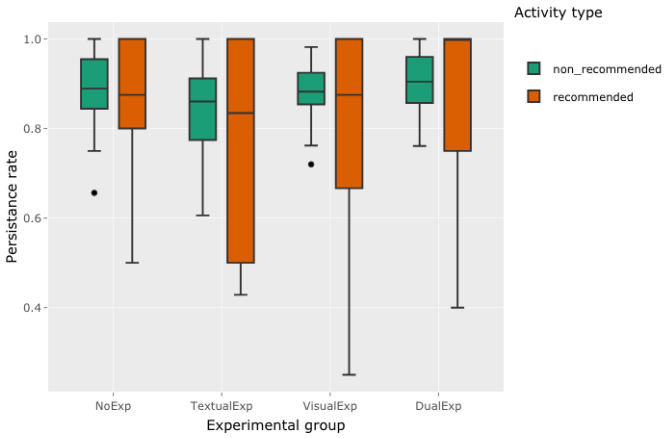

Figure 6: Average persistence rate per treatment group

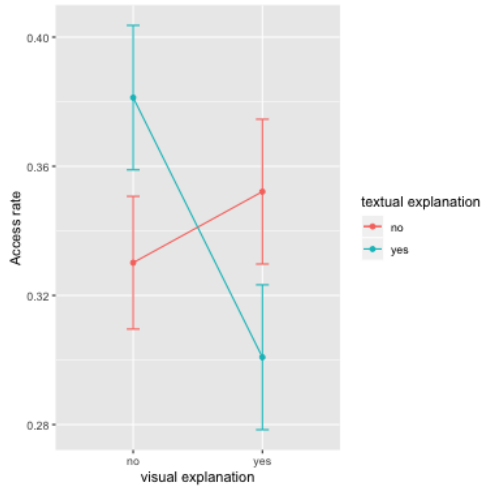

Figure 4: Overall access rate to problems in Mastery Grids per explanatory treatment

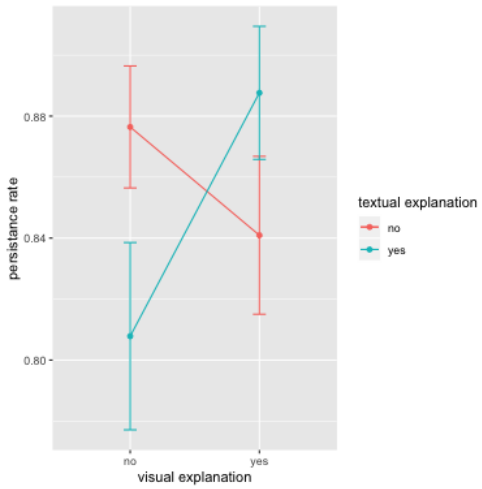

Figure 7: Overall persistence rate in problems in Mastery Grids per explanatory treatment

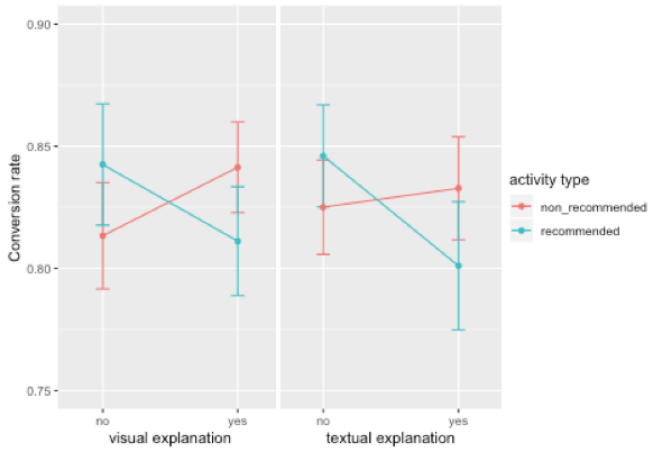

Figure 5: Effects of individual explanatory treatments on SQL problems' conversion rates

5.2 Effects of Recommendations on Learning

In order to try to link the effect of the work on remedial recommendations to the final exam scores of students in the course, we calculated a multiple linear regression model including variables like pretest and proportion of attempted problems that were recommended ones. In the resulting model, only success rate on SQL problems (regardless of their recommendation status) was the main significant predictor on the final exam grade (*Adjusted* R^2=0.185, F(6,53)=3.236, p=0.0088). No significant effect of interactions with remedial learning content was found in any of the four treatment groups.

5.3 Subjective feedback

At the end of the experiment, we asked students to fill a post-questionnaire which covered aspects like overall satisfaction with the system (3 items) and their perception about the quality of the

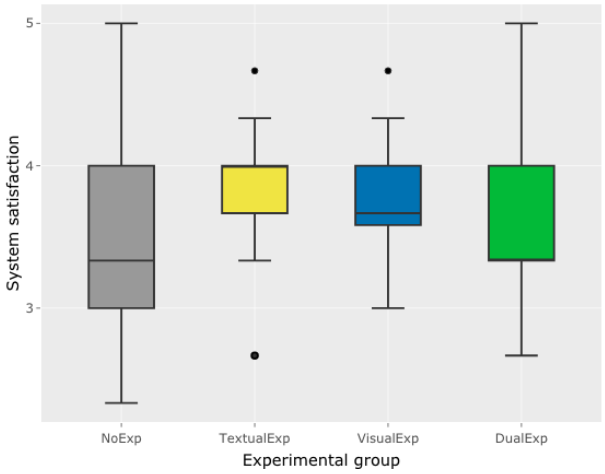

Figure 8: System satisfaction per treatment groups

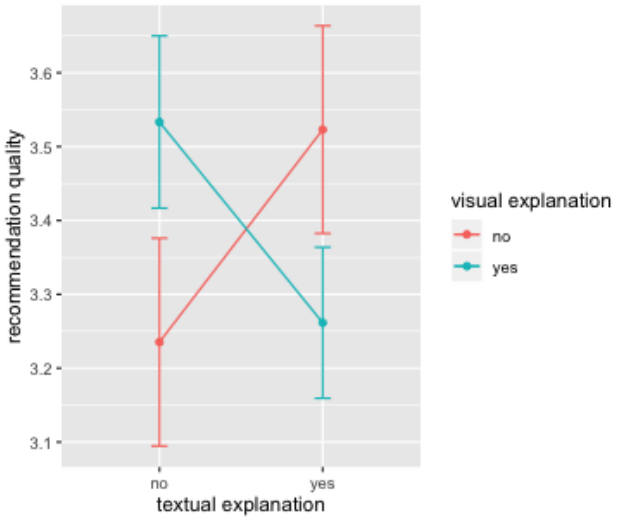

Figure 9: Recommendation quality per treatment group

remedial recommendations that were provided (5 items). The post-questionnaire included items in a *Likert* scale (1=strongly disagree, 5=strongly agree). Based on the survey responses we performed a factor analysis, which resulted in the confirmation of the existence of the aforementioned two factors (satisfaction and recommendations quality).

In terms of overall satisfaction with the system, we did not find any significant differences in the average satisfaction score given by the learners (see Fig. 8). On the other hand, we asked students about their opinion about the appropriateness of the recommendations in terms of helpfulness for their learning process (e.g., one of the items used was: *"The system recommended too many bad learning materials"*). We found a significant interaction effect of the two explanatory treatments (i.e., visual and textual explanations) on the final opinion of students about the quality of the given recommendations $F(1,57)=4.669$, $p=.035$ (see Fig. 9). We determined that in average, whenever they get partial recommendation explanations (only textual or only visual) they expressed a more positive opinion about the quality of the recommendations. Combining this finding with the conversion rate results explained in section 5.1 sheds light about learners reactions to having at least one component in the interface that provides some (incomplete) information about how the problem recommendations were generated.

6 DISCUSSION AND CONCLUSIONS

In summary, our studies confirmed that the presence of remedial recommendation affects learning content selection behavior of students – recommended activities highlighted with stars were on average more attractive. While this information correlates with past research on the impact of recommendations, it doesn't provide reliable evidence of recommendation quality or impact since users are known to trust recommendations even when they are deliberately deceiving [10]. Specifically, in the context of our study when the students accessed practice content at the end of their course, a chance for recommendations to be less than perfect was relatively high since the system was not able to track knowledge gained by the students through most of their work in class. In this situation, student reactions to recommendations *after* they accessed and examined the recommended content, and especially when they started to explore it, could provide a more reliable insight on the value of different kinds of explanation.

In this context, we found a rather unexpected impact of the provided explanations on the student conversion (i.e., a decision to start working with a problem when opened) and overall persistence (i.e., willingness to continue working with a *SQL-KnoT* problem once started). As the data show, the learners exhibited lower conversion rates for recommended problems than for non-recommended ones when any type of explanations (i.e., visual or textual) were provided. This result suggests that both explanation types enabled students to better add their own judgment on the top of recommendation and make a more balanced decision about working or not on an open recommended problem rather than rushing to anything that is recommended. In our context where student model was likely to be incomplete (leading to inadequate recommendations) it was an important advantage. A good evidence that the recommended problems in our context might not be the best match for student needs is the discovery that students' persistence on attempted problems was in average lower for recommended than for non-recommended problems (hinting that students were able to choose better problems if their decision was not affected by inadequate recommendations).

Another interesting finding is that learners were more persistent in problems (regardless if they were recommended or not) when they had access to full or no recommendations' explanations rather instead of partial ones. This behavior is opposed to the impact of explanations found on the access rate, where partial explanations positively influenced students' engagement in clicking on problems (when compared with none or full explanations). We hypothesize that this situation could be explained by the nature of partial explanations. With partial explanations, recommendations, even far from being perfect, were looking quite convincing – starting from encouraging the students to open recommended content and then to proceed with it with a considerable persistence. In contrast, full

explanations provided a better chance for the student to understand why a content was recommended and likely realize that their "current knowledge" referred by the explanations to justify recommended content doesn't represent to their true level of knowledge – which the system was not able to model. In this situation, explanations could reveal that recommendations within the system are not perfect, making students less inclined to start on a recommended problem. Yet, when the students had a chance to examine the problem in detail, they were more inclined to continue their work on the problem until solving it correctly.

This hypothesis is supported by the analysis of students' opinions about recommendation quality. According to these data, remedial recommendations were perceived to be of better quality when they were justified by partial recommendation explanation (either only textual or visual) than when received none or complete explanations. With these partial explanations, learners were able to only "partially" understand the underlying approach for generating those recommendations, which sometimes was not perfect given (a) the late stage of their learning process, and (b) the low accuracy of the student model. Results suggest that with full explanations (i.e., more information), the recommendations' quality was more likely to be critically judged.

7 LIMITATIONS AND FUTURE WORK

While our study brought some interesting and unexpected outcomes, it is important to stress that it had some practical limitations that constrained the interpretation of the presented results. As we mentioned, the use scenario appeared to be different from the expected study scenario. Instead of using the system throughout the term, the design of the credit rewards of the study encouraged students to use it very late in the course when their domain knowledge was quite high and yet not reflected by the system's learner model. This context allowed us to spot an unusual impact of explanations on student behavior. Yet, the resulting decrease of the recommendation quality might have some other unexpected impact on student work, which could affect our results.

Secondly, for data analysis purposes we decided to only sample the subset of learners that failed at a considerable rate and that, at the same time, had access to more than a couple of remedial recommendations (i.e., the target support population for the educational recommender system). However, these assumptions could be considered as too strict, which undermines the generalizability of the conclusions. For example, we did not get insights from students that were presented with remedial recommendations in Mastery Grids but never accessed or attempted the recommended content - which is one of the target aspects to explore in future studies.

Currently, a new classroom experiment is being performed, where incentives for students were clear from the beginning of the term. In this way, we will be able to measure the effects of having this remedial recommender support for students throughout all the incremental stages in the course.

ACKNOWLEDGMENTS

The work of the first author was funded by CONICYT PFCHA/ Doctorado Becas Chile/ 2018 - 72190680. This research has also been partially supported by NSF award DLR-1740775 to the third author.

REFERENCES

[1] Jordan Barria-Pineda, Kamil Akhuseyinoglu, and Peter Brusilovsky. 2019. Explaining Need-Based Educational Recommendations Using Interactive Open Learner Models. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization* (Larnaca, Cyprus) *(UMAP'19 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 273–277. https://doi.org/10.1145/3314183.3323463

[2] Jordan Barria-Pineda, Kamil Akhuseyinoglu, and Peter Brusilovsky. 2019. Explaining need-based educational recommendations using interactive open learner models. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*. 273–277.

[3] Peter Brusilovsky, Sibel Somyürek, Julio Guerra, Roya Hosseini, Vladimir Zadorozhny, and Paula J Durlach. 2015. Open social student modeling for personalized learning. *IEEE Transactions on Emerging Topics in Computing* 4, 3 (2015), 450–461.

[4] Peter Brusilovsky, Michael Yudelson, and Sergey Sosnovsky. 2004. An adaptive E-learning service for accessing Interactive examples. In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*. Association for the Advancement of Computing in Education (AACE), 2556–2561.

[5] Susan Bull and Judy Kay. 2010. Open Learner Models. In *Advances in Intelligent Tutoring Systems*, Roger Nkambou, Jacqueline Bourdeau, and Riichiro Mizoguchi (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 301–322.

[6] Hendrik Drachsler, Hans G.K. Hummel, and Rob Koper. 2009. Identifying the goal, user model and conditions of recommender systems for formal and informal learning. *Journal of Digital Information* 10, 2 (2009), 1–17.

[7] Mojisola Erdt, Alejandro Fernández, and Christoph Rensing. 2015. Evaluating Recommender Systems for Technology Enhanced Learning: A Quantitative Survey. *IEEE Transactions on Learning Technologies* 8, 4 (2015), 326–344.

[8] Roya Hosseini, I-Han Hsiao, Julio Guerra, and Peter Brusilovsky. 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*, Gráinne Conole, Tomaž Klobučar, Christoph Rensing, Johannes Konert, and Elise Lavoué (Eds.). Springer International Publishing, Cham, 155–168.

[9] Ville Karavirta, Petri Ihantola, and Teemu Koskinen. 2013. Service-oriented approach to improve interoperability of e-learning systems. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*. IEEE, 341–345.

[10] Mark Keane, Maeve O'Brien, and Barry Smyth. 2008. Are people biased in their use of search engines? *Communications of ACM* 51, 2 (2008), 49–52.

[11] Todd Kulesza, Simone Stumpf, Margaret Burnett, Sherry Yang, Irwin Kwan, and Weng Keen Wong. 2013. Too much, too little, or just right? Ways explanations impact end users' mental models. *Proceedings of IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC* (2013), 3–10.

[12] Tomasz D Loboda, Julio Guerra, Roya Hosseini, and Peter Brusilovsky. 2014. Mastery grids: An open source social educational progress visualization. In *European conference on technology enhanced learning*. Springer, 235–248.

[13] Vanessa Putnam 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 (IUI'19), Los Angeles, USA, March 20, 2019*.

[14] Olga C. Santos and Jesus G. Boticario. 2015. Practical guidelines for designing and evaluating educationally oriented recommendations. *Computers and Education* 81 (2015), 354–374.

[15] Aaron Springer and Steve Whittaker. 2019. Progressive Disclosure: Empirically Motivated Approaches to Designing Effective Transparency. In *Proceedings of the 24th International Conference on Intelligent User Interfaces (IUI '19)*. 107–120.

[16] Nava Tintarev and Judith Masthoff. 2010. Designing and Evaluating Explanations for Recommender Systems. In *Recommender Systems Handbook, Second Edition*. Vol. 2. 479–510.

[17] Nava Tintarev and Judith Masthoff. 2012. Evaluating the effectiveness of explanations for recommender systems: Methodological issues and empirical studies on the impact of personalization. *User Modeling and User-Adapted Interaction* 22, 4-5 (2012), 399–439.

[18] Katrien Verbert, Nikos Manouselis, Xavier Ochoa, Martin Wolpers, Hendrik Drachsler, Ivana Bosnic, and Erik Duval. 2012. Context-aware recommender systems for learning: A survey and future challenges. *IEEE Transactions on Learning Technologies* 5, 4 (2012), 318–335.

[19] Michael Yudelson, Peter Brusilovsky, and Vladimir Zadorozhny. 2007. A User Modeling Server for Contemporary Adaptive Hypermedia: An Evaluation of the Push Approach to Evidence Propagation. In *Proceedings of the 11th International Conference on User Modeling (UM '07)*, Cristina Conati, Kathleen McCoy, and Georgios Paliouras (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 27–36.