QuizGuide: Increasing the Educational Value of Individualized Self-Assessment Quizzes with Adaptive Navigation Support

Peter Brusilovsky, Sergey Sosnovsky, Olena Shcherbinina
School of Information Sciences
University of Pittsburgh
Pittsburgh PA 15260
peterb@pitt.edu, sas15@pitt.edu, ols1@pitt.edu

Abstract: The paper introduces QuizGuide, an adaptive system that we developed to help our students select the most relevant self-assessment quizzes. QuizGuide uses adaptive navigation support to show every student which topics are currently most important and which require further work. Despite relatively simple user modeling and adaptation techniques used in QuizGuide, the system has achieved a remarkable impact on student learning and performance. With QuizGuide the students explored more questions, worked on questions more persistently, and accessed a larger diversity of questions. This increased participation resulted in the larger increase of their knowledge at the end of the course.

1 Introduction

Individualized exercises and quizzes are considered among the most promising research topics in modern E-Learning (Brusilovsky & Miller, 2001). Starting with pioneer research on CAPA (Kashy et al., 1997), a number of projects have explored the use of individualized questions for different subjects and in different contexts. Our own work on this topic is focused on using individualized quizzes for *self-assessment of programming knowledge*. We have developed the system QuizPACK that is capable of delivering Web-based dynamic individualized quizzes for programming-related classes that use C language. An early version of QuizPACK was presented at ED-MEDIA'02 (Pathak & Brusilovsky, 2002) and the more resent version at E-Learn'03 (Sosnovsky, Shcherbinina & Brusilovsky, 2003). Over the last two years, we have run several classroom studies of QuizPACK. The results of these studies show that the use of QuizPACK for self-assessment significantly increases student knowledge of the subject. The students have also praised QuizPACK as a powerful learning tool especially stressing its ability to offer multiple version of the same question.

Despite the reported success, our analysis of student log data and free-form feedback indicated that we can potentially further increase the value of QuizPACK as an educational tool. We were concerned that students have not been using QuizPACK as much as they really need to. Through our KnowledgeTree portal (Brusilovsky, 2004), we provided 2-3 parameterized quizzes for every lecture of the course (Figure 1). Yet, most students used QuizPACK rather irregularly – just for some of the course lectures. Our hypothesis was that students require some adaptive guidance that can remind which quizzes are most important to take at any given moment of their study. Indeed, in the fall of 2003, answering the question about QuizPACK expansion, 58% of students said that the most important new feature to add to QuizPACK is an adaptive recommender for quizzes.

This paper presents QuizGuide, an adaptive system that we have developed to help our students select the most relevant quizzes. QuizGuide uses adaptive annotation, one of the adaptive navigation support technologies (Brusilovsky, 2001), to show every student which topics are currently most important and which require further work. Adaptive annotation is a popular technology in E-Learning. It has been used in several adaptive E-Learning systems (Carmona et al., 2002; De Bra, 1996; Henze & Nejdl, 2001; Papanikolaou et al., 2003; Weber & Brusilovsky, 2001). Empirical studies of adaptive annotation in educational context demonstrated that it could reduce navigation overhead, encourage non-sequential navigation, and improve learning outcome (Brusilovsky & Eklund, 1998; Brusilovsky & Pesin, 1998). QuizGuide continues to explore the educational value of adaptive annotation. The following sections present the interface of QuizGuide and the mechanism of user modeling and adaptation used in it. Also, we report some results of our recent classroom study of the system.

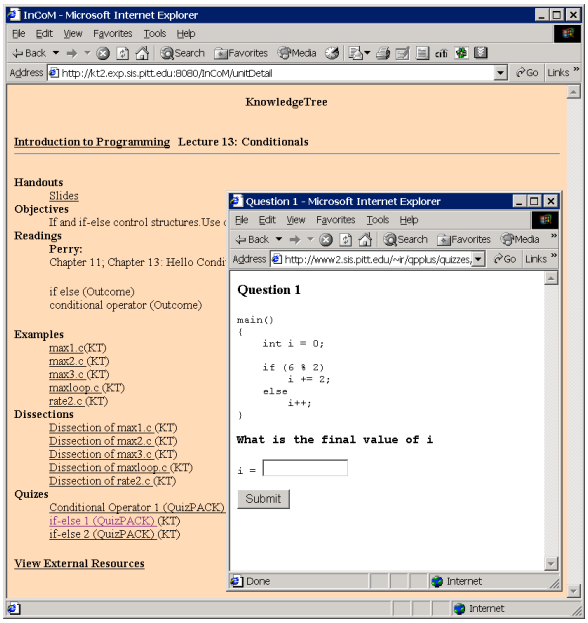
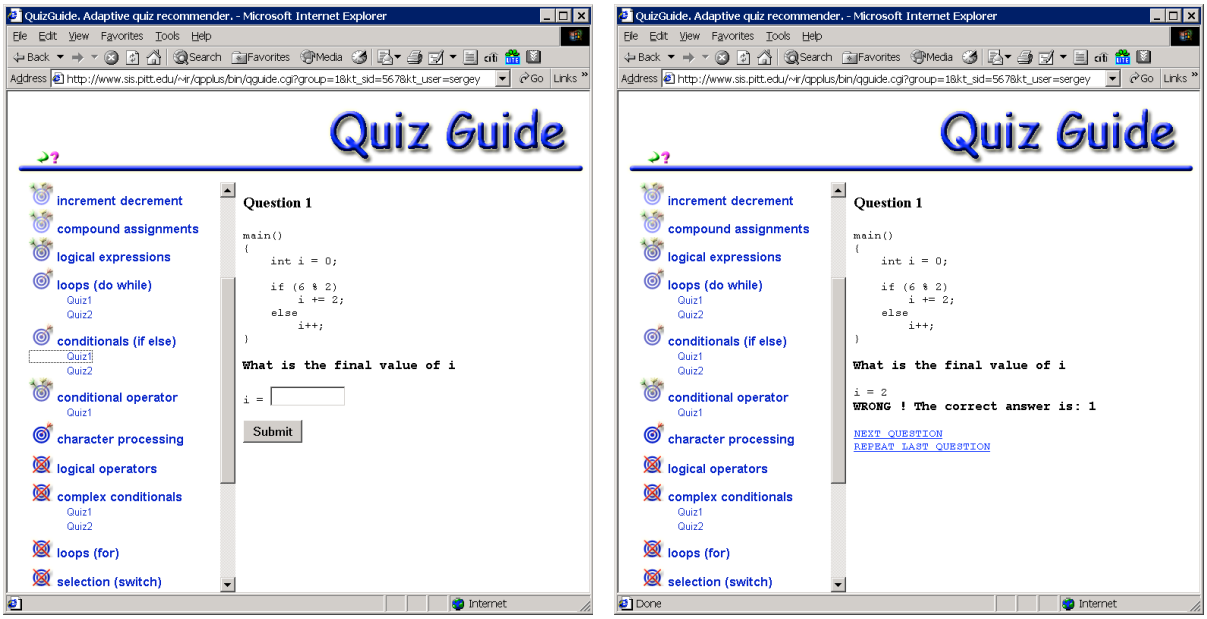


Figure 1: Traditional access to QuizPACK quizzes via KnowledgeTree portal.

2 The QuizGuide Student Interface

The student interface of the QuizGuide system consists of two main parts: the quiz navigation area and the quiz presentation area. The *quiz navigation area* (left on Figures 2a and 2b) uses stretchtext technology to provide hyperlinks to 40 quizzes organized in 20 topics. When a student clicks on the topic name, the topic opens, expanding the links to quizzes available for this topic. A click on an opened topic collapses the list back. The student can open several topics at the same time. A click on a quiz link loads the first question of this quiz in the quiz presentation area. The use of stretchtext technology decreases the information load on the student by removing links to unrelated quizzes while allowing them to see "the whole picture".



a) **b)**

Figure 2: Student interface of QuizGuide.

In the *quiz presentation area* (right on Figures 2a and 2b) students answer the questions and receive feedback. There are two types of questions: "*what is the final value of an expression?*" and "*what will be printed?*" The feedback indicates whether the answer is correct or not; if it is not, the correct answer is presented (fig. 2b). When used for self-assessment, each incorrect answer challenges the student to proceed with analyzing the execution of the code fragment in a visual debugger and to understand the source of the error. After finishing their work with a question, students are provided with two options to continue. They can move on to the next question in the quiz or repeat the previous one (with a different parameter value). The student's work with the quiz area of QuizGuide is supported by the QuizPACK server. Its underlying mechanisms are explained in more details in (Pathak & Brusilovsky, 2002).

Adaptive navigation support is provided in the quiz navigation area by adaptive icons shown to the left of each topic. QuizGuide adapts to the most critical characteristics of the user: the knowledge level and the learning goal. To reflect both the goal and knowledge relevance of each topic in one icon, QuizGuide uses the "target-arrow" abstraction (Figure 3). The number of arrows in the target reflects the level of knowledge the student has for the topic: the more arrows the target has, the higher the level is. The intensity of the target's color shows the relevance of the topic to the current learning goal: the more intense the color is, the more relevant the topic is. Topics that are not ready to be studied are annotated with the crossed target. In total, there are four levels of knowledge (from zero to three arrows) and four levels of goal relevance (not-ready, important, less-important and non-important). For example, icon 3a indicates a topic that belongs to an unimportant learning goal, and for which the student has the maximum level of knowledge. The icon 3b would indicates a topic related to the current learning goal; the student has a high level of knowledge for this topic. The icon 3c indicates that the topic is not ready to be studied. Since the student goals and knowledge are constantly changing, different icons will be shown practically each time the student accesses QuizGuide. To reflect changes in the user model that happened during the same session, the student clicks on the refresh icon.

a) **b)** **c)**

Figure 3: "Target-arrow" abstraction in QuizGuide.

3 The System Architecture and Mechanisms

Architecturally, the QuizGuide system is implemented as an adaptive value-added service (Brusilovsky, Sosnovsky & Yudelson, 2004) standing between the original QuizPACK system and the learning portal (Figure 4). QuizGuide generates the list of annotated hyperlinks to quizzes and provides an access to QuizPACK through the quiz presentation area. QuizPACK is responsible for the question presentation, checking student's answers, feedback generation, and the update of the central user model (UM). When generating the adaptive quiz navigation part, QuizGuide sends a request to UM and gets current values of the knowledge level and the learning goal for all topics. As a value-added service, QuizGuide does not change the original QuizPACK, but adds an extra level of support. As a result, students can access the same quizzes in two modes: in adaptive mode, by using the new QuizGuide interface (Figure 2) and in the traditional non-adaptive mode by calling QuizPACK quizzes directly from the KnowledgeTree portal (Figure 1). Below we present the adaptation mechanisms in more detail.

QuizGuide uses the UM server CUMULATE, which is a component of our distributed E-Learning architecture (Brusilovsky, 2004). Here we give only a brief overview of CUMULATE. The server maintains two levels of UM. The low level stores transaction data collected from applications that communicates with the student. This level is updated every time the student performs an action tracked by the application. The upper level of UM uses inference agents to transform transaction history into a set of name-value pairs that reflect various features of the user. The state of the upper level user model can be requested by any application that needs information about user features through an HTTP-based interface. The answer is provided in XML format.

In the case of QuizPACK, it tracks and sends to the UM all question answers. Once QuizPACK processes the answer, it sends the UM the time of transaction, application id, student id, session id, quiz name, question number, and the correctness (0 or 1). This data is processed by CUMULATE's *topic-based inference agent*. This domain-independent inference agent uses lower level transaction data and authored relationships between topics and learning activities to infer the user knowledge level for each topic (0 to 1). To use this inference agent in QuizGuide, we divided the body of C programming knowledge into 22 topics and associated each of the 44 developed quizzes with one of these topics.

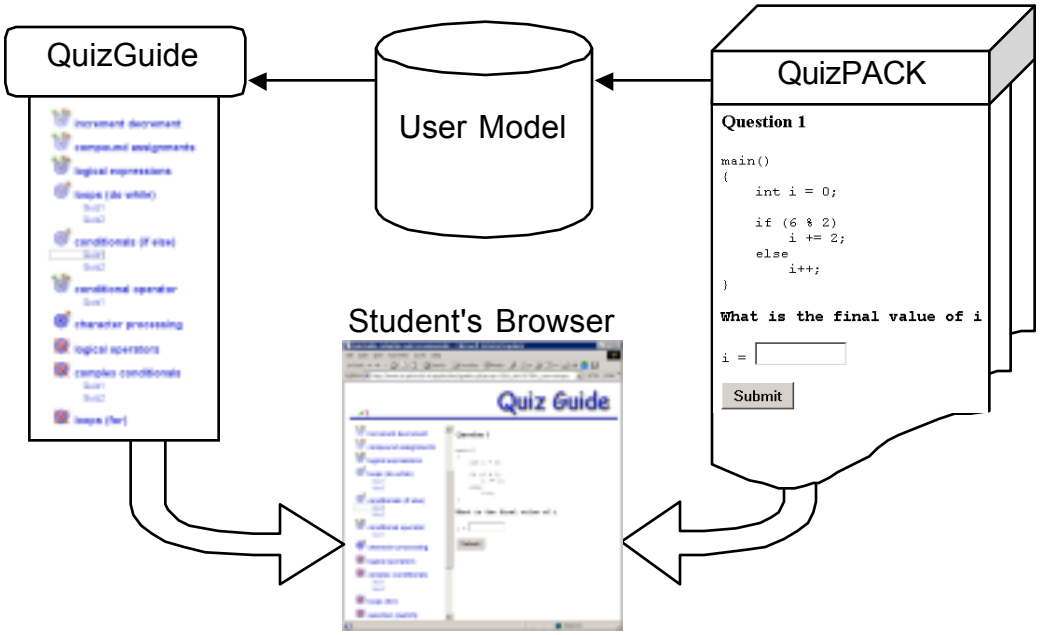


Figure 4: QuizGuide-QuizPACK interaction.

.

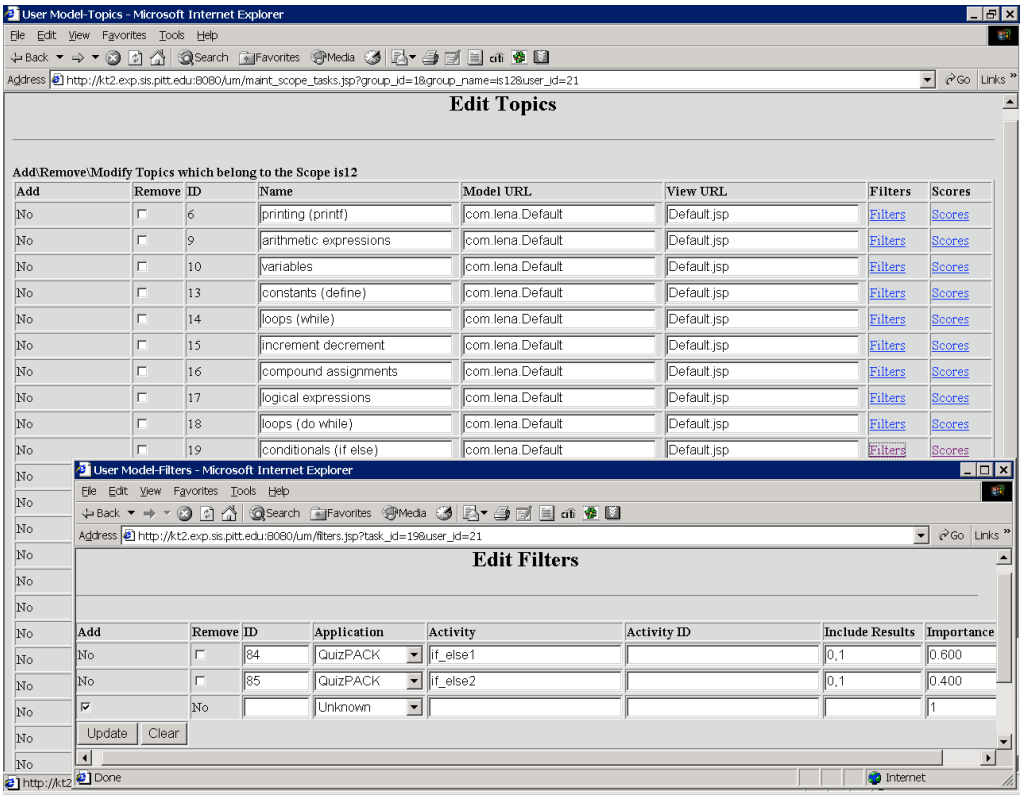


Figure 5: UM author interface for topic edition.

Figure 5 demonstrates CUMULATE's interface for defining topics and associating educational activities with topics. This is the place where QuizPACK and QuizGuide meet. The knowledge level of each topic is calculated as a weighted sum of the student's performance with corresponding QuizPACK questions. In

turn, QuizGuide uses the calculated knowledge levels as a source for adaptation. Figure 5 shows that the topic *conditionals (if else)* has two associated activities that are both QuizPACK quizzes - *if_else1* and *if_else2*. Note that the *Activity_id* (sub-activity) fields are empty, since in our current settings all questions of each quiz belong to the same topic. In general, the connection between topics and activities can be defined on the lowest sub-activity level (i.e., questions). The next field (*Include Results*) simply describes what results the UM has to consider for this activity. In the case of QuizPACK these results can be 0 (wrong answer) and 1(correct answer), but this data is application-dependent. Finally, *importance* of the activity for the topic defines the expected contribution of the activity for learning the topic. For the quizzes we distinguish three different levels of importance that correspond to three levels on the quiz complexity (Figure 5 shows two of them). In total, we have defined 22 topics for our course. These topics form 15 learning goals that correspond to lectures in the class. Topics are connected to each other with prerequisite relations. Different topics have from 1 to 3 quizzes with 3 to 5 questions in each quiz. In total, there are 44 quizzes with more than 150 questions.

The topic-based user modeling in CUMULATE uses a relatively simple method of knowledge calculation. For all activities belonging to the topic, it calculates the weighted average score of reported results. Note that each attempt to use a learning activity registered in the lower level of UM is counted separately. Thus, to calculate student performance on quiz *if_else1*, the system will average all attempts to answer all questions from this quiz. The weights for quizzes of different complexities were developed so that the students could achieve the maximum level by dealing only with quizzes of the first two levels of complexity. We do not require students to solve all quizzes perfectly to achieve mastery on the topic.

The current student knowledge level of all topics inferred by the agent is requested by QuizGuide through UM query interface and used to adaptively generate the quiz navigation area. Figure 6 shows the answer to a query requesting student knowledge for the *conditionals (if else)* topic. Figure 2 demonstrates the interface of QuizGuide for the same user, in the same learning situation. Comparing the knowledge of each topic with three pre-determined thresholds, the system selects an icon with the proper number of arrows (zero to three). Current thresholds are *0.1*, *0.3*, and *0.5*. As you can see from figure 5 the user *sergey* has the score *0.140* for the topic *conditionals (if else)*. Thus, the icon annotating the topic *conditionals (if else)* has one arrow that corresponds to the second knowledge level (Figure 2). The choice of one of four "goal intensity" levels for each topic is driven by a simpler time-based mechanism: The topic of the most recent lecture is most important, the topics of earlier lectures are less important, the topics of future lectures are not ready. QuizGuide does not forbid students to work with any topic, it just reflects the current knowledge level and goal state. A student can take any quiz belonging to any "non-ready" learning goal, and when the time comes for this topic to become the current learning goal, previous student's work with it will be taken into account.

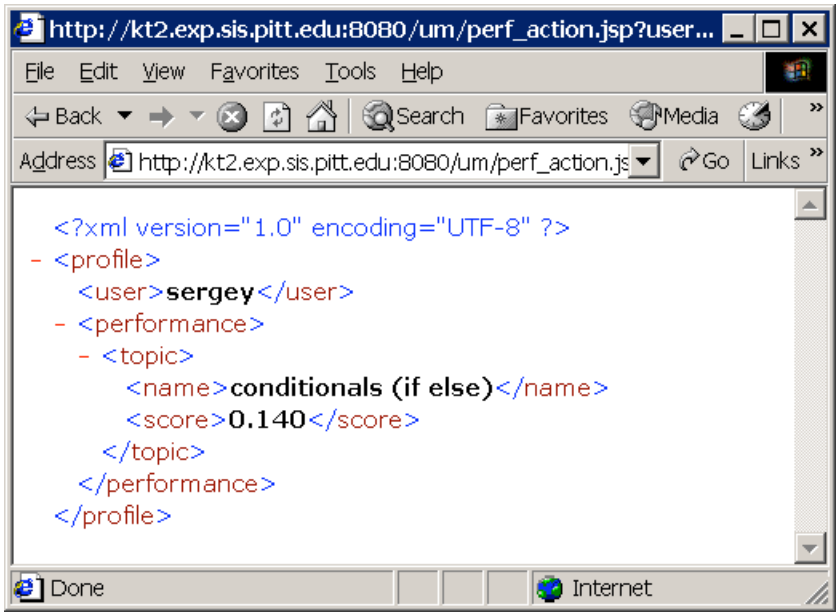


Figure 6: UM response to a topic performance query

4 The Evaluation of QuizGuide

In the Spring 2004 semester, QuizGuide was used in the context of an undergraduate course *Introduction to Programming,* taught in the School of Information Sciences at the University of Pittsburgh. We made QuizGuide available to the students after the Midterm exam. For the first part of the course the students could access QuizPACK quizzes without adaptive navigation support through the KnowledgeTree. During the second half, both adaptive and non-adaptive accesses to quizzes were available to students. Note that both QuizGuide and KnowledgeTree provide access to the *same set of quizzes*. The QuizPACK performance recording mechanism does not depend on the way of access, so data obtained from both QuizPACK and QuizGuide is equally considered for the user modeling.

From the log data stored in the lower level student model it was clear whether a quiz was called from the non-adaptive KnowledgeTree or adaptive QuizGuide. Analyzing this data, we noticed that about 1/3 of students had switched to a predominant QuizGuide use after its introduction, however, about 30% of students continued to use non-adaptive QuizPACK, and the rest used both systems in parallel (using the adaptive version during some sessions and non-adaptive during other sessions). The use of two systems in the same class allowed us to see the difference that was provided by the adaptive navigation support. We observed the two visible effects of QuizGuide. First, the average session length of a QuizGuide session is 26 questions, while the average session length for non-adaptive quiz access is only 10 questions. It means that adding adaptive navigation support to self-assessment quizzes motivates students to work with about 2.5 times more questions. Second, the percentage of correct answers for the students who mainly used QuizPACK is only 32%, while the students who had switched to QuizGuide and used it regularly achieved 43% correctness (a 1/3 increase). We think that the increase was caused by two factors. First, adaptive navigation support helped students see the quizzes that they were ready to take. Second, QuizGuide motivated students to work with a question after the first correct attempt when they have achieved some understanding of the topic and have a better chance to answer the question with different parameters correctly.

Interesting data can be discovered by comparing the current term and two previous terms (when students used only non-adaptive QuizPACK). As we noted above, we have been using QuizPACK as a self-assessment tool for this course for more than 2 years. Each term, 25 to 50 students of the same introductory programming course were able to use QuizPACK for self-assessment. During the last three terms the context of QuizPACK use was exactly the same, so we can assume that the students had the same external (i.e., tool-independent) motivation to use the tool. The only difference in the last term was the introduction of QuizGuide. Comparing the data, we noticed that QuizGuide had provided a strong internal motivation to work with quizzes: the use of QuizPACK quizzes has increased on 65%. While the average number of questions attempted by students during the Spring 2004 QuizGuide-enabled term is 181, the same number for students of Spring and Fall 2003 terms is only 110. The increase in the number of attempted questions has been achieved by working with more questions distributed over the course and by working more persistently with each question. The course coverage (percentage of questions attempted at least once) rose from 38% to 44% in QuizGuide term and the average number of attempts per question rose from 1.69 to 2.63.

More comprehensive and more persistent work with quizzes has resulted in the increase of knowledge gain, a parameter that we compute to isolate the past experience factor when assessing student progress. At the beginning and at the end of each term we have administered the same test of 10 questions (with different parameters) covering the whole content of the course. We have calculated the *knowledge gain* as the difference between post-test and pre-test scores. For the QuizGuide term the average knowledge gain was about 6.5. For the two previous terms is was less then 5.1. This is a very impressive increase.

To perform subjective evaluation, at the end of the term we administered a non-mandatory questionnaire that solicited students' opinions about key features of the system. Out of 27 students in the class, 26 filled the questionnaire. Some early data obtained from processing the answers is provided on Figure 7. As we can see, more then 80% of students considered the QuizPACK/QuizGuide system helpful, stating that it has helped or significantly helped them during the course (general attitude). Two principal features of QuizGuide (*parameterized question generation* and *adaptive annotation of quizzes*) were evaluated positively or strongly positively by the vast majority of the students (about 90% for *parameterized question generation* and about 80% *adaptive annotation of quizzes*). Answering a multiple-selection question about system features that they do not like, the largest fraction of students (35%) checked the "target-arrow" annotation. However, only one student proposed the alternative way: "*…a bar graph or numbers might be better for quick, simple recognition*". At the same time almost 25% thought that the system does not need any improvement. It's interesting to compare students' suggestions on the future development of the system before and after the introduction of QuizGuide. While in the Fall of 2003, 58% of students pointed out that the most important

feature to add is an adaptive recommender for self-assessment quizzes, in the Fall 2004 term only 31% said that the improvement of the current recommender (QuizGuide) is the most urgent need.

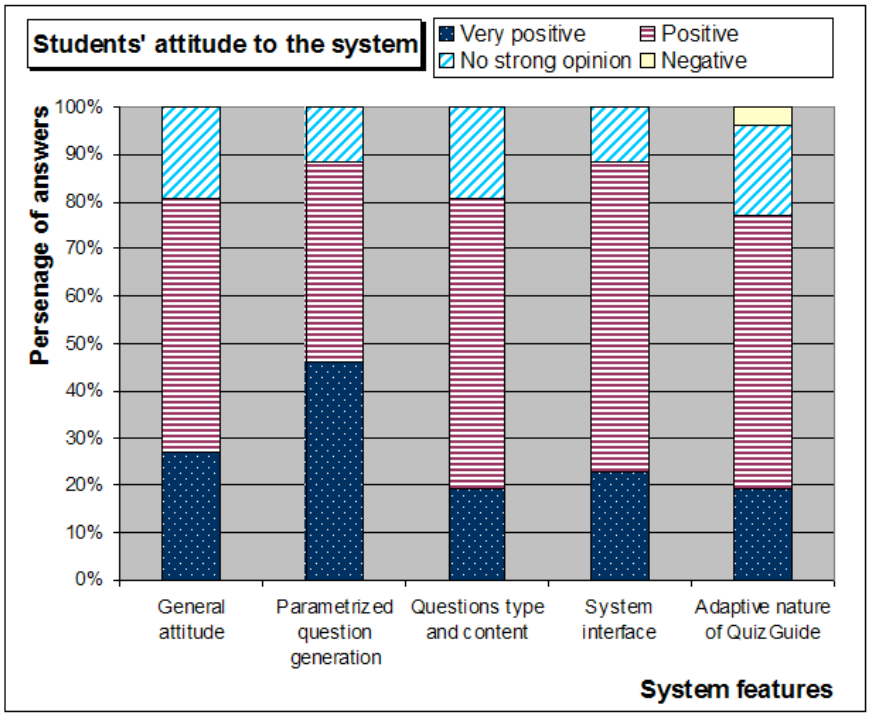


Figure 7: Subjective student evaluation of different features of QuizGuide

5 Summary and Future Work

We have presented the architecture and the evaluation of the QuizGuide system that helps students in selecting most relevant self-assessment quizzes in the context of a regular college introductory programming course. QuizGuide uses adaptive annotation technology to show the students their current knowledge level for each course topic and current relevance of these topics. Despite relatively simple user modeling and adaptation techniques used in QuizGuide, the system has achieved a remarkable impact on student learning and performance. With QuizGuide, the students have explored more questions, worked with questions more persistently, and accessed a larger variety of questions. The increase of their work with the system resulted in the larger increase of their knowledge at the end of the course.

To understand the value of adaptive navigation support in the context of self-assessment, we are going to perform more exhaustive evaluation of QuizGuide. The current architecture of QuizGuide/QuizPACK provides parallel adaptive and non-adaptive access to the same set of quizzes. This opens an exceptional opportunity for the evaluation of adaptive navigation support mechanism. Since adaptive annotation is implemented as a value-added service, the non-adaptive version of the system that is used as a control condition is a fully functioning and educationally meaningful application and not a partially disabled adaptive version. It allows us to avoid some concerns in evaluating adaptive systems expressed in (De Bra, 2000). In addition, we are interested in comparing different user modeling and adaptation approaches. The question here is whether the more advanced approaches that we have used in the past in ISIS-Tutor (Brusilovsky & Pesin, 1998) and InterBook (Brusilovsky & Eklund, 1998) provide a better value than the relatively simple approach used currently in QuizGuide.

6 References

Brusilovsky, P. (2001) Adaptive hypermedia. *User Modeling and User Adapted Interaction* 11 (1/2), 87-110, also available at http://www.wkap.nl/oasis.htm/270983.

Brusilovsky, P. (2004) KnowledgeTree: A distributed architecture for adaptive e-learning. In: Proceedings of The Thirteenth International World Wide Web Conference, WWW 2004 (Alternate track papers and posters), New York, NY, 17-22 May, 2004, ACM Press, pp. 104-113.

Brusilovsky, P. and Eklund, J. (1998) A study of user-model based link annotation in educational hypermedia. In P. Carlson (ed.) *Journal of Universal Computer Science* 4 (4), Special Issue on Assessment Issues for Educational Software, 429-448, also available at http://www.iicm.edu/jucs_4_4/a_study_of_user.

Brusilovsky, P. and Miller, P. (2001) Course Delivery Systems for the Virtual University. In: T. Tschang and T. Della Senta (eds.): *Access to Knowledge: New Information Technologies and the Emergence of the Virtual University*. Amsterdam: Elsevier Science, pp. 167-206., also available at http://www2.sis.pitt.edu/~peterb/papers/UNU.html.

Brusilovsky, P. and Pesin, L. (1998) Adaptive navigation support in educational hypermedia: An evaluation of the ISIS-Tutor. *Journal of Computing and Information Technology* 6 (1), 27-38.

Brusilovsky, P., Sosnovsky, S., and Yudelson, M. (2004) Adaptive Hypermedia Services for E-Learning. In: Proceedings of Workshop on Applying Adaptive Hypermedia Techniques to Service Oriented Environments at the Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2004), Eindhoven, the Netherlands, .

Carmona, C., Bueno, D., Guzman, E., and Conejo, R. (2002) SIGUE: Making Web Courses Adaptive. In: P. De Bra, P. Brusilovsky and R. Conejo (eds.) Proceedings of Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2002) Proceedings, Málaga, Spain, May 29-31, 2002, pp. 376-379.

De Bra, P. (2000) Pros and Cons of Adaptive Hypermedia in Web-Based Education. *Journal on CyberPsychology and Behavior* 3 (1), 71-77.

De Bra, P. M. E. (1996) Teaching Hypertext and Hypermedia through the Web. *Journal of Universal Computer Science* 2 (12), 797-804, also available at http://www.iicm.edu/jucs_2_12/teaching_hypertext_and_hypermedia.

Henze, N. and Nejdl, W. (2001) Adaptation in open corpus hypermedia. In P. Brusilovsky and C. Peylo (eds.), *International Journal of Artificial Intelligence in Education* 12 (4), Special Issue on Special Issue on Adaptive and Intelligent Web-based Educational Systems, 325-350, also available at http://cbl.leeds.ac.uk/ijaied/abstracts/Vol_12/henze.html.

Kashy, E., Thoennessen, M., Tsai, Y., Davis, N. E., and Wolfe, S. L. (1997) Using networked tools to enhanse student success rates in large classes. In: Proceedings of 27th ASEE/IEEE Frontiers in Education Conference, Pittsburgh, PA, November 5-8, 1997, Stipes Publishing L.L.C., pp. 233-237.

Papanikolaou, K. A., Grigoriadou, M., Kornilakis, H., and Magoulas, G. D. (2003) Personalising the interaction in a Web-based Educational Hypermedia System: the case of INSPIRE. *User Modeling and User Adapted Interaction* 13 (3), 213-267 .

Pathak, S. and Brusilovsky, P. (2002) Assessing Student Programming Knowledge with Web-based Dynamic Parameterized Quizzes. In: P. Barker and S. Rebelsky (eds.) Proceedings of ED-MEDIA'2002 - World Conference on Educational Multimedia, Hypermedia and Telecommunications, Denver, CO, June 24-29, 2002, AACE, pp. 1548-1553.

Sosnovsky, S., Shcherbinina, O., and Brusilovsky, P. (2003) Web-based parameterized questions as a tool for learning. In: A. Rossett (ed.) Proceedings of World Conference on E-Learning, E-Learn 2003, Phoenix, AZ, USA, November 7-11, 2003, AACE, pp. 309-316.

Weber, G. and Brusilovsky, P. (2001) ELM-ART: An adaptive versatile system for Web-based instruction. In P. Brusilovsky and C. Peylo (eds.), *International Journal of Artificial Intelligence in Education* 12 (4), Special Issue on Adaptive and Intelligent Web-based Educational Systems, 351-384, also available at http://cbl.leeds.ac.uk/ijaied/abstracts/Vol_12/weber.html.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0310576.