

Recommendation Engine for an Online Drill System

Toshikazu Iitaka^(✉)

Kumamoto Gakuen University, 2-5-1 Oe, Chuo-ku Kumamoto, 862-8680, Japan
iitaka2@yahoo.co.jp

Abstract. This paper presents a recommendation engine I have developed for an online drill system. The engine provides suitable quizzes for users, helping users learn effectively. The paper describes the features and effects of the recommendation engine.

Keywords: E-learning · Recommendation engine · Online drill system

1 Objective

I have developed an online examination (e-learning) system that includes practice drill functionality and a recommendation engine that recommends appropriate quizzes for each user automatically. The objective of this paper is to discuss the research background and the effects of the recommendation engine.

As shown in [2], the positive effect of an online drill system itself is confirmed. Further progress can be achieved by enabling each user to receive customized information by developing a recommendation engine that can provide appropriate quizzes for individual users. First, we aim to understand the background more precisely.

Second, the ongoing use of the online drill system in courses has prompted the implementation of a recommendation engine. We aim to prove the effect of the recommendation engine by using the data of the courses.

This paper reviews the significance and background of the research, provides the specifications of the recommendation engine, and then analyzes the recommendation engine's effects.

2 Background and Significance

This section describes the background of this research by describing the benefit of a recommendation engine's implementation in a drill system. The significance of the research derives from this benefit. Describing the use of the recommendation engine also sheds light on its significance.

People often read small paperback books of drills, while they commute to work. T. Iitaka stated that web applications for drills designed for mobile phones can be used instead of books [2]. Hence, the web application I have developed has drill functionality.

Recommendation engines recommend appropriate information based on preserved data. Some recommendation engines simply recommend popular information such as

articles that many people have evaluated. However, proper recommendations must be customized for each user. Therefore, simple recommendation systems cannot provide each individual user with appropriate and personalized information. An algorithm that measures similarity is required to provide appropriate information. The proposed system recommends articles evaluated by other users with similar tendencies.

However, as shown in Fig. 1, measuring similarity often takes a significant amount of time. The requirement that recommendation data be updated periodically, e.g., once a day, interferes with real-time recommendation.

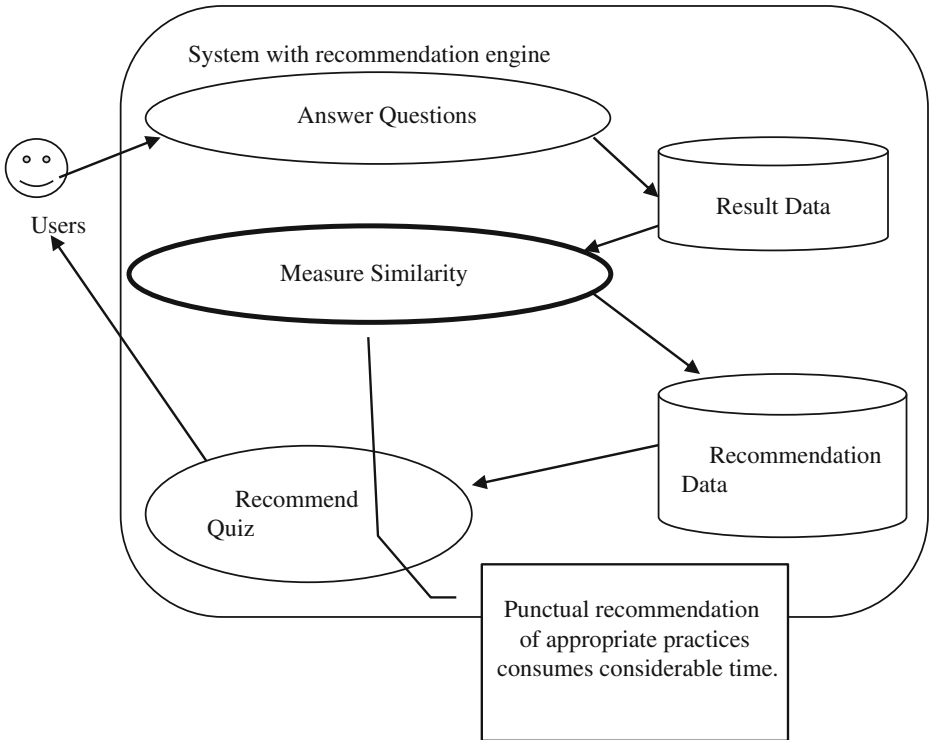


Fig. 1. Recommendation engine problems based on similarities

Consequently, recommendation engines often recommend articles that users have already read and do not need to read again. To address this problem, Itaka has suggested the use of a “restriction list,” which can be generated as quickly as a simple recommendation [3]. Both the simple recommendation data and the “restriction list” can be created using simple SQL. Here the “restriction list” is a list of quizzes that users have completed correctly; thus, these quizzes need not be attempted again.

Consequently, recommendation engines with “restriction lists” have been included in the proposed quiz system. The effects of the recommendation engines have been analyzed briefly [7] (Fig. 2).

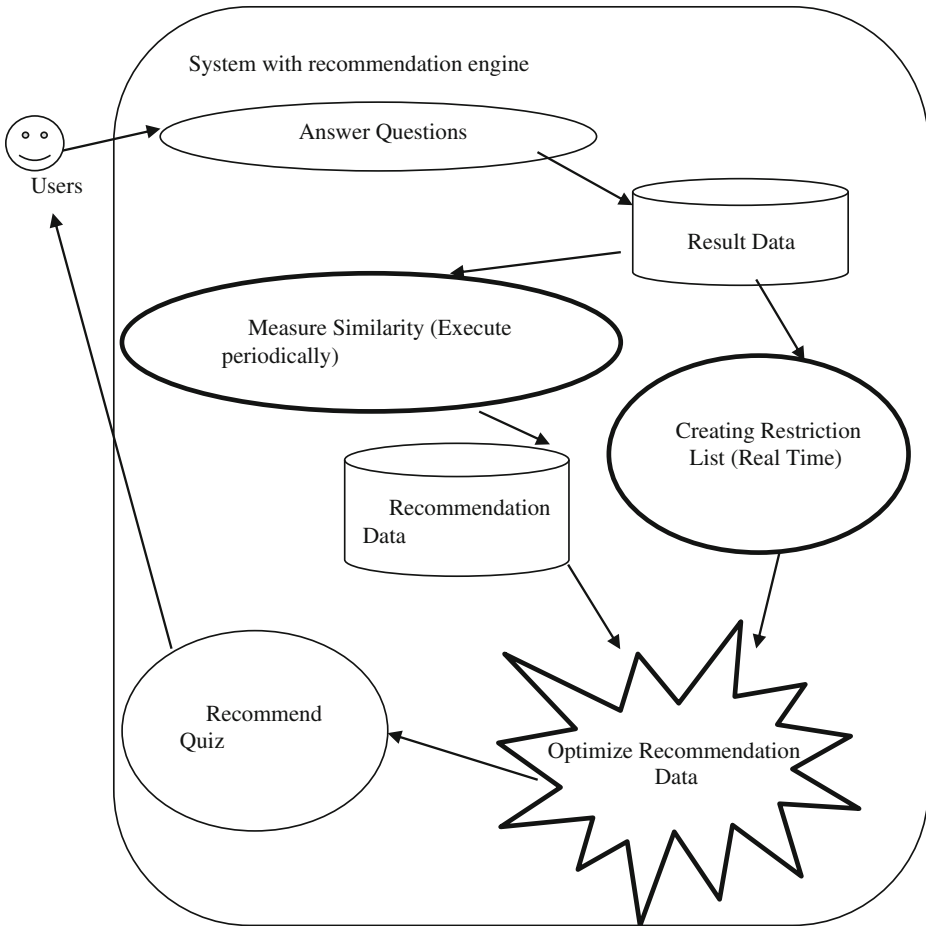


Fig. 2. Recommendation engine with restriction list

Now, the detailed features of the recommendation engine are described. As shown by [7], the basic data of the recommendation engine are the following.

- (1) Data regarding evaluation on quizzes
- (2) Flag data (Users can check quizzes that they want to try repeatedly)
- (3) Answer data (These data tell us whether a user has given the correct answer)

The system can provide simple recommendations from (1) and (2). In other words, (1) and (2) identifies popular quizzes. Each type of data (1–3) can provide similarity based recommendations.

Similarity data are calculated periodically from data (1), (2), and (3), using Pearson’s and Tanimoto’s correlation coefficients. The similarity data are optimized in real time and provided as recommendation data. Recommendations from data (3), in particular, can predict which quizzes are needed for specific users to reinforce the users’ learning.

Administrators of the system use the web page shown in Fig. 3 for periodical calculations.

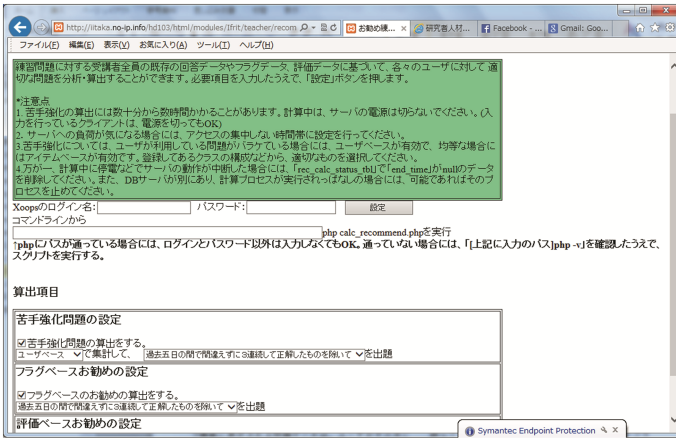


Fig. 3. Web page for creating similarity data

The program executed from this web page runs as a background program. Hence, the creation of similarity data does not interrupt normal system use.

The recommendation data (recommended quizzes) are shown on the following pages.

- (1) Users' top page
- (2) Special page for personalized training
- (3) Page after answering a quiz

As shown in Fig. 4, two different recommendations are shown on the user's top page. One is a list of popular quizzes. The other is a list of quizzes recommended based on similarity.

There is also a special page for personalized training, shown in Fig. 5.

Furthermore, various lists of recommendations are shown on the page that appears after answering a quiz. The page shows whether the user's answer is correct. The correct answer and explanation are also shown on this page. Users can evaluate and check the quiz on this page (The system allows users to try only checked quizzes repeatedly). This page can show a list of quizzes, when the user has answered incorrectly. Users who have given incorrect answers to the quiz that has just been attempted tend to give incorrect answers to the quizzes on the list. Different lists are also shown when users check or evaluate the quiz just attempted. Users who check the quiz also tend to check the quizzes on the list displayed. Users who evaluate the quiz also tend to evaluate the quizzes on the list displayed (Fig. 6).

A recommendation engine that provides personalized information can be expected to be beneficial for higher education in general, because, as is often said, personalized education is increasingly needed for higher education [13]. Hence, this research must be significant, because such a recommendation engine can enable us



Fig. 4. User's top page

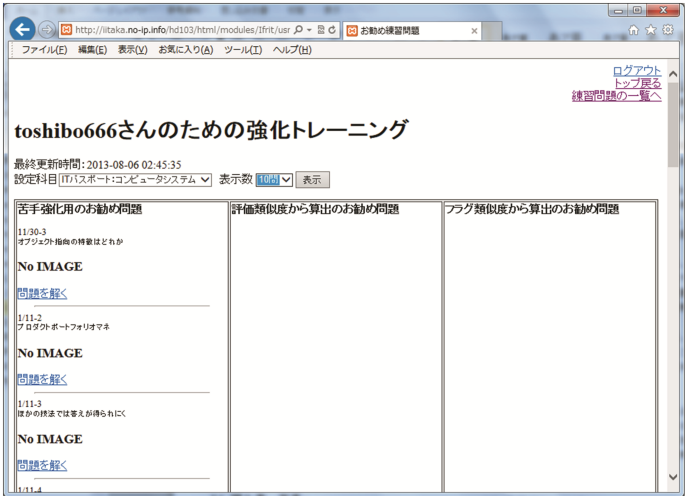


Fig. 5. Special page for personalized training

to achieve personalized education for effective e-learning. However, statistical analysis is required to prove this significance. Detailed analysis of the effects of the recommendation engine is required. The next section shows the detailed analysis.

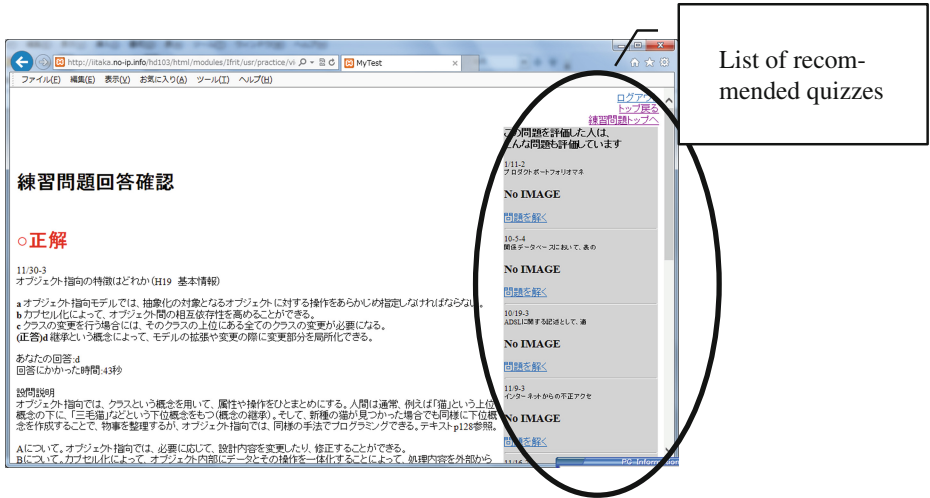


Fig. 6. Page that appears after answering a quiz

3 Methods

This section provides the statistical analysis of the recommendation engine based on the data from two courses in which the e-learning system is implemented.

The e-learning system with the recommendation engine is used in some Japanese IT classes. The classes prepare participants for a Japanese IT qualifying exam (i.e., IT passport). The effects of the recommendation engine are assessed in terms of the satisfaction and score of the periodical examinations.

Table 1 shows the course data.

Table 1. Course Data

	Participants	Participants taking exam
Course 1	144	129
Course 2	150	127

As shown in Table 1, Courses 1 and 2 have 144 and 150 participants, respectively. First, we check the satisfaction of the participants.

As shown in Fig. 7, more than 60 % of participants in both courses answered that the recommendation engine was useful. In Courses 1 and 2, 68 % and 67 % of participants, respectively, considered the recommendation engine useful.

Second, we estimate the effect of the recommendation engine using statistical analysis, with data drawn from the periodical examinations held in July 2013. Each examination consists of 40 questions previously set for the IT Passport (Fig. 8).

Now, we examine the basic data of the examination.

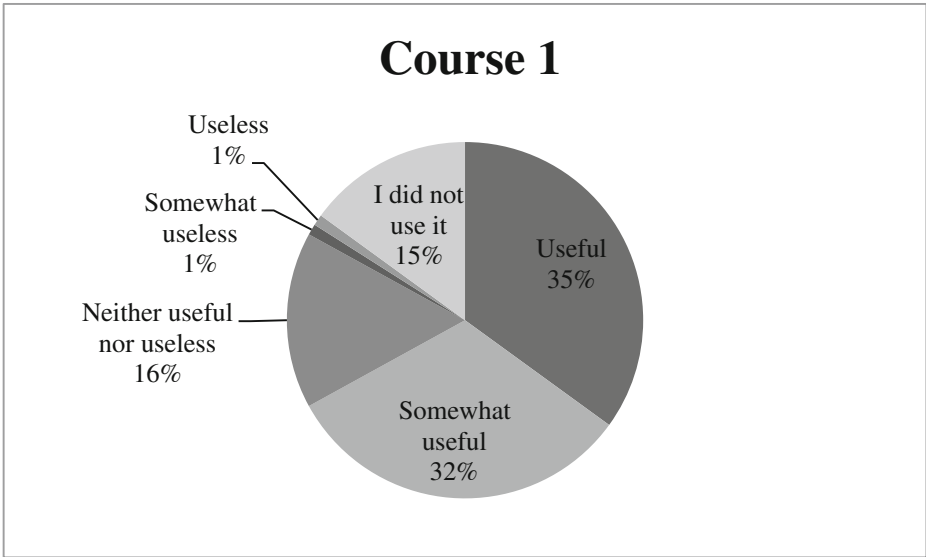


Fig. 7. Course 1 satisfaction

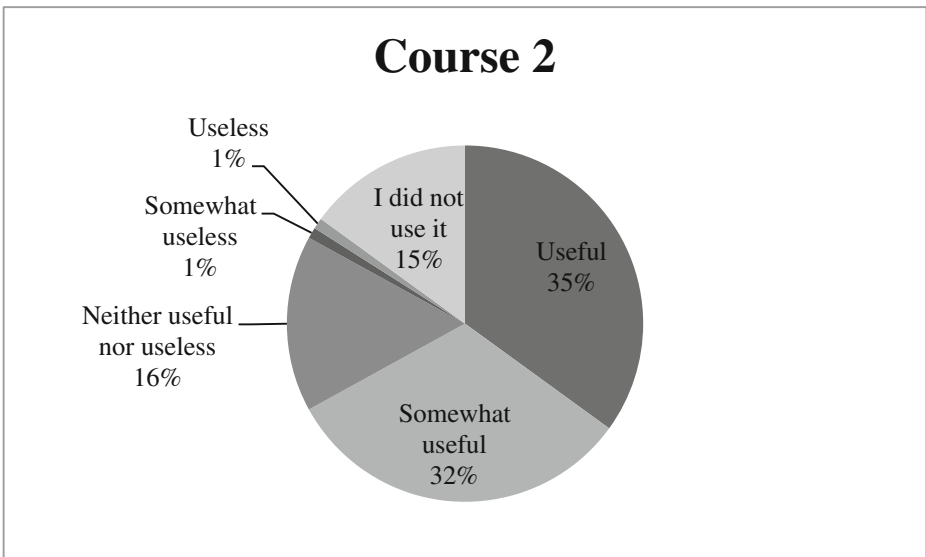


Fig. 8. Course 2 satisfaction

As shown in Table 2, skewness and kurtosis of scores in both courses are below 2.0. Hence, we can deal with the data distribution as a normal distribution. The average score of participants who used the recommendation engine is higher than that of the participants who did not. The average scores for Course 1 and 2 were 27.4 and 24.2, respectively. The users of the recommendation engine in Course 1 received 28.9 on average,

while the average score of the users who did not use it was 25.8. The users in Course 2 received 25.3 on average, while the average score of users who did not use it was 23.4.

Table 2. Results of examinations

Course 1	Average score	27.4
	Standard deviation	6.6
	Skewness	-0.6
	Kurtosis	-0.1
	Number of users	69
	Average score of users	28.9
	Standard deviation of users	6.20
	Participants who did not use the recommendation engine	61
	Average score of participants who did not use the recommendation engine	25.8
	Standard deviation of participants who did not use the recommendation engine	6.80
Course 2	Average score	24.2
	Standard deviation	6.8
	Skewness	-0.1
	Kurtosis	-0.6
	Number of users	54
	Average score of users	25.3
	Standard deviation of users	6.22
	Participants who did not use the recommendation engine	73
	Average score of participants who did not use the recommendation engine	23.4
	Standard deviation of participants who did not use the recommendation engine	7.10

Users of the recommendation engine in both courses received higher average scores. However, the statistical significance of the result is not perfectly confirmed, when we check the difference between users and non-users with the t-test. Only the difference in Course 1 was statistically significant ($t(127) = 2.17, p < 0.001$). The

difference in Course 2 was not statistically significant ($t(125) = 1.53, ns$). The difference is statistically significant only in the course in which the average score is relatively high. This tendency recurs in many courses.

This analysis suggests that the recommendation engine is more effective for relatively skillful users. The recommendation engine might be more effective in creating personalized recommendation data for users who use the drill system itself more often. Because the recommendation engine requires data for calculating recommendation data, users must use the drill system often. The average score is higher if the users use the drill system more frequently. Therefore, it is natural that the recommendation engine shows a clear effect only in Course 1, in which the average score is higher.

We must examine the use of the drill system to test this hypothesis. If the users in Course 1 used the system more often than the Users in Course 2 did, it is natural that the difference is statistically significant only in Course 1. First, we look at the basic data regarding drill system usage.

As shown in Table 3, there are 116 users in Course 1 out of 144 participants. More than 80 % of the participants in Course 1 used the drill system. There are 150 participants in Course 2 more than in Course 1. However, there are only 104 users in Course 2. Only 69.33 % of participants in Course 2 used the drill system. More important are the frequencies of drill system usage. Users in Course 1 used the drill system 219.8 times on average, while users in Course 2 used it only 148 times on average.

Table 3. Basic data of drill system’s use

Course 1	Number of users	116
	Utilization rate	80.56 %
	Usage count	219.8
	Standard deviation	368.53
Course 2	Number of users	104
	Utilization rate	69.33 %
	Usage count	148.05
	Standard deviation	217.23

The hypothesis is confirmed to some degree. However, the assertion of this section is reinforced if the drill system was more effective in Course 1.

As shown in Table 4, the average score of users of the drill system in Course 1 was 28, while the average score of the participants who did not use the drill system was 21.92. The average score of users in Course 2 was 24.38, while the average score of the participants who did not use the drill system was 23.5. Users in both courses tend to receive higher marks than the participants who did not use the system. However, the difference between the users and the other participants is remarkable in Course 1, in which the difference was more than six points.

Table 4. Comparison of average scores

Course 1	Average score of participants who did not use the drill system	21.92
	Average score of users	28
Course 2	Average score of participants who did not use the drill system	23.5
	Average score of users	24.38

As mentioned previously, it is natural that recommendation engines based on similarity are effective only when the systems that provide data for them are used frequently. Reliable similarity data can be created, only if there are sufficient data. The analysis of the data of Table 4 might be telling us that this phenomenon is occurring. Therefore, we can conclude that the recommendation engine is effective.

4 Discussion of Results

This paper described a recommendation engine for an online drill system that I have developed. The background, the features, and the effect of the recommendation engine were explained.

The paper revealed a weakness in popular recommendation engines, which cannot provide appropriate real-time recommendations. Providing recommendations based on similarity consumes too much time. However, the recommendation engine described in this paper overcomes this weakness by using a “restriction list.”

The statistical significance of using the recommendation engine was then confirmed. However, there are various algorithms that can create recommendation data. Consequently, there are many types of recommendation data, and we must consider the differences among these different types. However, there is currently insufficient data for such an analysis. In the future, we must gather more data to enable effective analysis of the difference.

References

1. Iitaka, T., Hirai, A.: CMS module for online testing. IEICE Tech. Rep. **107**(462), 25–29 (2008). Tokyo
2. Iitaka, T.: Mobile practice system using web item databank. IEICE Tech. Rep. **111**(237), 31–36 (2011). Tokyo
3. Iitaka, T.: CMS module for interactive lecture. IEICE Tech. Rep. **111**(394), 41–46 (2012). Tokyo
4. Iitaka, T.: Practice system using collective intelligence and restriction list. IEICE Tech. Rep. **111**(473), 167–172 (2012). Tokyo
5. Iitaka, T.: Introduction of the CMS module for interactive lecture. IEICE Tech. Rep. **111**(166), 35–40 (2012). Tokyo

6. Iitaka, T.: ARS module of contents management system using cell phones. In: Marcus, A. (ed.) DUXU 2013, Part IV. LNCS, vol. 8015, pp. 682–690. Springer, Heidelberg (2013)
7. Iitaka, T.: Analysis on the effect of recommendation engine for online practice system. IEICE Tech. Rep. **113**(229), 23–28 (2013). Tokyo
8. Kakitani, S., Watanabe, H.: Development of the portal site for support of programming learning. IEICE Tech. Rep. **109**(387), 7–12 (2010). Tokyo
9. Saito, Y., Hakamatsuka, A., Kuno, T., Suzuki, T., Kumazawa, H.: Development of attendance registration system using QR code and mobile phone ID. IEICE Tech. Rep. **109**(387), 13–18 (2009)
10. XoopsCube Web Page. <http://xoopscube.jp/>
11. Xoops Module E-frit Web Page. <http://iitaka.no-ip.info>
12. Xoops Module Nome Web Page. <http://iitaka.no-ip.info/norm/>
13. Yuan, F.: The Use of ICT in Higher Education: Media and Education, pp. 113–127. NHK Publication, Tokyo (2013)