

A Dynamic System to Generate Randomly Relocated Multiple Choice Questions Using Ready-made Arrays

Jae-Young Lee¹

¹*Department of Computer Engineering,
Hallym University, Gangwon, Korea 200-702
jylee@hallym.ac.kr*

Abstract

In this paper, we proposed a system to generate randomly relocated multiple choice questions using ready-made arrays instead of random functions in order to reduce the time to make test set. Before making the test set, such as, multiple choice questions consisting of questions, their correct answers and their distractors, first predefined multiple choice questions sets are chosen and then the selected questions are relocated randomly using a ready-made array. Finally, multiple choices for each question are relocated using another ready-made array. The relocated answers for each question are sent to a correct answer array to check applicant's answers later. Applicants take the tests to choose answers for multiple choice questions to evaluate applicants' abilities. Thus, the method to make randomly relocated tests set using arrays is executed faster than one using random functions.

Keywords: *distance education, system to generate dynamic tests, automatic selection of question tests*

1. Introduction

Multiple-choice tests are sets of test items each of which consists of a question, the correct answer, and distractors. This type of test has proved to be an efficient tool for measuring students' achievement [1] and is used worldwide both for evaluation and diagnostics.

According to Questionmark Computing Ltd, who have licensed their Perception software to approximately three million users so far, 95% of their users employ this software to administrate multiple-choice tests [2].

Test questions and distractors in multiple choice tests are generated from electronic instructional documents by a computer-aided procedure and various NLP techniques including term extraction and shallow parsing, and using language resources such as a corpus and WordNet [2, 3]. Generating items of multiple choice question in automatic way is based on the Semantic Web standard technology OWL (Ontology Web Language) and it is independent of lexicons such as WordNet or other linguistic resources [4, 5].

Distance education is the education that enables both applicants to be evaluated and learners to study anytime anywhere. It has been considered as an alternative education to complement space problem and quality problem that occur in a traditional education. However, when evaluating the students' abilities anytime and anywhere, the most important thing keeps on fair evaluation. To do this, techniques to generate questions obtained from database are classified as two groups: automatic selection of test questions and dynamic generation of test questions. The automatic selection of test questions is the system that selects each complete question from database according to criteria that examiner decided [6-10]. The disadvantage of the system is that the positions of choices including the correct answer are always identical whenever applicants try to. The same

question and the same position of choices may have negative influence on evaluating applicants because good memory can help them with overestimating their abilities.

Dynamic generation of question tests, on the other hand, first extracts questions, correct answers, and distractors from database which has sets of components about incomplete questions. In the next process, the questions are randomly relocated and the choices, such as, a correct answer and distractors for each question, are randomly relocated using random functions [11-13]. This system has an advantage that offers different question tests whenever applicants try to take tests because the positions of questions as well as those of choices are changed but it has disadvantage that generating questions spends lots of time because of random functions.

In this paper, we proposed a system to generate randomly relocated multiple choice questions using ready-made arrays instead of random functions in order to reduce the time to make test set.

2. A System to make Multiple Choice Questions

In the system to make multiple choice question tests, the system consists of client, server, and database. The system receives information about multiple choice questions from the examiner and then saves them in the database. The process at the server makes question tests and sends it to client. The processes have two types: one is automatic selection of multiple choice question tests and the other is dynamic generation of the multiple choice question tests.

In automatic selection of question tests, the system consists of client and server with database in which the main process of the server is random selection of complete multiple choice questions from database, as shown in Figure 1. The each complete question is composed of one question, one correct answer, and three distractors. This question has immutable form because the positions of four choices are fixed, that is, not movable. The complete multiple choice questions mean that the answer of the same question from database is always identical position. These questions made by examiner are stored in database at the server side, and then the server selects a set of suitable question from database before sending it to the client for applicants. The questions are suitable to evaluate applicant's learning ability. But the disadvantage of the system is easy to overestimate applicant's ability when the applicant tries to take the same question more than two times, because the position of answer for the same question is always same.

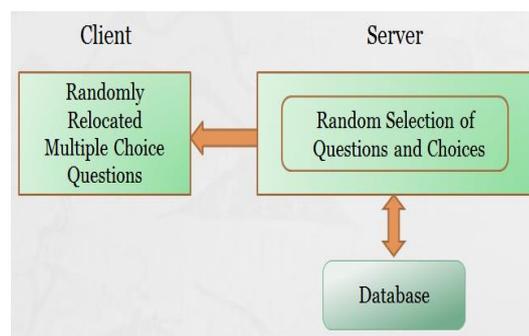


Figure 1. A System for Automatic Selection of Multiple Choice Questions

In dynamic generation of multiple choice questions, the system is also composed of client and server with database in which the first process randomly selects components of incomplete questions, such as, question sentences, their correct answers, and their distractors, and then the second one randomly relocates question

sentences selected at the previous process, and finally the third one randomly relocates four choices, as shown in Figure 2.

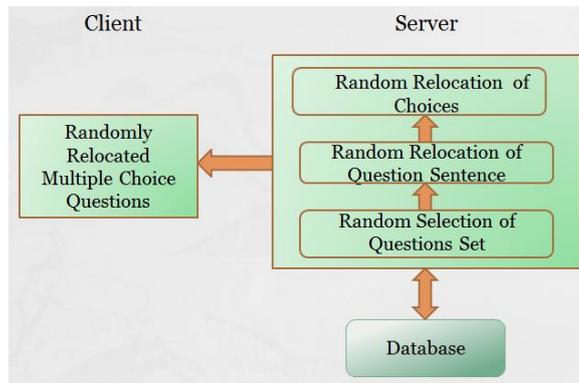


Figure 2. A System for Dynamic Selection of Multiple Choice Questions

In this paper, we propose the dynamic system for generation of randomly relocated multiple choice questions using ready-made arrays instead of random functions to reduce relocation time.

3. Algorithm of Relocation for Multiple Choice Questions using Ready-Made Arrays

In client-server system to make randomly relocated multiple choice questions, server to test lots of applicants should make lots of question tests with high speed. To reduce total time of making tests, relocation time need to be reduced since the time mainly affects the system. It needs much time to relocate some question sentence using random function. In the special case of evaluation system, applicants should solve questions within limited and so it is sufficient to less randomize. Instead of random function, array with less randomized numbers is used for relocation in this system. The advantage of the array is to generate relocated question quickly. Figure 3 shows a ready-made array with 10x10 in which the random number in each cell is determined arbitrarily. The numbers to generate using the array randomizes less than ones using random function but the former is suitable for relocation of questions because applicants must solve these questions within limited time.

0	2	3	1	4	9	6	8	7	5
2	0	1	4	3	8	9	6	5	7
4	3	0	2	1	6	7	5	8	9
3	1	4	0	2	7	5	9	6	8
1	4	2	3	0	5	8	7	9	6
9	6	8	0	2	3	7	5	1	4
8	9	6	2	0	1	5	7	4	3
6	7	5	4	3	0	8	9	2	1
7	5	9	3	1	4	6	8	0	2
5	8	7	1	4	2	9	6	3	0

Figure 3. A Ready-Made Array with the Size of 10x10 for Question Relocation

In the first phase to make tests in the client-server system, server extracts question and answer set in suitable size, for example, 10 questions, from database randomly and save

the questions and answers into question array and answer array, respectively. To relocate the 10 selected questions, the server should first choose one row, for example, the seventh row, from Figure 3. Figure 4 shows the process to relocate question sentences using the selected row of ready-made array.

In the second phase to relocate, for this example, load random number 8 of the 1-st column at the 7-th row for $i=0$ and then move Question 0 at index 0 into the 9-th position of a new question array. For $i=1$, load 9 of the 2-nd column and then move Question 1 at index 1 into the 10-th position of a new question array, for $i=2$, load 6 of the 3-rd column and then move Question 3 into the 7-th position, and so on, as shown in Figure 4. The relocation of their answers is the same as one of questions.

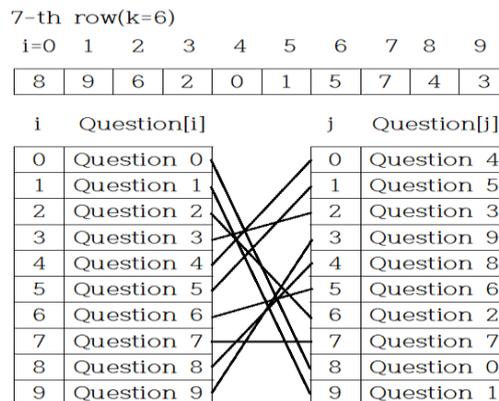


Figure 4. A Process to Relocate Question Sentences using One Row of Ready-Made Array

Figure 5 shows a ready-made array with 4x5 in which the random number in each cell is determined arbitrarily. The array consists of 5 columns: the first four columns are random numbers and the last column is the position of a correct answer which points the column with the value 0. The numbers to generate using the array randomizes less than ones using random function but the former has an advantage with high speed and is suitable for relocation of questions because applicants must solve these questions within limited time.

0	2	1	3	1
1	0	3	2	2
2	3	0	1	3
3	1	2	0	4

Figure 5. A Ready-Made Array with the Size of 4x5 for Choice Relocation

To relocate the 4 choices for each question, the server should first choose one row, such as, the fourth row, from Figure 5. Figure 6 shows the process to relocate four choices using selected row of ready-made array.

In the third phase to relocate, for example, load random number 3 of the 1-st column at the 4-th row for $i=0$ and then move Choice 0 at index 0 into the 4-th position of a new choice array. For $i=1$, load 1 of the 2-nd column and then move Choice 1 at index 1 into the 2-nd position of a new choice array, for $i=2$, load 2 of the 3-rd column and then move Choice 3 into the 3-rd position, and so on, as shown in Figure 6. The value 4 of the 5-th

column at the 4-th row is the answer for this question. This answer is moved into a correct answer array for checking applicant's answers.

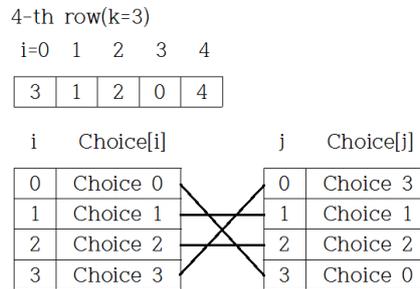


Figure 6. A Process to Relocate Choices using One Row of Ready-Made Array

Algorithm for dynamic generation of multiple choice questions is the followings:

- [Step 1] Extract components of multiple choice questions, such as, question sentence, a correct answer, and multiple distractors from database randomly up to the predefined number x and save the questions, and both an answer and multiple distractors into question array QA in one dimension and choice array CA in two dimensions, respectively.
- [Step 2] If it checks the duplicate, then repeat [step 1].
- [Step 3] Load ready-made random numbers into random array RQA for sentences, and then load other random numbers into random array RCA for choices.

Steps for relocation of question sentences with four choices

- [Step 4] The p -th row($p=0, \dots, x-1$) is chosen from ready-made array RQA using a single random number p obtained by a random function.
- [Step 5] Load random number j of the i -th column at the p -th row in RQA from $i=0$ to $i= x-1$ one by one and repeat the next [Step 6].
- [Step 6] Load the i -th question QA[i] into the j -th position of a new question array QA2[j] and then load four choices of the i -th choice array CA[i][c] ($c=0, \dots, d-1$) into the same position of a new choice array CA2[j][c].

Steps for relocation of multiple choices including an answer

- [Step 7] Repeat the next [Step 8-11] for each question from $i=0$ to $i= x-1$ one by one.
- [Step 8] The q -th row($q=0, \dots, d-1$) is chosen from ready-made array RCA using a single random number q obtained by a random function.
- [Step 9] Load random number j of the k -th column at the q -th row in RQA from $k=0$ to $k= d-1$ one by one and repeat the next [Step 10] four times.
- [Step 10] Move each choice of the choice array CA2[i][k] into a new choice array CA3[i][j].
- [Step 11] Load the last value of the choice array CA2[i][k+1] ($c=0, \dots, d-1$) into a correct answer array CAA [j].
- [Step 12] To make question screen and answer table, separate randomly relocated multiple choice question and answer into two part: one is question test which is sent to applicant and the other is answer table which is sent to scoring program to check applicant's answers.

4. Implementations and Discussions

The algorithm for dynamic generation of multiple choice using randomly relocated array is implemented on environments of JSP, apache-tomcat server, file system, and mysql [14]. The system to implement the algorithm has stored the set of question sentences, their correct answers, and their distractors about the topic on data communications separated by chapter in database. For example, 30 multiple

choice questions at the first chapter are stored in data base. If applicant clicks the particular area which he wants to take a test, the system selects 10 multiple choices questions randomly and then relocates the questions and finally relocate their choices in which each choice consists of one correct answer and 3 distractors.

Figure 7 shows the screen of the multiple choice questions to select a correct answer and shows blank to write answer at the right side. One test set has 10 questions which are divided into 5 parts with 2 questions in order to show easily for applicants. The advantage of this system offers different question test whenever applicants try to take tests or refresh. Figure 7 shows the last part including applicant's answers.

No	Multiple Choice Questions	Answer
9	What is an international, nonprofit organization formed in 1992 to provide support for the Internet standard process?	1
	(1) ISOC(Internet Society)	
	(2) IETF(Internet Engineering Task Force)	
	(3) IAB(Internet Architecture Board)	
	(4) IRTF(Internet Research Task Force)	
10	What is topology in which every device has a dedicated point-to-point link to every other device?	3
	(1) Star topology	
	(2) Multipoint connection	
	(3) Mesh topology	
	(4) Point-to-point connection	

Figure 7. A Screen to Show the Last Part of Multiple Choice Questions

The another algorithm for dynamic generation of multiple choice using random function has the same screen as shown in Figure 7 but it displays the screen slower than the previous algorithm using randomly relocated array.

Let's compare two algorithms in execution time. Before comparing the two execution times, we discuss three common phases to generate multiple choice questions consisting of questions, their correct answers and their distractors: at the first phase, predefined multiple choice questions sets are selected from the database and then, at the next one, the selected questions are relocated randomly, at the final one, multiple choices with one correct answer and 3 distractors are relocated randomly. The two algorithms use different methods to relocate in the second phase and the third one. In order to relocate in the second phase and the third one, the first algorithm proposed in this paper use the ready-made array with random numbers but the second algorithm use random function.

To compare time, each execution time in nanosecond is checked after the relocation part in the second phase is executed 100 times, because the time is too small. In this phase, 10 questions are randomly relocated 100 times using both random functions for the second algorithm and ready-made arrays for the first algorithm. Figure 8 shows the comparison of execution times in the second phase for two algorithms. In the third phase, 4 choices, such as, one correct answer and 3 distractors, are randomly relocated 100 times using both random functions for the second algorithm and ready-made arrays for the first algorithm. Figure 9 shows the comparison of execution times in the third phase for two algorithms.

No	Type	Execution Time (Random Function)	Execution Time (Relocated Array)
1	10	12691	436
2	10	16828	419
3	10	14104	413
4	10	12347	428
5	10	15635	419
6	10	15406	413
7	10	15406	416
8	10	15940	372
9	10	8439	425
10	10	18086	416
Total time		144882	4157
Ratio		34.85	1

Figure 8. Comparison Execution Times of Two Algorithms in the Second Phase

No	Type	Execution Time (Random Function)	Execution Time (Relocated Array)
1	4	7688	1486
2	4	11875	1492
3	4	9524	1474
4	4	10881	1415
5	4	12371	1480
6	4	11494	1395
7	4	11905	1463
8	4	12239	1334
9	4	6014	1372
10	4	14643	1477
Total time		108634	14398
Ratio		7.545	1

Figure 9. Comparison Execution Times of Two Algorithms in the Third Phase

As we discuss the experimental results of the two algorithms, the first algorithm proposed in this paper is faster than the second one in both the second phase and the third one. The third phase is less improved than the second one because every question of the first algorithm use random function once to select one row at the array in the third phase but only one random function in total is called to select one row at the array in the second phase.

5. Conclusions

In this paper, we proposed a system to generate randomly relocated multiple choice questions using ready-made arrays instead of random functions in order to reduce the time to make test set. Making multiple choice questions has three phases: the first phase is to select predefined multiple choice questions sets from the database and then the next one is to relocate the selected questions and the final one is to relocate multiple choices with one correct answer and 3 distractors randomly. As the results, the algorithm using ready-made arrays is faster than the algorithm using random functions in both the second phase and the third one.

Acknowledgements

This research was supported by Hallym University Research Fund, 2015(HRF-201505-009).

References

- [1] C. Fairon, "A Web-based System for Automatic Language Skill Assessment: EVALING", Proceedings of Computer Mediated Language Assessment and Evaluation in Natural Language Processing Workshop, (1999).
- [2] R. Mitkov, L. A. Ha and N. Karamanis, "Computer-aided Environment for Generating Multiple-choice Test Item", Natural Language Engineering, vol. 12, no. 2, (2005).
- [3] R. Mitkov and L. A. Ha, "Computer-aided Generation of Multiple-choice Tests", Proceedings of the First Workshop on Building Educational Applications using Natural Language Processing, (2003).
- [4] A. Papasalouros, K. Kotis and K. Kanaris, "Automatic Generation of Multiple Choice Questions from Domain Ontologies", In IADIS e-Learning 2008 conference, (2008).
- [5] T. Alsubait, B. Parsia and U. Sattler, "Generating Multiple Choice Questions from Ontologies: Lessons Learnt", The 11th OWL: Experiences and Directions Work-shop (OWLED2014), (2014).
- [6] K. A. Kim and E. M. Choi, "Automated Selection System of Estimation Questions in Web-Based Instruction", Journal of Korean Information Processing Society, vol. 9-A, no. 3, (2002).
- [7] H. J. Lee, M. S. Lee, S. M. Hong, C. H. Lee and S. H. Jung, "Web-based Automatic Question-Issuing System Using Level Estimation for Learners", Journal of Korean Information Processing Society, vol. 10-A, no. 5. (2003).
- [8] E. J. Kim, "Estimation Questions Selection Algorithm in Web-based Engineer Test Education System", Journal of Korea Society of Industrial Information Systems, vol. 9, no. 3, (2004).
- [9] E. J. Kim, "A Study on Selection Method and Mediateness Degree of Difficulty of Estimation Questions in Web-based Education System", Journal of Korean Information Processing Society, vol. 12-D, no. 3, (2005).
- [10] C. Gutl, K. Lankmayr and J. Hofler, "Enhanced Automatic Question Creator-EAQC: Concept, Development and Evaluation of an Automatic Test Item Creation Tool to Foster Modern e-Education", Electronic Journal of e-Learning, vol. 9, no. 1, (2011).
- [11] J. S. Oh, S. W. Chu, Y. S. Kim and J. Y. Lee, "A Design of Dynamic Question Generation System Using Korean WordNet", Journal of Korean Information Processing Society, vol. 4, no. 5, (2006).
- [12] A. Dryver, "The Enhancement of Teaching Materials for Applied Statistics Courses by Combining Random Number Generation and Portable Document Format Files via LATEX", Journal of Statistical Software, vol. 31, no. 3, (2009).
- [13] J. Y. Lee, "A System to Generate Dynamic Test Using Classes for replacing Numbers and Terms with Similar Data In Mathematical and Scientific Questions", Journal of Korean Information Technology, vol. 4, no. 5, (2014).
- [14] S. J. Guem, I. S. Kim and K. D. Choi, Tomcat, Mysql, and JSP, Daerim, (2007).

Author



Jae-Young Lee, received the B.E. degree from Hanyang University, Korea, in 1978 and M.E and Ph. D. degrees from Seoul National University, Korea, in 1981 and 1966 respectively. He is a professor at Department of Computer Engineering, Hallym University, Korea. His research interests include computer network and e-learning.