T. Hirashima et al. (Eds.) (2011). Proceedings of the 19th International Conference on Computers in Education. Chiang Mai, Thailand: Asia-Pacific Society for Computers in Education

Knowledge Externalization Based on Differences of Solutions for Automatic generation of Multiple-choice Question

Hisashi OGAWA[a], Hiroki KOBAYASHI[a], Noriyuki MATSUDA[b], Tsukasa HIRASHIMA[c], Hirokazu TAKI[b]
[a]*Hyogo University of Teacher Education, Japan*
[b]*Wakayama University of System Engineering, Japan*
[c]*Hiroshima University of Engineering, Japan*
ogawa@hyogo-u.ac.jp

Abstract. We study about a method to describe knowledge for generating an adequate number of multiple-choice questions. We focus on the fact that instructors can easily explain concrete multiple-choice questions, even if they cannot describe the domain knowledge completely. We propose the Environment of Abstract Rule (EAR), which enables an instructor to describe abstract knowledge by explaining the difference among some concrete multiple-choice questions, and by representing the knowledge in the form of a diagram. In this paper, we describe the concept of EAR and report the results of a preliminary experiment in which we explored whether instructors could describe knowledge easily. As a result, we suggest that the EAR has the potential for generating appropriate knowledge.

Keywords: Knowledge externalization, Multiple-choice question, e-Learning, Prolog.

1. Introduction

It is important to be aware of one's own incomplete knowledge in order to acquire knowledge. A student's voluntary understanding is an advantage, and it needs appropriate support to provide awareness or adjustment [1]. Educational material fulfills an important role in raising student awareness. Particularly in situations where students learn without a teacher, the support materials have to be designed in such a way as to call the student's attention to his/her mistakes, and to enable the student to learn better from his/her mistakes. In recent years, e-learning has been used extensively in learning situations without an instructor. In such a situation, it is difficult to work through complicated learning content when compared to a supervised learning situation. However, a learning situation without an instructor has the advantage of allowing students to learn at their own pace without needing hours of instruction; the instructor can avoid being in constant attendance on the students, such as when the students check facts they have already learned, or when they practice repeatedly.

Multiple-choice is the typical method of designing questions used in e-learning support materials. The design needs to satisfy the following two conditions in order to raise student awareness effectively. (1) Distractors that reflect the students' typical errors are required. (2) A student can be expected to acquire awareness by choosing the distractor that the instructor included. In a learning situation without an instructor, measures need to be taken to prevent any misunderstanding about the cause of error. This danger can be avoided by preparing supporting documentation that describes the cause of the error for each distractor.

However, preparing an adequate number of such multiple-choice questions that satisfy these conditions can be a burden. Therefore, an authoring system that can reduce the burden is desired. There were various types of multiple-choice questions automatic generation systemssuch asusing thesaurus [2] [3] [4], domainontology [5] [6], SWRL rules [7] or machine learning approach [8].However, there are problems in using this as support material, because it does not satisfy the two conditions described earlier.

On the other hand, there are studies that explore the possibility of generating questions that need complicated solutions by adding an advanced type of knowledge. Funaoi, Akiyama, &Hirashimaproposed a method of generating multiple-choice questions based on the knowledge of solutions in physics and dynamics[9]. Matsuda describes the Gramy, which can generate the solution and the correct answer of geometric proof problems and can explain complicated solutions [10]. However, these studies that deal with complicated solutions or knowledge have to include an in-depth discussion of the complete knowledge. This becomes an obstacle for the instructors in preparing the supporting material.

There are studies that consider renewing knowledge easily.Richards &Compton proposed the Ripple-Down Rules, which is a method for experts to easily renew knowledge [11]. They maintain that experts can judge truth and falsity and easily explain the adjustment of practical knowledge, while it is difficult for them to judge and correctly explain abstract knowledge. They propose a method that induces and reflects the shift from abstract knowledge to practical knowledge gradually, by renewing a practical question.

In this paper, we propose a method to describe knowledge for generating an adequate number of multiple-choice questions. We focus on the fact that instructors can easily explain concrete multiple-choice questions, even if they cannot describe the domain knowledge completely as with Ripple-Down Rules. We propose the Environment of Abstract Rule (EAR), which enables an instructor to describe abstract knowledge by explaining the difference among some concrete multiple-choice questions, and by representing the knowledge in the form of a diagram. In this study, we use Simulation of Erroneous Solution (SES) to generate multiple-choice questions. SES generates the knowledge described by Prolog, the key, and the solution, using the question template. Then, SES generates distractors and the erroneous solution that the instructor assumed, using the perturbation operator, which rewords a part of the knowledge. The EAR that we propose in this paper is a method of generating the Horn clause in Prolog which is the knowledge of SES.

This paper describes the concept of EAR. Section 2 discusses a precondition of the EAR, and Section 3 describes its structure. Section 4 reports the results of a preliminary experiment in which we explored whether instructors could describe knowledge easily.

2. Precondition

The precondition assumed in this paper is as follows. First, the EAR targets the multiple-choice type supporting material, which we henceforward refer to as “multiple-choice.” It is composed of the following: a given, the beginning of the question that explains the information necessary to solve the problem; a query, the end of the question that presents the item as a problem to be solved; the choices, consisting of one key and some distractors; the guide, which explains the correct solution to get a key and the location of the error in the solution of each distractor. Second, the EAR uses Prolog to generate the multiple-choices. Therefore, the EAR targets those multiple-choices for which procedural knowledge can derive solutions. Third, if the multiple-choices are few in number, there is the possibility that someone would memorize the set of questions and the key. The EAR is effective in preparing a sufficient number of multiple-choices. Lastly, the

EAR targets instructors, who can explain the solution for multiple-choices, and prepares distractors that include typical errors made by the students. However, the EAR requires no understanding of knowledge engineering.

3. Structure of EAR

3.1 Multiple-choicegeneration module

First, we implemented a multiple-choice generation module based on SES before developing the EAR. SES is a method of generating a solution from wrong knowledge to wrong answer, by a perturbation from problem-solving knowledge to knowledge of wrong answer derivation. The multiple-choice generation module requiresa given condition, a query condition, a knowledge base, and a perturbation operator as shown in Figure 1. The perturbation is projected on a student's typical error, and the perturbation operator expresses the difference between the erroneous solution and the correct solution.

First, the multiple-choice generation module generates a key and the correct solution by deriving the knowledge base using the given condition and the query condition. It ispossible to generate a guide to the correct solutions by replacing a variable in the templatewith the correct solution. Next, the multiple-choice generation module adds the perturbation from correct solution to erroneous solution, and generates distractors using the erroneous solution. Finally, it is possible to generate a guide to the erroneous solutions, which explains the difference between the correct solution and the student's erroneous solution, by replacing a variable in the template with the erroneous solution.

We confirmed that forty questions and a hundred sixty explanation text about the key and the distractors are automatically generated from seventeen prolog clauses as knowledge into SES.In the following section, we explain how to create the given condition, the query condition, the knowledge base, and the perturbation operator (Figure 2).

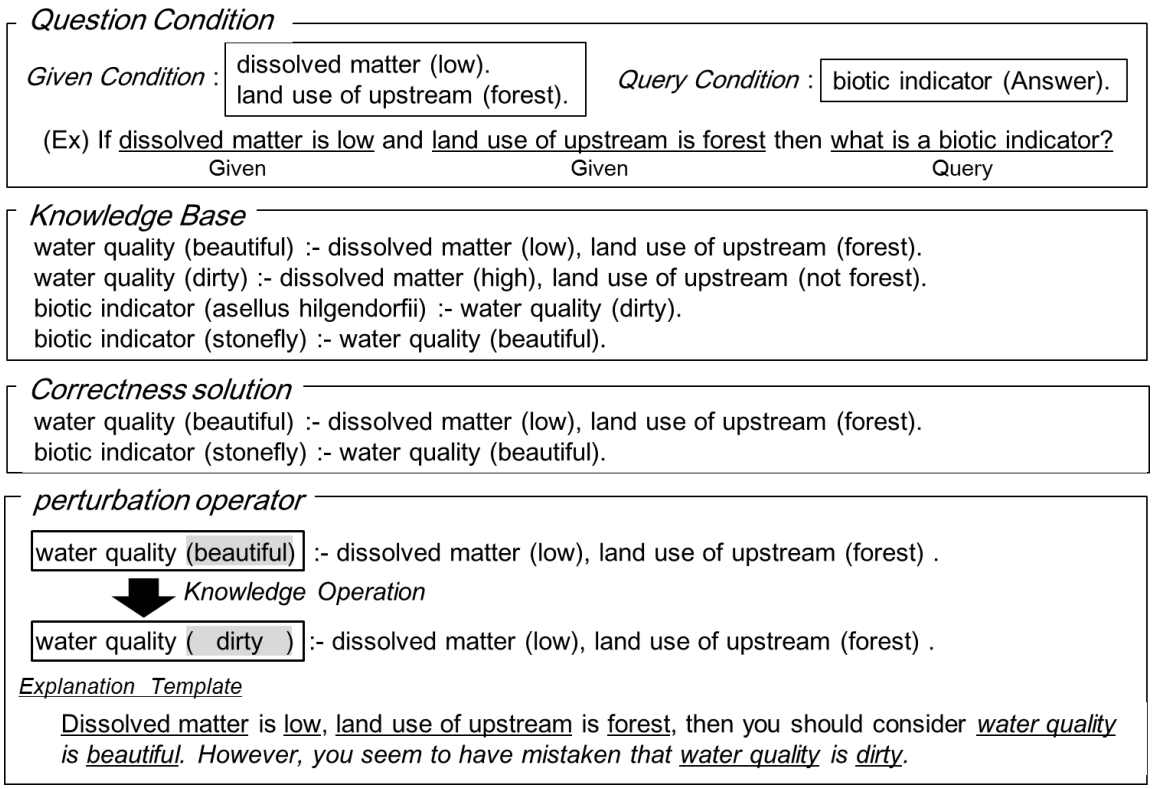


Fig. 1Knowledge List generated by EAR

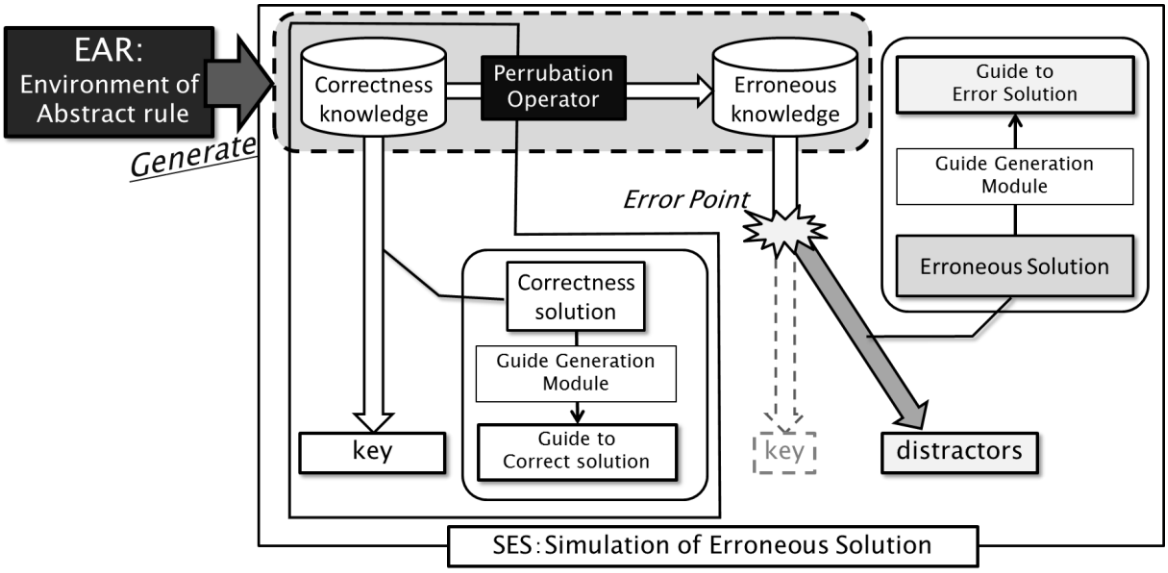


Fig. 2 Relation between SES and EAR

3.2 Overview of the EAR

The knowledge used in the multiple-choice generation module is composed of an attribute, an attribute-value, an attribute level's causal connection, and an attribute-value level's causal connection, as shown in Figure 3. Instructors have implicit knowledge, and therefore it is easy for them to shape up and explain the knowledge. However, it is difficult for them to abstract the knowledge. Therefore, the EAR helps the instructors to abstract knowledge. The procedure for abstracting knowledge is described below:

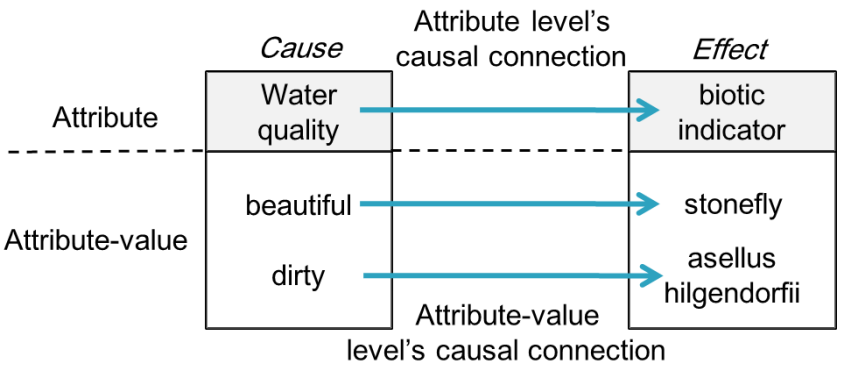


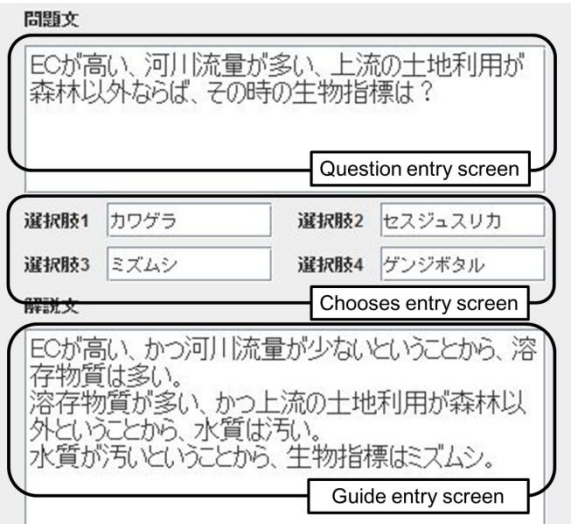
Fig. 3 A knowledgecomponent

(1) *Creating a base-question*: The instructor creates some multiple-choices that include a question, the choices (the key and some distractors), and a guide, which explains each choice. We call it the base-question.
(2) *Generating a query condition*: The system generates the query condition by asking the instructor about the attribute of the choices in the base-question.
(3) *Generating a given condition*: The instructor alters the question and explains it. The parts before and after the change are attribute-values that belong to the same attribute. For example, if the instructor alters the word "stonefly" to "*Asellus hilgendorfii*" these would be attribute-values that belong to the same attribute, "biotic indicator." Thus, this enables the generation of the given condition, because attributes are generated.
(4) *Generating an attribute*: The system generates the attribute used by the correct solution, by altering the guide of the base-question, and explaining it as described in (3).

(5) *Generating an attribute level's causal connection*: The system generates the attribute level's causal connection by explaining the relationship between two attributes that do not connect to the query condition. When the instructor explains it, he/she uses the attribute generated in (2), (3), and (4). If there is no attribute on the attribute list generated, it implies a lack of attributes, which are essential for solving the question. Therefore, the system directs the instructor to extract the lacking attribute, by means of (2). This helps to prevent a shortage or an error of attribute. Then, the instructor describes the direction of the causal connection. When the instructor explains the causal connection, there is a significant difference between "the biotic indicator depends on the quality of water" and "the quality of water depends on the biotic indicator." Therefore, the direction of the causal connection is very important.
(6) *Error check of knowledge generated*: The system shows the instructor some questions generated by the multiple-choice generation module. If the knowledge generated is correct, the question that shows the correct causal connection is displayed. On the other hand, if the knowledge generated is incorrect, the system generates an error question. It is possible to notify the instructor if the knowledge is defective, and to edit the causal connection again. If the knowledge is defective, the instructor repeats steps (2) to (5).
(7) *Inputting the attribute-value*: The correct knowledge is constructed at this stage, and the instructor adds the attribute-value to the attribute that the system generated.
(8) *Generating the attribute-value level's causal connection*: The system generates the attribute-value level's causal connection by making the instructor explain the relation among the attribute-values.
(9) *Generating a perturbation operator*: The system needs error knowledge to generate the distractors and the guide to the erroneous solutions; therefore, it is necessary to add the perturbation at the specific point of correct knowledge. Therefore, the system generates the perturbation operator by changing either of the attribute-values that has a causal connection to the error attribute-value, which reflects a student's typical error.

3.3 System component

We implemented the EAR to the knowledge revelation module in a multiple-choice generation system. In this section, we describe the knowledge revelation module.
(a) *Base-question editor*: The instructor inputs a base-question (question, choices, and guide) from the base-question creation interface, as shown in Figure 4. It is possible for the instructor to describe a base-question freely.
(b) *Attribute Editor*: First, the instructor alters the attribute-value and explains to what attribute the attribute-value belongsusing the attribute editor. Figure 5 shows the screen shot where the instructor is asked to what attribute "stonefly" and "*Asellus hilgendorfii*" belong. If instructor answers "biotic indicator," the knowledge revelation module generates the attribute "biotic indicator (stonefly)" and "biotic indicator (*Asellus hilgendorfii*)."Second, the attribute editor shows the attributes that do not connect to the query condition. Next, the attribute editor encourages the instructor to complete the relation graph; the relation graph has to satisfy the conditions that a root node is the query condition, and that all attributes connect to the query condition. Then, the instructordescribes in the textbox the relation between the attributes that do not connect to the query condition, as shown in Figure 6. When an attribute appears in the description, it needs to pick the attribute from the list. If there is no attribute that the instructor wants to describe in the list, it lacks an attribute, and the knowledge revelation module directs the instructor to extract the lacking attribute. Finally, the instructor chooses from the choices prepared by the knowledge revelation module to decide the direction of the causal connection.



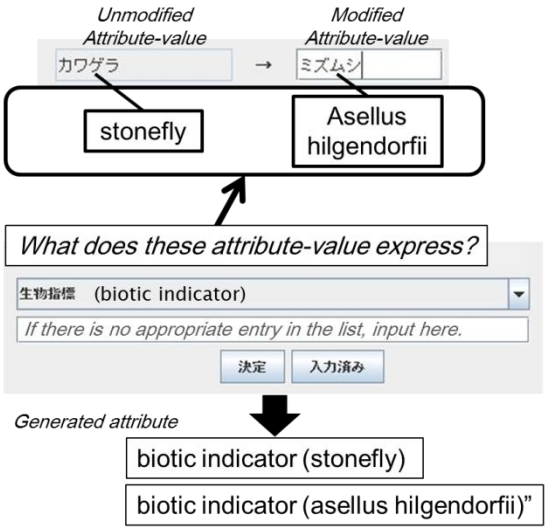


Fig. 4 Base-question editorFig. 5 Attribute editor

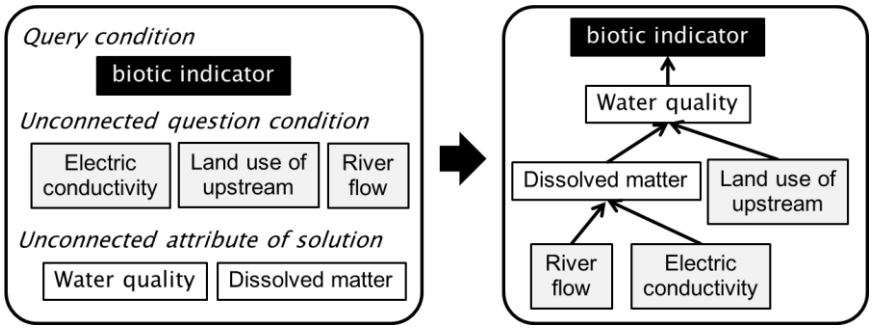


Fig. 6 An attribute list and a relation graph

(c) *Error-check interface*: The multiple-choice generation module generates some multiple-choices from the attributes and the attribute level’s causal connection, and presents them to the instructor. The instructor can learn about the defective knowledge by referring to the error question or the guide generatedby the multiple-choice generation module. An error-check interface can prompt the instructor to edit the attributes and causal connections.
(d) *Attribute-value editor*: First, the instructor edits the attribute-value against each node (attribute) of the graph. Next, the instructor describes the attribute-value level’s causal connection with the editor, as shown in Figure 7. Then, the instructor chooses the adding point of the perturbation with the interface, and describes the erroneous causal connection in the same manner.

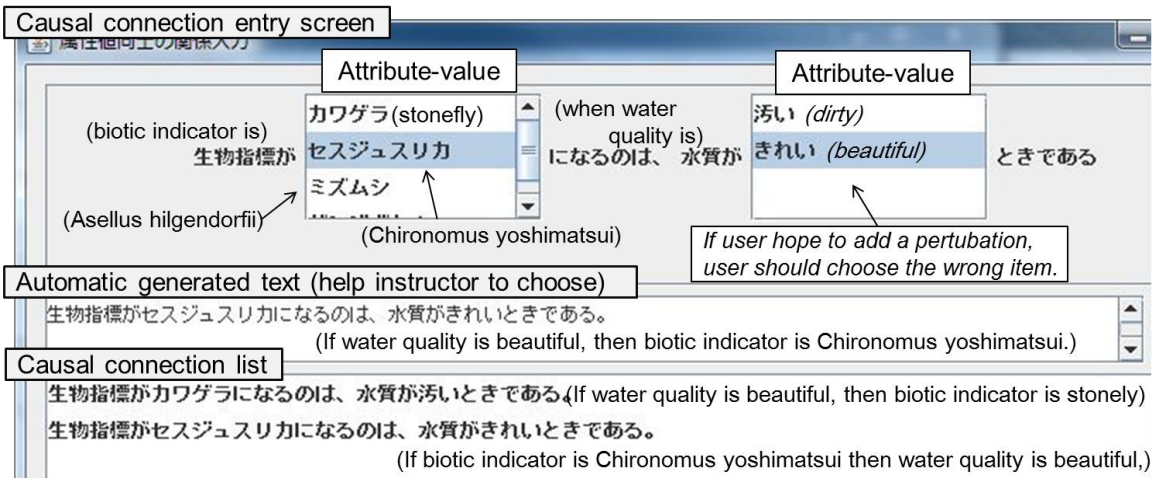


Fig. 7 Attribute-value editor

4. Preliminary experiment

In this section, we explore whether instructors can express knowledge, by comparing the control group whose subjects described the solution knowledge using Prolog with a text editor, and the experimental group whose subjects used the knowledge revelation module that implemented the EAR. To compare these two groups, we calculated the conformance ratio (the ratio of correct knowledge in the solution knowledge that the subjects described) and the recall ratio (the ratio of the solution knowledge that the subjects described in the correct knowledge). The seven subjects who participated in our experiment were university students without any understanding of knowledge engineering.

4.1 Procedure

First, we showed material consisting of English (verb conjugations) and biology (stream dwellers), and got the content across to the subjects. Second, we conducted a test on the material to verify each subject's ability to create appropriate multiple-choices. If a subject could not pass the test, the subject had to go over the contents of the material again. Third, we divided the subjects into two groups, the control group and the experimental group, and asked the subjects to express their knowledge of either English or biology. There were three subjects in the control group (two English and one biology), and four in the experimental group (two English and two biology). We explained how to describe knowledge using Prolog, and answered the subjects' queries, which had little impact on the experimental result. Lastly, the subjects alternated between the control group and the experimental group to express their knowledge of either English or biology for counterbalancing; this was not done in (3).

4.2 Result

Table 1 shows the conformance ratio and the recall ratio of the causal connections of the attribute levels and the attribute-value levels. We conducted two sample t-tests for the difference of the average value, and confirmed the significantly different value between the groups at 5% significance. Thus, we demonstrated that the experimental group had a higher score than the control group, both for the conference ratio and the recall ratio.

Table. 1 Result of the preliminary experiment

Examinee	Subject	experimental group		Subject	control group	
		conformance ratio	recall ratio		conformance ratio	recall ratio
A	English	3/3 (100%)	3/3 (100%)	Biology	0/7 (0%)	0/4 (0%)
B	English	2/2 (100%)	2/3 (67%)	Biology	1/4 (25%)	1/4 (25%)
C	English	3/3 (100%)	3/3 (100%)	Biology	2/5 (40%)	2/4 (50%)
D	English	3/3 (100%)	3/3 (100%)	Biology	1/4 (25%)	1/4 (25%)
E	Biology	4/4 (100%)	4/4 (100%)	English	2/3 (67%)	2/3 (67%)
F	Biology	4/4 (100%)	4/4 (100%)	English	2/3 (67%)	2/3 (67%)
G	Biology	4/4 (100%)	4/4 (100%)	English	3/5 (60%)	3/3 (100%)

4.3 Discussion

We determined that all the conformance ratios of the causal connection of both the attribute levels and the attribute-value levels in the experimental group was 100%, as shown in Table 1. The conformance ratio needs to be 100% for developing the expression for Prolog. However, all subjects in the control group didn't have a 100% conformance ratio, whereas all subjects in the experimental group had a 100% conformance ratio. Meanwhile, the recall ratios of the experimental group were not as high as the conformance ratios. However, thiswas not a serious problem, even though the recall ratio affects the number of the multiple-choices generated. Three of the seven subjects did not have a 100% ratio; they could express the causal connection of the attribute-value levels approximately 80% of the time. This was only a preliminary experiment, and therefore it is still too early to conclude that the EAR is helpful. However, our study suggests that the EAR has the potential for generating appropriate knowledge.

5. Summary

In this paper, we describe the concept of EAR and report the results of a preliminary experiment in which we explored whether instructors could describe knowledge easily. While this was only a preliminary experiment, our study suggests that the EAR has the potential for generating appropriate knowledge to generate the multiple-choice by the SES. Future works, we will perform an experiment for a lot of subjects and different domain.

References

[1] Perkinson, J, H. (1984) Learning from our mistakes:reinterpretation of twentieth century educational theory. Contribution to the study of education, greenwood press, Pennsylvania.

[2] Sumita, E., Sugaya, F., & Yamamoto, S. (2005) Measuring non-native speakers' proficiency of English by using a test with automatically-generated fill-in-the-blank questions.*Proc. of ACL '05 Workshop: Building Educational Applications Using Natural Language Processing*, 61-68.

[3] Brown, J. C., Frishkoff, G. A., & Eskenazi, M. (2005) Automatic Question Generation for Vocabulary Assessment.*Proc. of HLT/EMNLP '05*, 819-826.

[4] Lin, Y. C., Sung, L. C., & Chen, M. C. (2007) An Automatic Multiple-Choice Question Generation Scheme for English Adjective Understanding.*ICCE 2007 Workshop Proc. of Modeling, Management and Generation of Problems / Questions in eLearning*, 137-142.

[5] Mitkov, R., Ha, L., & Karamanis, N. (2006) A computeraided environment for generating multiple-choice test items.*Natural Language Engineering*, 12(2):177-194.

[6] Holohan, E., Melia, M., McMullen, D., & Pahl, C. (2005) Adaptive courseware generation based on semantic web technology.*Proc. of SW-EL 2005*, 29-36.

[7] Zoumpatianos, K., Papasalouros, A., & Kotis, K. (2011) Automated Transformation of SWRL Rules into Multiple-Choice Questions.*Proc. of FLAIRS-24*, 570-575

[8] Funaoi, H., Akiyama, M., & Hirashima, T. (2006) Automatic Creation of Mis-choices and Comments for Multiple Choice Question Based on Problem Solving Model.*ICCE 2006 Workshop Proc. of Problem-Authoring, -Generation and -Posing in a Computer-Based Learning Environment*, 49-54.

[9] Gotoh, T.,Kojiri, T.,Watanabe, T., Iwata, T., and Yamada, T. (2010) Automatic Generation System of Multiple-Choice Cloze Questions and its Evaluation.*Knowledge Management & E-Learning: An International Journal*, 2(3), 210-224.

[10] Matsuda, N., &Vanlehn, K. (2004) GRAMY: A Geometry Theorem Prover Capableof Construction.*Journal of Automated Reasoning*, 32, 3-33.

[11] Richards, D., &Compton, P. (1998) Taking up the situated cognition challenge with ripple down rules.*International Journal of Human-Computer Studies*, 49(6), 895-926.