Organization of Solution Knowledge Graph from Collaborative Learning Records

Yuki Watanabe, Tomoko Kojiri, and Toyohide Watanabe

Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan
{ywatanabe,kojiri,watanabe}@watanabe.ss.is.nagoya-u.ac.jp

Abstract. In collaborative learning, participants generate their own answers by exchanging their opinions through a discussion. Since the discussion in a collaborative learning includes knowledge for solving an exercise, the collaborative learning record is useful for other learners who tackle the same exercise. We propose a method for organizing solution knowledge in collaborative learning records as a solution knowledge graph. In this method, utterance collections of the same answering method are extracted and structured from a viewpoint of their effectiveness based on annotations attached by participants. In addition, the structure of the solution knowledge graph is refined by learning records of self-learners who use it as knowledge for solving exercises.

Keywords: solution knowledge graph, learning record, collaborative learning, self-learning, knowledge extraction, effectiveness of utterances.

1 Introduction

Recently, to support a collaborative learning under a distributed environment is one of the hottest subjects[1][2]. In the collaborative learning, participants propose their own ideas, ask questions, and reply to the questions in order to solve a common exercise. Since such utterances contain hints to solve the exercise, participants acquire knowledge to derive answers from the utterances which they cannot think of by themselves. Such knowledge is useful for other learners who did not participate into the collaborative learning, but tackle the same exercise.

Participants in different learning groups may derive answers with different answering methods. Even if they follow the same answering methods, they discuss differently with different knowledge. Therefore, it is useful for other learners to observe plural discussion records of the same exercise. However, if there are many discussion records for the same exercise, it is difficult for learners to find utterances that are appropriate for their learning situations. Our objective is to extract knowledge for deriving the same exercise automatically from the discussion records in the several collaborative learnings in order for other learners to utilize the discussion records as knowledge to solve the same exercise.

Several researches have been investigated for utilizing collaborative learning records. Kayama, et al.[3] developed a system which supports learners to review

J.D. Velásquez et al. (Eds.): KES 2009, Part II, LNAI 5712, pp. 564–571, 2009.

the contents of the collaborative learning according to tags that are automatically attached to specific actions of participants by the system. Goodman, et al.[4] constructed SAILE which organizes collaborative learning records based on status in common workspace. This system provides asynchronous collaborative learning environment where learners can review all actions, and replay to actions of the scenes that are defined by chat events. In addition, learners can increase the branch of a learning process by attaching comments to the discussion record. These researches allow learners to review collaborative learning record from specific scene. However, it is difficult to find scenes that may help them to derive the answer.

In order to extract parts which can be used for deriving answers from collaborative learning, it is necessary to detect topics. Adams, et al.[5] developed a system which extracts utterances of the same topic. This system calculates similarities between utterances according to feature vectors of utterances and detects topics. The feature vectors are estimated with frequency of terms in all utterances based on *tf-idf*. Okazaki, et al.[6] proposed a method for extracting and organizing topics from text documents. In this method, similarities between sentences are calculated with vocabularies, and sentences are organized according to their similarities. However, these researches only arrange topics in terms of similarities. Utterances in the collaborative learning should be organized from a viewpoint of their effectivenesses in solving exercise.

Currently, we focus on *self-learners* who did not participate into the collaborative learning and study individually using the collaborative learning records as hints for solving the exercise. Self-learners refer to the collaborative learning records when they cannot derive answers by themselves. Thus, utterance collections that are effective for solving the exercise need to be distinguished from other utterances of the same topic. Kakehi, et al.[7] developed a system which extracts useful utterance collections for deriving answers from a discussion record and organizes them along the time sequence. The system detects effective utterances based on annotations that are attached by participants to useful utterances for deriving their answers during the collaborative learning. This system could extract utterances that may be useful for solving the exercise, but could not show effectiveness of extracted utterances. Moreover, it is able to handle only a discussion record of a single collaborative learning.

In this research, effective utterances in the plural discussion records are extracted and structured as one solution knowledge graph according to the effectiveness for solving an exercise. Effective utterances may be referred by many participants/self-learners. In our approach, useful utterances are detected based on the participants' intentions for solving the exercise. As same as Kakehi's method[7], participants' intentions are grasped by annotations that are attached to utterances which are referred in solving the exercise. Effectiveness of utterances is inferred by the number of participants who attached annotations. Thus, extracted utterances are arranged in the solution knowledge graph according to the number of annotations.

The number of participants in the group may be small, so utterances that are annotated by participants are not always effective for other self-learners. Moreover, meanings of utterances in the discussion may not be the same for other self-learners, since they do not know the context of utterances. Therefore, the structure of the solution knowledge graph is refined based on the self-learners' learning records. By considering intentions of self-learners, for the solution knowledge graph, it is able to organize knowledge discussed in the collaborative learnings according to the importance in solving the exercise.

2 Approach

2.1 Collaborative Learning Environment

We focus on a collaborative learning of programming exercises. The solution of a programming exercise is composed of several answering steps which correspond to sub-exercises. For example, in an exercise of "construct a program which retrieves an input string from a file", there are there answering steps such as "to obtain input string", "to operate a file", and "to retrieve string from the file". Most sub-exercises are independent to others. Several answering methods can be applied to solve the sub-exercises. Moreover, each answering step holds keywords that can identify the step.

We assume the group of participants who have the similar understanding levels and try to solve the same exercise. Participants compose their own programs while discussing their ideas with others through a chat tool. When generating utterances, participants need to indicate target utterances. Also, they are required to attach annotations to utterances that are used for deriving their answers.

2.2 Framework for Organizing Knowledge in Discussion Records

When solving an exercise, self-learners cope with answering steps individually. Therefore, utterances of the same topic need to be extracted as an utterance collection and be corresponded to the discussed answering steps. In addition, in order for self-learners to find the effective hints quickly, utterance collections should be evaluated and ranked according to their effectiveness.

In a discussion record, several utterances may be generated for one topic. It is necessary to detect successive utterances of the same topic as an utterance collection and specify their answering steps. At this time, utterance collections are extracted from all the collaborative learning records for the same exercise. If more than one answering methods exist, differences among the answering methods should be specified. In addition, the effectiveness of utterance collections for deriving an answer may help learners to find useful hints for each answering method.

In order to organize solution knowledge from collaborative learning records, we design the learning environment which consists of mechanisms for extracting and structurizing the hints in the discussion records as solution knowledge graph

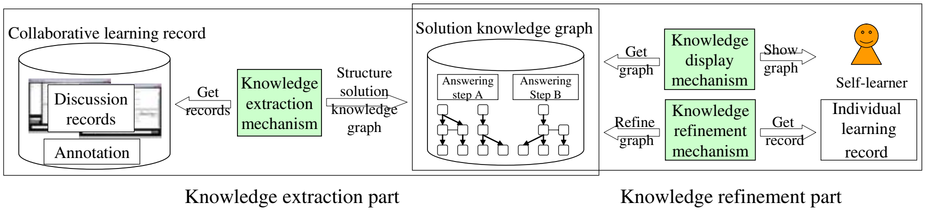

Fig. 1. Outline of our system

automatically, and for refining it through its use in individual learnings. Figure 1 illustrates the outline of our system. In the knowledge extraction mechanism, utterance collections are extracted based on the annotations attached by the participants and are classified into answering steps according to their contents. Since annotations attached by the same participants may indicate utterances of the same answering method, utterance collections are classified into answering methods based on the participants who attached the annotations and are structured as a solution knowledge graph. In knowledge display mechanism, solution knowledge graph is shown to self-leaners. In knowledge refinement mechanism, the structure of the solution knowledge graph is changed based on the learning records of self-learners who use the solution knowledge graph as hints for solving the same exercise. By reflecting intention for not only participants but also self-learners, the solution knowledge graph is able to represent utterance collections that are effective for many learners.

3 Solution Knowledge Graph

3.1 Definition of Solution Knowledge Graph

In the solution knowledge graph, useful utterance collections are arranged for answering methods in each answering step. Figure 2 shows the conceptual imagination of the solution knowledge graph. The solution knowledge graph is composed of nodes and links. A node shows a useful utterance collection and contains information on participants/self-learners who used this utterance collection to derive the answer. Nodes at higher positions in the solution knowledge graph correspond to more effective utterance collections, and nodes at lower positions are used by only a few participants/self-learners. Links connect utterance collections with the same answering method. Undirected links are attached to utterance collections whose effectiveness are the same, and directed links indicate that target nodes are supplementary to source nodes.

3.2 Construction of Solution Knowledge Graph

Nodes in the solution knowledge graph express utterance collections. Nodes consist of successive utterances of the same topic that are used for deriving answers. Utterance collections of the same topic is extracted by target utterances of the

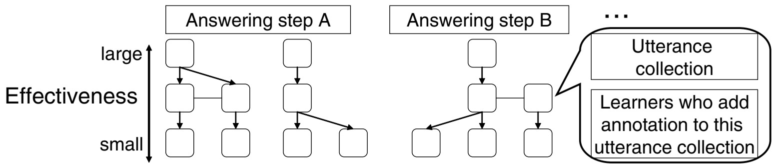

Fig. 2. Solution knowledge graph

utterances that are indicated by participants. Utterances may be derived by their target utterances, and utterances and their target utterances are regarded to express the same topic. So, utterances that are originally derived by the same utterances are gathered and compose utterance collections.

Of all utterance collections, useful ones are used by participants to derive the answer. Since annotations are attached to utterances used for deriving the answer, utterance collections including the utterances to which annotations were attached are extracted as useful utterance collections. The extracted utterance collections form nodes of solution knowledge graph. The number of participants who attached annotations is defined as the degree of effectiveness of the nodes.

The useful utterance collections include knowledge which is necessary for deriving the answer. Therefore, they belong to one of the answering steps. By comparing keywords in each answering step with words included in the utterance collections, nodes are classified into the corresponding answering steps. A participant solves the exercise along one answering method for each answering step. So, if annotations are attached to two utterance collections in one answering step by the same participants, they may belong to the same answering method. *Corresponding rate* is defined as the possible rate that two utterance collections belong to the same answering method.

Corresponding rate between nodes i and j is calculated by Equation 1. N_i is a set of the participants who attached annotations to the utterance collection in node i. When a corresponding rate is more than a threshold, a link is added between the two nodes. When the degrees of effectiveness of nodes i and j are equal, an undirected link is generated between them. If their degrees of effectiveness differ, a directed link is generated from the node with the large effectiveness to the small node.

$$corresponding\ rate = \frac{||N_i \cap N_j||}{||N_i||} \qquad ||N_i|| \geq ||N_j|| \tag{1}$$

3.3 Refinement of Solution Knowledge Graph

The structure of the solution knowledge graph, such as positions of nodes and links, is reconstructed based on learning records of the self-learners. Nodes in the solution knowledge graph that help self-learners to derive answers are counted as useful utterance collections. Therefore, after a self-learner finishes a individual learning, the degree of effectiveness is modified and positions of nodes in the solution knowledge graph are rearranged. Corresponding rates between nodes are also re-calculated based on effective degree that is the number of self-learners who refer to the node to the number of annotations. Equation 2 is the calculation method of corresponding rate in refining the solution knowledge graph. R_i is a set of the self-learners who used the node i. According to the corresponding rate, links of the solution knowledge graph are generated, changed, or deleted.

$$corresponding\ rate = \frac{||(N_i \cup R_i) \cap (N_j \cup R_j)||}{||(N_i \cup R_i)||} \quad ||(N_i \cup R_i)|| \geq ||(N_j \cup R_j)|| \quad (2)$$

4 Prototype System

e construct a prototype system for an individual learning which displays the solution knowledge graph as a knowledge source. The system constructs a solution knowledge graph from discussion records and shows it to a self-learner. Figure 4 shows the interface for the individual learning. Self-learners need to input an exercise number from the combo box and start learning by pushing the start button. When the start button is pushed, an exercise sentence is emerged in the exercise display area, and the solution knowledge graph is drawn in the solution knowledge graph display area. The solution knowledge graph display area is composed of more than one answering step tabs which correspond to each answering step. When an answering step tab is selected, the solution knowledge graph in the corresponding answering step is shown in the solution knowledge graph display area. The rectangles in the solution knowledge graph display area express nodes, and the words in a rectangle represent words that appear more than twice in the corresponding utterance collection.

When the node is clicked, utterances in the selected node are displayed in the discussion display area. When self-learners refer to the utterances that are currently displayed in the discussion display area, the utilization button is pushed.

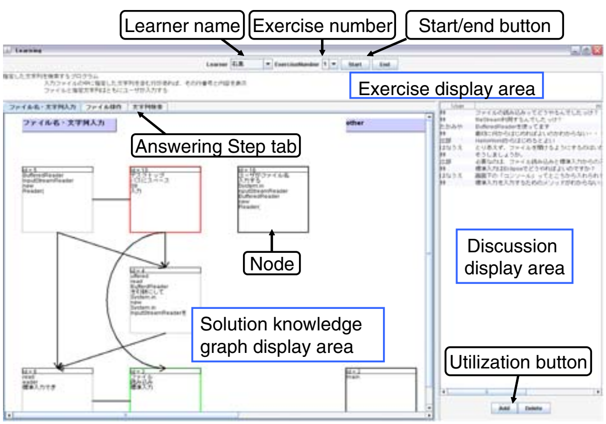

Fig. 3. Interface for individual learning

Then, the system recognizes that the node is used during the individual learning. By pushing the end button, the individual learning is finished and the structure of solution knowledge graph is changed.

5 Experiment

We evaluate the refinement method of a solution knowledge graph. The solution knowledge graph was constructed with discussion records in collaborative learnings of two groups (five and four students in our laboratory). The exercise which participants tackled in the collaborative learnings was "construct a program which retrieves an input string from a file". It is composed of three answering steps, such as "to obtain input string", "to operate a file", and "to retrieve string from the file". The solution knowledge graph was composed of 22 nodes and 10 links. The threshold for generating links was set to 0.5.

Four other students in our laboratory were asked to tackle the same program with the prototype system one by one. They were asked to push the utilization button when referring the node. After each student finished learning, the solution knowledge graph was updated.

Table 1 shows the total number of links changed by four learners. The appropriateness of changed links are evaluated by checking their contents. For example, if links were generated between nodes that indicate the same answering methods, they were regarded as correct links. However, if nodes did not belong to the same answering method, links were attached incorrectly. Eight links were correctly changed, and five links were operated incorrectly. Currently, the influence of one push of the utilization button was large, since the number of students in this experiment was small. Therefore, it is necessary to change the threshold according to the number of self-learners.

On the other hand, Table 2 shows the number of pushing utilization button for nodes in each position in the solution knowledge graph. Layer 1 holds to the most effective nodes and effectiveness of nodes gets smaller as the layer becomes lower. The nodes that were useful for deriving answer were successfully arranged at a higher layer in the solution knowledge graph.

Table 1. Total number of changing links by four self-learners

	correct	incorrect
No. of generated links	2	4
No. of deleted links	6	1
Total No. of changed links	8	5

Table 2. The number of pushing utilization button for nodes in each layer in solution knowledge graph

Position in solution knowledge graph	1st	2nd	3rd	4th	5th	6th
No. of pushing utilization button	10	5	3	4	1	0

6 Conclusion

In this paper, the method for structuring a solution knowledge graph was proposed. The solution knowledge graph was organized by useful utterance collections for deriving answer that were extracted from discussion records. Moreover, it was refined based on records of self-learners who used it. From the experimental result, we confirmed that more useful utterance collections were arranged in higher layer. However, it turned out that links were not generated correctly, especially when the number of participants/self-learners were small. We should reconsider the method for generating links between the nodes of the same answering method by grasping contents of the nodes.

Currently, our system regards utterances collections that many learners referred were more effective. However, all learners do not necessarily need the same knowledge. Effective knowledge maybe different if learners' understanding levels are different. Therefore, a mechanism of recommending utterance collections according to learners' understanding levels should be added.

In addition, our system only allows self-learners to browse the discussion records. In order to organize more effective solution knowledge, the mechanism which modifies discussion records, such as nodes in the solution knowledge graph, need to be developed. For example, a function to comment or correct the utterance collection in nodes is considered.

References

1. Kam, M., Wang, J., Iles, A., Tse, E., Chiu, J., Glaser, D., Tarshish, O., Canny, J.: Livenotes: A System for Cooperative and Augmented Note-Taking in Lectures. In: Proc. of CHI 2005, pp. 531–540 (2005)
2. Dimitracopoulou, A.: Designing Collaborative Learning Systems: Current Trends and Future Research Agend. In: Proc. of CSC 2005, pp. 115–124 (2005)
3. Kayama, M., Okamoto, T.: Knowledge Management Framework for Collaborative Learning Support. In: van Elst, L., Dignum, V., Abecker, A. (eds.) AMKM 2003 Part II. LNCS, vol. 2926, pp. 107–117. Springer, Heidelberg (2004)
4. Goodman, B., Geier, M., Haverty, L., Linton, F., Mcready, R.: A Framework for Asynchronous Collaborative Learning and Problem Solving. In: Proc. of AIED 2001, pp. 188–199 (2001)
5. Adams, P.H., Martell, C.H.: Topic Detection and Extraction in Chat. In: Proc. of ICSC 2008, pp. 581–588 (2008)
6. Okazaki, N., Matsuo, Y., Matsumura, N., Ishizuka, M.: Sentence Extraction by Spreading Activation through Sentence Similarity. IEICE Trans. on Information and Systems E86D(9), 1686–1694 (2003)
7. Kakehi, M., Kojiri, T., Watanabe, T., Yamada, T., Iwata, T.: Organization of Discussion Knowledge Graph from Collaborative Learning Record. In: Diekert, V., Volkov, M.V., Voronkov, A. (eds.) CSR 2007. LNCS (LNAI), vol. 4649, pp. 600–607. Springer, Heidelberg (2007)