Information Engineering Express
International Institute of Applied Informatics
2018, Vol.4, No.1, P.77 - 86

Pattern Analysis of Learners' Artifact Making Processes and Errors in Conceptual Modeling Exercises

Takafumi Tanaka* Kousuke Mouri* Keiichi Kaneko*

Abstract

Conceptual modeling (CM) is an important technique in database design. Education of the technique is a challenge of computer science departments in higher education institutions. The aim of this study is to find patterns of learners' errors in their artifacts and their artifact making processes in conceptual modeling exercises. This study collected the data of learners' artifacts and the process of creating them in an exercise of CM using a system, KIfU 3.0, and conducted an association analysis with this data. As a result, this study found the association rules regarding learners' errors in the artifacts and/or their characteristics of the artifact making processes. For example, there are some errors that have a tendency to co-occur in the learners' artifacts, and learners who refine their artifacts frequently tend to detect attributes of concepts successfully even though they are hard to detect from a requirement specification for the exercise. This fact implies the effectiveness of our approach and a suggestion to improve teaching of the CM.

Keywords: artifact making process, association analysis, conceptual modeling, software engineering education.

1 Introduction

Conceptual modeling (CM) [1, 2] is an important activity in database design [5, 6]. In CM, we detect concepts, which are things and events that are handled in software [2, 3, 4], and model them with their relationships. As a pedagogical issue in CM, it is often argued to improve learners' making processes [7, 8, 9]. It is necessary for teachers to find points in the learners' making processes that need intervention. So far, little attention has been paid to analyzing their artifact making processes in order to clear up learners' misunderstandings and work on their bad making processes. Analyzing the learners' making processes in CM exercises leads to enhancement of their skills. Therefore, this study analyzes them to focus on the following three points:

(1) Patterns of learners' errors in CM exercises;

(2) Characteristics of learners' making processes;

(3) Relations between (1) and (2).

* Tokyo University of Agriculture and Technology, Tokyo, Japan

The rest of this paper is structured as follows. Section 2 presents related works. Section 3 describes our system called KIfU 3.0. Section 4 describes the method of data collection and analysis. Section 5 discusses the results of analysis. Section 6 describes the conclusions and future research plans.

2 Literature Review

2.1 Conceptual Modeling Education

Suraweera and Mitrovic developed a learning environment of Entity-Relationship (ER) modeling called KERMIT [10], which is similar to CM [11]. It monitors learners' artifacts in ER modeling and provides feedbacks for each learner through an agent. The feedback includes suggestions of actions the learners should take next and indications of errors in their artifacts. However, they did not consider data analysis to find patterns of learners' errors.

Kung et al. conducted an empirical comparison of learners' performance between the top-down and bottom-up approaches in a conceptual data modeling exercise [12]. They conducted an experimental exercise where the participants made ER models using the specified methods. Kung et al. analyzed the result with focusing on the differences of the participants' error rates by the methods. However, they did not consider data collection of learners' artifact making processes.

Tanaka et al. developed a system, KIfU 2.0, to support learners to express their thinking in CM exercises [13]. KIfU 2.0 is a UML Class diagram editor that can collect log data of learners' artifact making processes. KIfU 2.0 provides an environment for learners to express their thinking while reflecting on the processes. They succeeded in collecting the data of learners' thinking associated with the artifact making processes. However, they did not focus on the pattern analysis regarding the learners' errors or their artifact making processes.

An originality of this study is in finding patterns of learners' errors and making processes from actual data.

2.2 Association Analysis

This study tries to detect patterns using the technique of association analysis, which is a popular analytical method to detect or mine relations between certain parameters of complex data, and is widely used in learning analytics[18, 19, 20]. In our research, association analysis is conducted with the Apriori algorithm [14]. Each association rule consists of a condition part, a conclusion part, and three indicators: *support*, *confidence*, and *lift*. *Support* indicates how frequently the detected rule appears in the data set. *Confidence* indicates how often the rule has been found to be true. *Lift* indicates how the rule is effective in the data set. Let A and B be the condition part and the conclusion part of a rule (A,B), respectively. Let $p(X)$ be the probability of a phenomenon X within the data set. *Support* of the rule (A,B), $\mathrm{sup}(A,B)$, is calculated by Formula (1). *Confidence* of the rule (A,B), $\mathrm{conf}(A,B)$, is calculated by Formula (2). *Lift* of the rule (A,B), $\mathrm{lift}(A,B)$, is calculated by Formula (3).

$$\mathrm{sup}(A,B) = p(A \cap B) \tag{1}$$

$$\mathrm{conf}(A,B) = p(B \mid A) = \mathrm{sup}(A,B)/p(A) \tag{2}$$

$$\mathrm{lift}(A,B) = p(B \mid A)/p(B) = \sup(A,B)/p(A)/p(B) \tag{3}$$

Mouri et al. [15, 16] used association analysis for detecting useful patterns from ubiquitous learning logs accumulated in a ubiquitous learning system [17]. They indicated that learning activities in informal settings can be improved by providing appropriate association rules for the learners.

This approach contributes to finding out more detailed patterns of learners' errors and artifact making processes. This study expects that the improvement of learners' artifact making processes based on the patterns leads to prevention of the learners' errors.

3 Design of System KIfU 3.0

UML class diagrams are often used to describe conceptual models [4]. A previous study focusing on CM education by Tanaka et al. [13], developed a UML class diagram editor, KIfU 2.0, to collect data of learners' thinking during their artifact making processes. Based on that system, we developed KIfU 3.0 to collect data of the artifact making processes. KIfU 3.0 was implemented by Google Web Toolkit (GWT), which is a Web application framework for Java. KIfU 3.0 also uses MySQL as a database to accumulate the log data.

3.1 Interface of KIfU 3.0

Fig. 1 shows the interface of the class diagram drawer of KIfU 3.0.

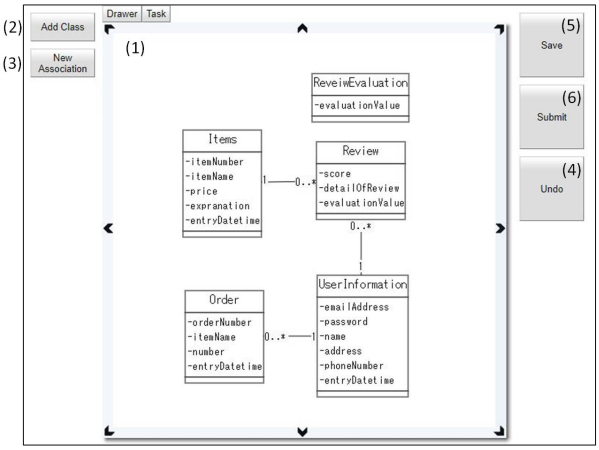

Figure 1: Interface of KIfU 3.0.

The drawer has the following functions:

(1) Canvas: A learner makes his/her diagram on the canvas. The learner can open a context menu by right clicking on the canvas, and he/she can add/remove/edit the diagram elements using the menu. KIfU 3.0 records the log data each time the learner operates the canvas.

(2) Add class: The learner can add a new class on the canvas using *Add class* button. When he/she clicks the button, a new class appears under the mouse cursor. He/she can move the new class to any location on the canvas using the mouse, and drop it at the location by clicking on the canvas.

(3) New Association: The learner can add a new relation between two classes in the diagram using *New Association* button. There are two ways to make a new relation. One is that the learner clicks the button, and selects two classes by clicking them. In the other method, the learner selects a class, clicks the button, and selects another class. Either way, a new relation appears between the classes.

(4) Undo: The learner can cancel the previous operation on his/her artifact by clicking the button. In that case, KIfU 3.0 logs an "Undo" and restores the state of his/her artifact before the previous operation instead of deleting the previous operation.

(5) Save: The learner can save the diagram. When the *Save* button is clicked, KIfU 3.0 records a "Save" log with the state of the learner's artifact. In this timing, KIfU 3.0 encodes the artifact into string data and stores it into the database.

(6) Submit: The learners can submit the diagram to their teacher by clicking the button. KIfU 3.0 records a "Submit" log with the latest state of the learner's artifact in the same way as the *Save* operation.

3.2 Log Data from KIfU 3.0

The system collects the log data of learners' operations on their diagrams. Table 1 shows a sample of the the log data.

Table 1: Sample of the log data from KIfU 3.0.

Target type	Event type	Target Id	Before edit	After edit	Timestamp
Class Name	Create	9	Class8	Stock	2016-11-29 10:23:16
Class	Place	9	152, 124	315, 282	2016-11-29 10:23:18
Attribute	Remove	1	-number OfStock	(none)	2016-11-29 10:25:48
(note)	Save	(none)	(none)	(none)	2016-11-29 10:25:55
Class Name	Edit	9	Stock	Items	2016-11-29 10:29:10
⋮	⋮	⋮	⋮	⋮	⋮

In Table 1, *Target type* refers to the type of the edit target; *Event type* refers to the type of event: 'Start', 'Create', 'Edit', 'Remove', 'Place', 'Undo', 'Save', and 'Submit'; and *Target Id* refers to the id of the target element in the artifact. In addition, *Before edit* and *After edit* refer to the target object strings before and after the edit, respectively; and *Timestamp* refers to the time when the event occurred. For example, the event in the top row of Table 1 logs an operation where the name of a class "Class8" was changed to "Stock". Therefore, the values of 'Before edit' and 'After edit' are "Class8" and "Stock", respectively. In KIfU 3.0, the initial default name of a class consists of the string "Class" followed by an integer. KIfU 3.0 generates the "Create" log when the default name is changed to any original name by a user. Hence, the value of 'Event type' of the log in Table 1 is "Create".

4 Analysis Method

4.1 Data collection

We collected data of learners' artifact making processes from a CM exercise using KIfU 3.0. The exercise was carried out as a part of a database design course in a university. The course was offered for the second-year undergraduate students. Thirty-two learners participated in the exercise: all of them were male (age about 20 years except one).

In the exercise, the learners were asked to make a conceptual model from a given requirement specification (RS) in 30 minutes. A part of the RS about a function of user registration is as follows.

RS of an online shopping system.

- User registration

When a user performs user registration, he/she inputs his/her email address, password, name, address, and phone number into the system. The system records the user information that was input by the user. In addition, the system records the date and time when the information was input into the system.

Fig. 2 shows the expected answer. In order to make the correct model, the learners have to detect the necessary elements from the RS.

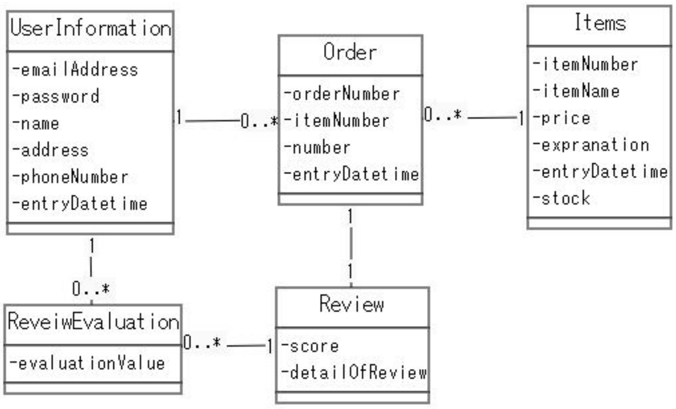

Figure 2: Expected answer of the exercise.

Then, this study detected errors in the learners' artifacts by comparing the learner's artifacts with the expected answer. Fig. 3 shows an example of a learner's artifact.

The artifact shown in Fig 3 does not contain some necessary elements such as an entity *Order*, and attributes *User.entryDatetime* and *Items.stock*. Table 2 shows a part of the transaction data regarding the artifact errors, where 0s and 1s represent the successes and failures, respectively, in the detection of corresponding elements. We evaluated all learners' artifacts in this way.

In the artifact making processes, almost all of the learners refined their artifacts. However, the number of refinements varied depending on the individuals, so we classified the learners based on it. If a learner refines his/her artifact frequently, they perform the 'Edit' and 'Remove' operations many times. Therefore, we decided to use the total number of 'Edit' and 'Remove' operations as an indicator of the number of refinements. Fig. 4 shows

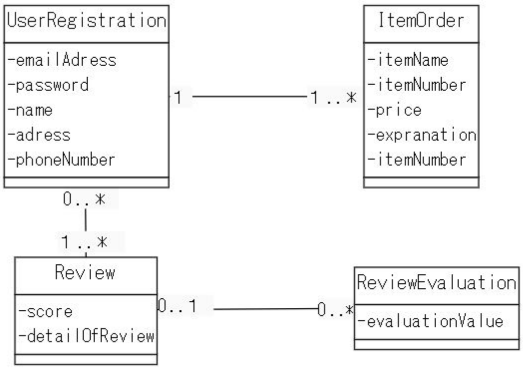

Figure 3: Example of a learner's artifact.

Table 2: Transaction data regarding a learner's errors.

User				Review	User.	User.	Items.	Relation.
Information	Items	Order	Review	Evaluation	mailaddress	entryDatetime	Stock	User–ReviewEvaluation
0	0	1	0	0	0	1	1	1

the histogram of the learners' number of refinements. The minimum number of 'Edit' operations by one learner was 0, whereas the maximum one was 19. The average number of 'Edit' operations by one learner was 6.3. The minimum number of 'Remove' operations by one learner was 0, whereas the maximum number was 20. The average number of 'Remove' operations by one learner was 6.8. The minimum number of refinements by one learner was 0, whereas the maximum number was 33. The average number of refinements by one learner was 13.2.

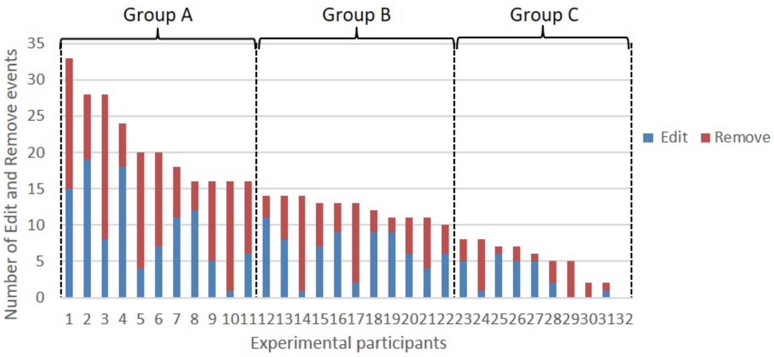

Figure 4: Numbers of the learners' refinements.

This study classified the learners into 3 groups A, B, and C:

A: One third of the learners whose numbers of refinements are large. Their numbers of refinements ranged from 16 to 33.

B: One third of the learners whose numbers of refinements are medium. Their numbers of refinements ranged from 10 to 14.

C: One third of the learners whose numbers of refinements are small. Their numbers of refinements ranged from 0 to 8.

4.2 Data Analysis and Result

We applied association analysis to the collected data by using the "arules" package of the R language. As a result, 506 association rules were detected. From this set, we decided to adopt only those rules that satisfy the conditions: (1) *support* ≥ 0.2 and (2) *confidence* ≥ 0.6. The experts investigated the association rules that do not satisfy these conditions, and found were no meaningful or effective rules. Table 3 shows the six rules derived by association analysis. *UserInformation.entryDatetime* and *Items.stock* in Table 3 are the elements of the expected answer. *UserInformation.entryDatetime* means the attribute *entryDatetime,* which is one of the attributes of the class *UserInformation*. *Items.stock* means the attribute *stock,* which is one of the attributes of the class *Items*. In Table 3, they indicate that the learners failed to detect these elements. 'Group A', 'Group B', and 'Group C' in Table 3 mean that the learners belong to group A, group B, and group C that described above.

Table 3: Result of association analysis.

No.	Condition part	Conclusion part	Support	Confidence	Lift
1	UserInformation. entryDatetime	Items.stock	0.50	0.89	1.24
2	Items.stock	UserInformation. entryDatetime	0.50	0.70	1.24
3	Group C	UserInformation. entryDatetime	0.22	0.70	1.24
4	Group A	Items.stock	0.22	0.64	0.89
5	Group B	Items.stock	0.25	0.73	1.01
6	Group C	Items.stock	0.25	0.80	1.11

5 Discussion

The experimental result depends on reading comprehension and carefulness of the participants and the difficulty of the RS. However, this study observes tendencies described below.

To carry out CM correctly, learners have to read an RS to comprehend the necessary elements of the model. Hence, novice learners may fail to comprehend the elements if they cannot understand the business flow from the RS. According to the submitted artifacts of the learners in the experiment, many of them failed to notice the necessity of the following elements: *Order*, *ReviewEvaluation*, *UserInformation.entryDatetime*, and *Items.stock*, where *Order* and *ReviewEvaluation* are class elements.

First, we focus on the learners' errors. We extracted two rules (No. 1 and No. 2 in Table 3) regarding the errors that co-occur in the artifacts. The rule No. 1 means that about 90% of the learners who failed to detect *UserInformation.entryDatetime* failed to detect *Items.stock*. The rule No. 2 means that about 70% of the learners who failed to detect *Items.stock* failed to detect *UserInformation.entryDatetime*. To notice the necessity of *UserInformation.entryDatetime*, the learners had to understand the RS clearly. To notice the necessity of *Items.stock*, the learners had to understand the business flow from the RS. These results suggest that the learners who read the RS with poor attention probably failed to notice both of them.

Next, we extracted a rule (No. 3) regarding the classification of learners by the numbers of refinements. *UserInformation.entryDatetime* is one of the elements hard to find. Rule No.3 means that about 70% of the learners who classified into group C failed to detect *UserInformation.entryDatetime*. We also extracted three rules from No. 4 to No. 6 regarding *Items.stock* in all groups. Among these three rules, the rule No. 6 has the highest confidence while the rule No. 4 has the lowest confidence. This result indicates that the learners with fewer numbers of refinements have a tendency to fail to notice the necessity of the attribute *Items.stock*. It suggests in general that there is a relation between the characteristics of the learners' artifacts making processes and their tendency for making errors.

Though *Order* was the most important element in the RS, only 16 learners could find it. They understood the RS better than the others. We conducted additional association analysis using the data of the learners who could find *Order*. Table 4 shows the result.

Table 4: Result of additional association analysis.

No.	Condition part	Conclusion part	Support	Confidence	Lift
1	Group C	UserInformation. entryDatetime	0.31	1.00	1.78
2	Group C	Items.stock	0.31	1.00	1.60

Since *Confidences* of rules in Table 4 were both 1.00, we know that all learners in the group C failed to detect *UserInformation.entryDatetime* and *Items.stock*. If we focus on the learners who could understand the RS, we see a positive correlation between the learners' errors and their fewer refinements. From the result and discussion, we suggest that there is a relation between the learners' error patterns and the characteristics of their artifact making processes.

6 Conclusion

The goal of this study was aimed to find (1) patterns of the learners' errors in their artifacts, (2) characteristics of their artifact making processes, and (3) relations between the patterns of errors and the characteristics, in CM exercises. We conducted an experiment in a CM exercise using our system KIfU 3.0 to collect data of learners' artifacts and the artifact making processes. We applied association analysis to the collected data. From these result, we found some rules regarding the errors that tend to co-occur in a learner's artifact. We also found some characteristics of learners' artifact making processes regarding their refinements of the artifacts. Moreover, we found that learners, who do not refine their artifacts many times tend to fail to detect the elements that are difficult to comprehend from the RS, for example, *UserInformation.entryDatetime* and *Items.stock* in the RS for the experiment. Conversely, the learners who refine their artifacts frequently tend to succeed in detecting such elements.

In the future work, we plan to address effective feedbacks in the CM exercises based on the data of learners' artifacts and the process by which they are created. We will also apply techniques of sequential pattern mining to the data of the artifact making processes to find patterns for an effective feedbacks.

Acknowledgments

We are deeply grateful to Dr. Hiroaki Hashiura. We would also like to express special thanks to the students of Nippon Institute of Technology who participated in the experiment. We appreciate Dr. Bipin Indurkhya for proofreading this paper. The part of this research work was supported by the Japan Society for the Promotion of Science, Grant-in-Aid for Young Scientists (B) under No. 17K12947.

References

[1] D. Batra and J.G. Davis, "Conceptual data modelling in database design: similarities and differences between expert and novice designers," Int'l J. Man-Machine Studies, vol. 37, no. 1, pp. 83–101, 1992.

[2] M. Ohki and K. Akiyama, "A propose of the conceptual modeling criteria and its validity evaluation (in Japanese)," IEICE Trans. Information and Systems, vol. J84-D1, no. 6, pp. 723–735, 2001.

[3] V.C. Storey and R.C. Goldstein, "Knowledge-based approaches to database design," MIS Quarterly, vol. 17, no. 1, pp. 25–46, 1993.

[4] M.A. Chilton, R. McHaney, and B. Chae, "Data modeling education: The changing technology," J. Information Systems Education, vol. 17, no. 1, pp. 17–20, 2006.

[5] R. Ramakrishnan and J. Gehrke, Database Management Systems, 3rd ed., McGraw-Hill Inc., New York, NY, USA, 2003.

[6] J.D. Ullman, H. Garcia-Molina, and J. Widom, Database Systems: The Complete Book, 1st ed., Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[7] S.D. Ryan, B. Bordoloi, and D.A. Harrison, "Acquiring conceptual data modeling skills: The effect of cooperative learning and self-efficacy on learning outcomes," SIGMIS Database, vol. 31, no. 4, pp. 9–24, 2000.

[8] M. Kayama, S. Ogata, K. Masumoto, M. Hashimoto, and M. Otani, "A practical conceptual modeling teaching method based on quantitative error analyses for novices learning to create error-free simple class diagrams," Proc. 2014 3rd Int'l Conf. Advanced Applied Informatics, pp. 616–622, 2014.

[9] L. Kazi, B. Radulović, I.F. Berković, and Z. Kazi, "Integration of conceptual data modeling methods: Higher education experiences," Proc. 2014 37th Int'l Convention on Information and Communication Technology, Electronics and Microelectronics, pp. 845–850, 2014.

[10] P.P. Chen, "The entity-relationship–Toward a unified view of data," ACM Trans. Database Systems, vol. 1, no. 1, pp. 9–36, 1976.

[11] P. Suraweera and A. Mitrovic, “An intelligent tutoring system for entity relationship modelling,” Int’l J. Artificial Intelligence in Education, vol. 14, pp. 375–417, 2004.

[12] H. Kung, L. Kung, and A. Gardiner, “Comparing Top-down with Bottom-up Approaches: Teaching Data Modeling,” Proc. Information Systems Educators Conf., New Orleans Louisiana, vol. 29, 2012.

[13] T. Tanaka, H. Hashiura, A. Hazeyama, S. Komiya, Y. Hirai, and K. Kaneko, “A method for collecting learners’ thinking process in ER modeling exercises and its application experiment,” Proc. 2016 Int’l IEEE Conf. Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress, pp.482-488, 2016.

[14] R. Agrawal, T. Imielinski, and A. Swami, “Mining association rules between sets of items in large databases,” Proc. ACM SIGMOD Int’l Conf. Management of Data, pp. 207–216, 1993.

[15] K. Mouri, H. Ogata, and N. Uosaki, “Ubiquitous learning analytics in the context of real-world language learning,” Proc. 5th Int’l Conf. Learning Analytics and Knowledge, pp. 378–382, 2015.

[16] K. Mouri, H. Ogata, and N. Uosaki, “Context-aware and Personalization Method based on Ubiquitous Learning Analytics,” J. Universal Computer Science, vol. 22, no. 10, pp. 1380–1397, 2016.

[17] H. Ogata, B. Hou, N. Uosaki, K. Mouri, and S. Liu, “Ubiquitous learning project using life-logging technology in Japan,” J. Educational Technology & Society, vol. 17, no. 2, pp. 85-100, 2014.

[18] G. Hwang, H. Chu, and C. Yin, “Objectives, Methodologies and Research Issues of Learning Analytics,” Interactive Learning Environments, vol. 25, no. 2, pp. 143–146, 2017.

[19] C. J. Yin, N. Uosaki, H. C. Chu, G. J. Hwang, J. J. Hwang, I. Hatono, E. Kumamoto, and Y. Tabata, “Learning Behavioral Pattern Analysis based on Students ’ Logs in Reading Digital Books,” Proc. Int’l Conf. Computers in Education, pp. 549–557, 2017.

[20] B. Minaei-Bidgoli, G. Kortemeyer, and W. Punch, “Association analysis for an online education system,” Proc. 2004 IEEE Int’l Conf. Information Reuse and Integration, pp. 504–509, 2004.