Computer Science and Information Systems 21(2):491–505 https://doi.org/10.2298/CSIS220830069C

A revised Girvan–Newman Clustering Algorithm for Cooperative Groups Detection in Programming Learning

Wen-Chih Chang

International Master Program in Information Technology and Applications,
National Pingtung University, Pingtung City 900, Taiwan
yilan.earnest@mail.nptu.edu.tw

Abstract. Learning to program is a challenging task for novices. Students vary substantially in their ability to understand complex and abstract topics in computer programming logic, such as loop logic, function recursion, arrays, passing parameters, and program structure design. Cooperative learning is an effective method of learning and teaching programming. In traditional cooperative learning, students group themselves, or teachers group students intuitively. This paper proposes a clustering method based on item response theory (IRT) and the revised Girvan–Newman clustering for clustering students by learning ability. Item response theory calculated the learner's ability and interpersonal relationship questionnaire generated by the social network analysis. The proposed method was validated by conducting a quasi-experimental test in a freshmen programming course, and the method significantly improved learning outcomes in this course.

Keywords: Learner ability, Girvan–Newman clustering, Social Network Analysis, Programming.

1. Introduction

Cooperative learning is a form of learning in which students learn and work together to accomplish shared goals. It has been applied in numerous fields. In most cases, cooperative learning is performed in small groups. Students in these groups discuss topics; through these discussions, all students learn and achieve beneficial outcomes. Cooperative learning can also be competitive; for example, groups might compete to see which group can answer the most questions in a limited time. Competitive group goals require all group members to work together to improve their learning. If the conditions in which competitive and purely cooperative learning should be applied are determined, a cooperative learning course can be designed for any subject.

Cooperative base groups are long-term, heterogeneous cooperative learning groups with stable membership [1]. Heterogeneous cooperative learning groups include students with different learning abilities. The term "stable membership" indicates that group members can work together over a long time or have good relationships. However, selecting people with good relationships in a class is challenging.

Girvan and Newman (2002) [2] proposed the Girvan–Newman clustering method for investigating communities. The authors test the method with computer-generated communities and real-world community structures. The result showed high sensitivity and reliability.

There are some studies, which applied AI and metaverse methods to support education. Omonayajo, Fadi, and Nadire (2022) [3] examined the smart technologies that have assisted smart education in achieving educational goals. These smart technologies enhanced the teaching and learning process in today's education. Yu and Lin (2022) [4] explained the data mining status and the college students' psychological health problems. This research used the decision tree to analyze the psychological health problem data.

Innovation thinking and computational thinking affect students' learning, which promotes students learning performance. Dagienė, Jevsikova, Stupurienė, and Juškevičienė (2022) [5] surveyed 52 countries with a qualitative study of 15 countries, which helped them to identify teachers' understanding level of computational thinking and its integration approach in the class activities. It is useful for e-learning systems and content developers to improve teachers' computational thinking. In the other research Zheng et al. (2022) [6] made a training system, that made the major in computer science students have better academic performance and significantly improved compared with the performance before the innovative thinking.

Dale's Cone of Learning [7] model states that activities in which students experience, discuss, do, and participate cause greater retention than simply reading, watching, or hearing. In cooperative learning, students must be active participants in discussions and must support their team members. Thus, cooperative learning activities improve student learning, understanding, and retention.

A teacher can flexibly modify their lecturing style or learning material to maximize teaching quality based on student feedback. However, teachers typically prepare their teaching materials before classes begin. Thus, predicting student learning ability is key for preparing appropriate class activities. However, measuring learner ability is challenging for teachers. Therefore, a method that can be used to estimate learner ability and cluster students appropriately to obtain learning groups comprising heterogeneous members would be of considerable benefit to teachers and student outcomes. Assessments are typically used to measure and analyze student performance and learning skills. These assessments also can be used as feedback for teachers and students, which is crucial in learning and development.

The remainder of this paper is organized as follows. Section 2 describes item response theory (IRT) and the adopted clustering method. Section 3 presents details regarding how IRT and clustering are used to estimate student learning ability and identify cluster learners. The experimental results are presented in section 4, and section 5 provides the conclusions of this study and suggestions for future research.

2. Related Studies

With the increased acceptance of e-learning, numerous researchers have proposed various student assessment methods. For example, the researchers [8] designed a teaching for students to assess the smartphone to study Geography. With simple test items, the proposed system provides individual learning profile and test analysis report for each student. The result shows an interesting approach and reveals the learning profile and test analysis for students is a good reply and suggestion for students. Some teachers applied the social network analysis clustering method [9] for cooperative learning in programming courses at the university. The relationship among all the classes is considered the connection between students. It shows significant differences in students' performance and scores. Some teachers used the combining flipped learning and online formative assessment platforms to enhance students' learning performance [10]. Research has increasingly focused on assessments to assist learning and teaching. IRT is often used to estimate learner ability. In IRT, the probability that a student answers a particular question (item) correctly is expressed using a continuously increasing graph called the item characteristic curve. The item characteristic curve is defined in terms of one, two, or three of the following parameters: item discrimination, item difficulty, and student guessing. Item discrimination refers to the extent to which an item discriminates between high- and low-ability students. Item difficulty indicates whether an item is easy or difficult, and student guessing can be included as a corrective factor if students are likely to guess the correct answer. Figure 1 presents the three-item characteristic curves of three items with the same discrimination of 1 and distinct difficulties of 1, 3, and 5.

The characteristic curve of each item in IRT is a logistic function that is expressed as follows.

In this function, e is Euler's number, b is the difficulty parameter (typically $-3 \le b \le 3$), a is the discrimination parameter (typically $-2.8 \le a \le 2.8$), $L = a(\Theta - b)$ is the logistic deviate (logic), and Θ indicates student ability level. A one-parameter item characteristic curve presents only the difficulty of the problem; the discrimination and guessing are ignored (set to 1). A one-parameter model (Equation 1) is expressed as follows:

$$P(\theta) = \frac{1}{1+e^{-a(\theta-b)}} \text{ ....Equation 1}$$

The two-parameter logistic model considers the discrimination and item difficulty (Equation 2), and the three-parameter logistic model (Equation 3) considers the discrimination, item difficulty, and the probability that a guess is correct c. A three-parameter model is expressed as follows:

$$P(\theta) = \frac{1}{1+e^{-a(\theta-b)}} = \frac{1}{1+e^{-1(\theta-b)}} \text{ ....Equation 2}$$

$$P(\theta) = c + (1-c)\frac{1}{1+e^{-a(\theta-b)}} \text{ ....Equation 3}$$

The parameter c can theoretically range between 0 and 1; in practice, values greater than 0.35 are rarely used.

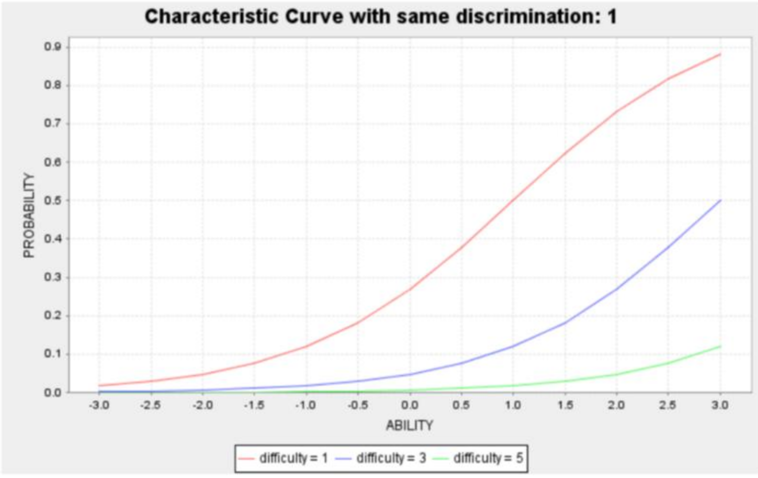

Fig. 1. Same discrimination of 1 with distinct difficulties in one-parameter models

Clustering is a method of organizing a collection of unlabeled data by grouping similar items. Clustering algorithms have been applied in biology, marketing, earthquake studies, and city planning. The K-means method is one of the most commonly used clustering techniques. This method is used to group a collection of data samples into *k* clusters based on a distance measurement. Distance is usually determined according to a data attribute, such as the price of a product, the score of a student, or the time and location of an earthquake. In this study, we clustered students in a learning community by using the Girvan–Newman (GN) community clustering algorithm. In social network analysis (SNA), social relationships between members of a social structure of any scale are considered to define nodes, ties, groups, and betweenness centrality [15]. In simple terms, SNA is a method of surveying all relationships among actors in a community [16]. Betweenness centrality indicates the extent to which a vertex or edge lies on a path between vertices. Nodes or vertices with high betweenness might have considerable influence on a network. Because of their presence on numerous paths, nodes or vertices can control considerable information flowing through a network.

3. Research Method

This paper proposes a methodology that combines K-means clustering with the GN community clustering algorithm, and the proposed methodology involves considering the distance (the betweenness value) between communities. Moreover, we propose a grouping algorithm combined with IRT for estimating learner ability to achieve heterogeneous groupings for cooperative learning.

3.1. Pretest

A learner's ability can be approximated by their test scores. However, the difficulty and discrimination of items differ; thus, students with the same score might still have different abilities. We applied the two-parameter logistic model based on IRT. As we mentioned in Section 2, the Two-parameter logistic model considers the discrimination and item difficulty (Equation 2). Using the discrimination and difficulty, we can get Θ which indicates the student's ability level.

We adopted Kelly's method to determine the item difficulty and discrimination indices. The best percentage for subsequent calculations was 27%, and acceptable percentages were 25%–33% [17]. We selected a percentage of 25% for these calculations. We then sorted students by their exam scores and defined the top and bottom 25% of students by test score. The total number of correct answers in the higher and lower groups for each question are denoted as PH and PL, respectively. The item difficulty index for each problem was calculated using the equation $b = (\mathrm{PH} + \mathrm{PL})/2$, and the item discrimination index for each problem was calculated using the equation $a = \mathrm{PH} - \mathrm{PL}$. The default learner ability θ was set as 1. The parameters were input into the item characteristic equation to obtain P for item 1. For any student, P was calculated for the 20 items to calculate the student's learning ability.

3.2. Learner Clustering in Cooperative Programming Learning

Learner clustering is critical for cooperative programming learning. We revised a social network clustering method (GN iteration) (Figure 2), a heterogeneous function, and then used a grouping algorithm for clustering.

3.3. GN iteration [14]:

(1) Compute the betweenness of every edge in the graph. For node X, perform a breadth-first search to determine the number of shortest paths from node X to each node, and assign these numbers as scores to each node.
(2) Beginning at the leaf nodes, calculate the credit of an edge as [1 + (sum of the edge credits)] × (score of the destination node/score of the starting node).
(3) Compute the credits of all edges in the graph G and repeat from step 1 until all nodes have been selected.
(4) Sum all the credits computed in step 2 and divide by 2. The result is the betweenness of each edge.
(5) Remove the edges with the highest betweenness.
(6) Compute the modularity Q of the communities split.
(7) If Q > 0.3–0.7, repeat from step 1. (0.3-0.7 is the experimental result for better performance)

Heterogeneous function is used to make sure learner ability is distributed in different levels. We applied Equation 2, $P(\theta)$ is the learner ability. With the discrimination index and difficulty index, the learner ability P(θ) can be calculated. Learner ability was classified as high, middle, and low. The most appropriate candidates were selected into teams according to the betweenness centrality and learner ability. Learner ability is calculated by item response theory.

```
IF(N >5)
      REPEAT
           FOR i=0 to n-1
                  LET B[i] BE betweenness centrality of edge i
                  IF B[i] > max_B
                  THEN max_B = B[i]
                         max_B_edge = i
                  ENDIF
           ENDFOR
            remove edge i from graph
       UNTIL number of edges in graph is 0
      //Divided into 2 groups
       Heterogeneous( );
ELSE IF ( 0< N && N<=5)
       Heterogeneous( );
```

*N is the number of nodes in the group graph, n is the number of edges in the group graph

Fig. 2. A revised GN algorithm

3.4. Quasi-Experimental Method and Posttest

This study referenced the research [9], which is designed based on a mixed approach. The difference part between the research [9] is the algorithm design and algorithm complexity comparison. This study also optimizes the Grouping algorithm. The study includes experimental and control groups. The experimental group has 34 male students and 10 female students. The control group has 38 male students and 6 female students. Two groups received the same teaching material and teaching progress in the semester. However, the clustering method in cooperative learning is different. The experimental group was clustered by social network analysis results, and the control group was clustered by the students they chose by themselves.

The experimental group of students was designed to answer two questions. The first question is “Who you will choose to be the team members?”. The second question is “Who is the person you will ask or discuss when you encounter some problems in learning programming course?”. Students can write 1~3 students’ names. The study applied 1st question SNA clustering result and a little modified based on 2nd question answer to generate the cooperative learning team members.

The course taught variables, control commands, loop, pointer, array, function, recursion, and project. It took 18 weeks, including preparation, pretest (week 1~week 2), clustering of team members, posttest (week 18), answer questionnaire, and interview procedures. The pretest is composed of five programming questions (such as int, double, calculate BMI, string decomposition, and if command operation).

T-test measures the difference between two means, which may or may not be related to each other. It also indicates the probability of the differences to have happened by chance. A T-test is usually a test for two experimental numbers, which has a difference between them. For example, the experimental result is better than the control result.

Paired Sample is the hypothesis testing conducted when two groups belong to the same group or population. In this experiment, P is a statistical measure that helps to determine whether the hypothesis is correct or not. Furthermore, it assists in demonstrating the significance of the results. In the experimental design, the null hypothesis is a default situation that which there is no relationship between two measured phenomena. H0 denotes the null hypothesis. The other hypothesis H1, is the researcher's belief that the null hypothesis is false. P-value is a number between 0 and 1. The significance level is a predefined threshold, which is set at 0.05 generally.

The assumption of statistics test is performed below:

Null Hypothesis: $\mu_d = 0$: There is no significance between our revised GN clustering algorithm and the students' willingness group.

Alternative Hypothesis: $H_1: \mu_d \neq 0$: There is significance between our revised GN clustering algorithm and the students' willingness group.

The pretest scores of the experimental and control groups were not significantly different (*p* = 0.804, Table 1). However, the posttest scores of the experimental group were significantly higher than their pretest scores (*p* = 0.0001, Table 2) and the posttest scores of the control group (*p* = 0.024, Table 3).

Table 1. Pretest scores [9]

Group	Average Score	Standard Deviation	t	p	Significance
Experimental Group	52.93	11.38	-0.248	0.804	No significance
Control Group	53.45	7.86			

Table 2. Pretest and post-test scores of the experimental and control groups [9]

Group	Test	Average Score	Standard Deviation	t	p	Significance
Experimental Group	Pretest	52.93	11.38	-3.796	0.0001 ***	No significance
	Posttest	63.72	16.94			
Control Group	Pretest	53.45	7.86	-0.737	0.465	
	Posttest	55.43	16.89			

*: $p \leq 0.05$, **: $p \leq 0.01$, ***: $p \leq 0.001$

Table 3. Posttest scores of the experimental and control groups [9]

Group	Average Score	Standard Deviation	t	p	Significance
Experimental Group	63.72	16.94	2.298	0.024*	Significance
Control Group	53.43	16.89			

*: $p \leq 0.05$, **: $p \leq 0.01$, ***: $p \leq 0.001$

The statistic test shows that there is significance. We reject the null hypothesis, or it means that the alternative hypothesis is accepted. The average mean between using our revised GN clustering algorithm and the students' willingness group is a significant difference of 0.52. Moreover, the standard deviation between the two groups is similar at 11.38 and 7.86. This implies that the learning performance in the pretest is quite the same, however, some students in our revised GN clustering algorithm can improve the average mean from 52.93 to 63.72. This concludes that using our revised GN clustering algorithm has an efficiency to apply in programming learning.

The final T-test interpretation could be obtained in either of the two ways:

A null hypothesis signifies that the difference between the means is zero and where both the means are shown as equal.

An alternate hypothesis implies the difference between the means is different from zero. This hypothesis rejects the null hypothesis, indicating that the data set is quite accurate and not by chance.

3.5. Comparison of Clustering Method

Our proposed revised GN clustering algorithm has better clustering result for teaching and learning, cost less time than K-means clustering, and is significant in the quasi-experimental method described in section 3.4. The following introduces the compared clustering results for teaching need, time complexity comparison, pretest, and posttest learning effectiveness comparison.

There are three clustering algorithms, such as k-means clustering, our revised GN clustering, and students' willingness clustering. Figure 3 shows the solid line in the k-means clustering algorithm, which shows the result in 6 groups. Each group with large, varied student number (1,3,2,3,15,20). K-means results are not appropriate for real classroom teaching. The second method is our revised GN clustering, shown in the dashed line in Figure 3, which generated 9 groups with close numbers (4,5,5,5,5,5,5,5,5). Our revised GN clustering result is the best case for cooperative learning. The third is traditional teaching, which is grouped by students' willingness to cooperate in learning with a dotted line in Figure 3 with large, varied numbers (3,4,5,5,6,10,11). Most teachers need to negotiate with students in groups again.

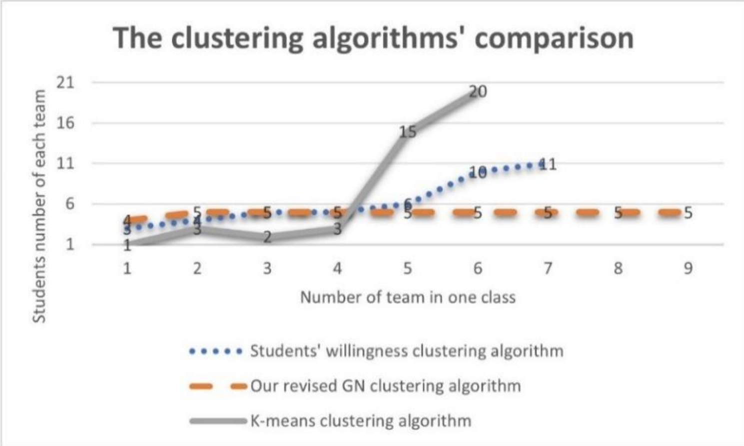

Fig. 3. The cluster difference among Weka K-means clustering [18], students' willingness clustering, and our revised GN clustering

Time complexity

The K-Means algorithm is a good example, which is one of the most widely used in literature. K-Means algorithm time complexity is O(N) [19]. The Girvan-Newman algorithm time complexity is $O(N^3)$ and $O(m^2n)$ [19], which we adapted in our research. In this experiment and most teaching experience, the number of the class will not be bigger than 100 students. Therefore, the cost time will not have a large influence.

Pretest and posttest learning effectiveness comparison

Figure 4 shows the mean score in our revised GN algorithm makes students' scores improve from pretest 53.45 to posttest 63.72 (Figure 4, dashed line). The student's willingness mean score improved from the pretest 52.93 to the posttest 55.43 (Figure 4, dotted line). Section 3.4 concludes that using our revised GN clustering algorithm is more efficient than the other method in programming learning.

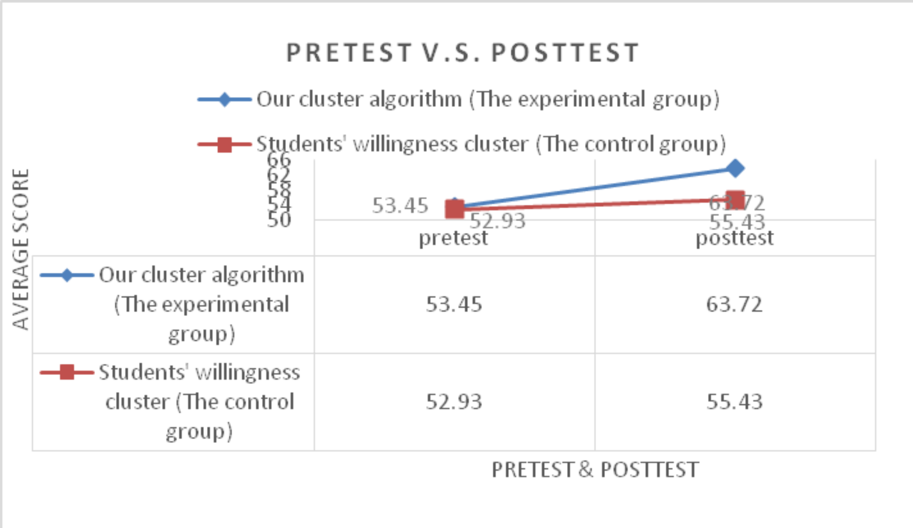

Fig. 4. The comparison between students' willingness clustering and our clustering

Social Network Analysis & Interpersonal Relationship

Social network analysis applied the student's interpersonal relationship questionnaire to generate the SNA (social network analysis) graph. The first step in our algorithm is using the student's interpersonal relationship to produce the SNA graph.

In Figure 5(a), the number of each group is too different. In a cooperative learning environment, it is not easy to arrange more than 5 students in a group. The more students in a group, the learning efficiency becomes lower. The cooperating learning suggested number is four to five. All the teams are arranged with high, middle, and low-score group students. Figure 5(b) shows the final clustering result.

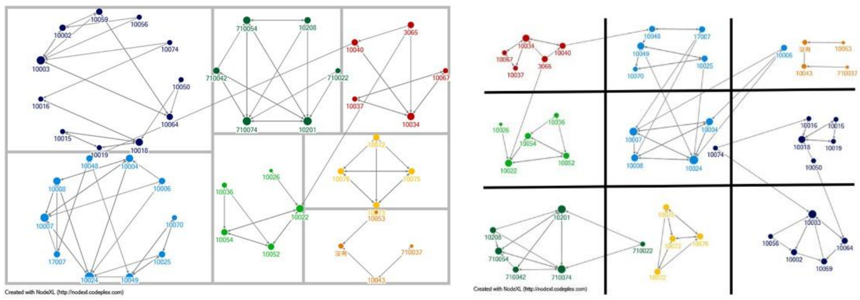

Fig. 5. (a) Original experimental group clustering graph (b) The experimental group clustering graph after our revised GN clustering

4. Lag Sequential Analysis of Programming Exam Videos

Content analysis involves the study of documents and communication methods, including text, images, audio, or video. Scientists have applied content analysis to investigate communication patterns in a replicable and systematic manner. The noninvasive nature of content analysis is a crucial advantage when using it for examining social phenomena. Researchers can simulate social situations, collect survey questionnaires, or record videos to reveal patterns. Computer-based content analysis methods are being increasingly used [20-23]. Video, answers to open-ended questions, newspaper articles, online discussions, medical records, or experimental observations can be systematically analyzed after conversion to a machine-readable format. The input is analyzed and coded into categories to reveal patterns. Some computer-assisted methods can reduce the time required to analyze large digital data sets. Certain studies have eliminated the need to establish intercoder reliability for multiple human coders. However, human coders are still critical in content analysis because they are superior to computers for recognizing nuance and latent meanings in text.

Table 4. Coding scheme

Code	Phase / Description
C1	Coding/Debug: The process of students writing programs or debugging, and it also includes debugging, copying and pasting code, and compilation and testing.
C2	Search for information: Search for information on the internet, watch programming videos, or read other programs. It involves Internet references, assignments previously uploaded to the platform, reference materials, files on the platform, or recorded teaching videos.
C3	Review questions/code/Debug information: Viewing or reading the exam questions, the student’s program, the debug information, or program execution results.
C4	Thinking: Think about how to code or what to do next.
C5	Others: Other than the above four codes. For example, asking a teacher a question on the platform, opening a folder or file, saving a file, saving as a new file, switching windows quickly with no obvious action, and other miscellaneous actions not covered by the other four categories.

Our coding schema is introduced in Table 4. The problem-solving behaviors displayed by students in our recorded videos were analyzed and labeled using five codes. The recorded videos are recorded on students’ computer screens, which is automatic recording. We can record students’ movements when they are solving problems and writing programs.

Lag sequential analysis [24,25] has become an important tool for researchers of interpersonal interaction. This method [26] enables one to explore and summarize cross-dependencies occurring in complex interactive sequences of behavior.

Lag sequential analysis of individual interactions was explored as a tool to generate hypotheses regarding the social control of inappropriate classroom behavior of students with severe behavior disorders. Gunter et al. [27] proposed three coded events (student hand raise, teacher attention, and the “stop code”) that were identified as highly related to the student's disruptive behavior. The results are discussed in terms of the usefulness of the analysis procedures in contributing to the functional analysis of students' classroom behavior.

This study [28] then discusses the different learning behavior patterns based on the theoretical framework of Hofstede’s National Cultural Dimensions (NCD). The obtained results highlighted that students from each culture behave differently due to several interconnecting factors, such as educational traditions.

This study [29] examines it in the context of 83 elementary schoolers’ mobile serious game-playing behaviors. Lag-sequential analysis of the participants’ observed behavioral patterns, and of differences in such patterns between two performance subgroups (i.e., students with high vs. low academic performance), yielded two main findings. First, all these young learners exhibited knowledge construction, and moved smoothly from lower to higher phases of it in the mobile environment; and second, the high-performing group attained a deeper level of knowledge construction through the negotiation of meaning than the low-performing group did. Some theoretical and practical implications of these results are also discussed.

This study applied lag sequential analysis to find out the obvious transition of the programming actions. The five codes are discussed and referenced [30] the problem-solving code. The coding scheme definition is listed in the following.

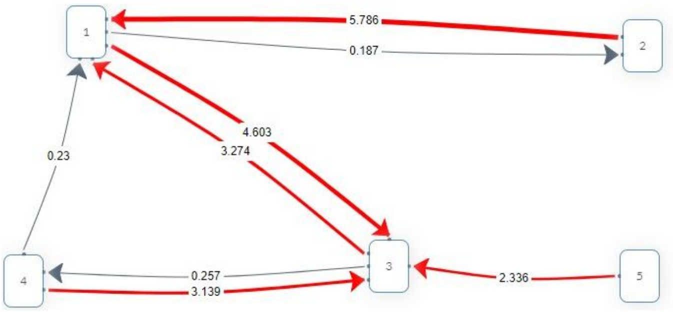

Fig. 6. The experimental group’s sequential analysis

This section describes the sequential analysis of the experimental group and compares the group to check if there is obvious movement from one state to the other state. The distribution of the experimental group content analysis is as follows: C1 1223 times, 51.3%; C2 325 times, 13.6%; C3 862 times 36.2%; C4 525 times, 22%; C5 130 times, 5.5%. The distribution of the experimental group sequential analysis is as follows: C1 to C3 4603 times; C2 to C1 5786 times; C3 to C1 3274 times; C4 to C3 3139 times; C5 to C3 2336 times. According to the [31] Allison and Liker (1982) used the z score to calculate. We obtain the following obvious transition in Figure 6.

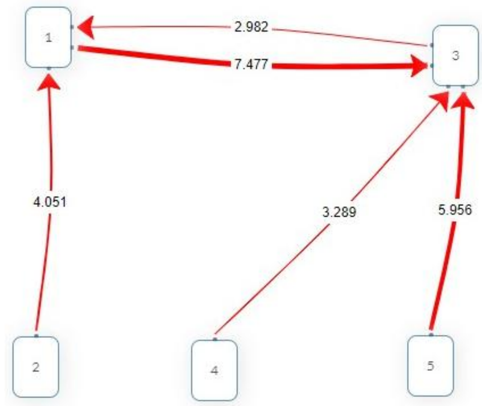

Fig. 7. The compared group's sequential analysis

The distribution of the experimental group content analysis is as follows: C1 968 times, 45.8%; C2 386 times, 18.3%; C3 996 times 47.1%; C4 315 times, 14.9%; C5 128 times, 6.1%. The distribution of the experimental group sequential analysis is as follows: C1 to C3 7477 times; C2 to C1 4051 times; C3 to C1 2982 times; C4 to C3 3289 times; C5 to C3 5956 times. According to [31] Allison and Liker (1982) used the z score to calculate. We obtain the following obvious transition in Figure 7.

The Experimental group has two obvious loops, C1C3, C3C4 and C1C2, even C1 to C2 and C3 toC4 are not so obvious. However, it shows the experimental group with our revised GN clustering makes more learning efficiency in programming.

5. Conclusion

In this study, GN clustering based on the betweenness value between students was combined with a grouping algorithm based on IRT to develop a combined methodology for estimating learner ability to achieve heterogeneous student grouping for cooperative learning. An experimental group of students clustered using our proposed SNA approach had significantly higher post-test scores than did a control group of students who grouped themselves.

Acknowledgment. I thank the Taiwanese Ministry of Education for financially supporting this study under the Teaching Practice Research Program. And thanks to Prof. Yang Hsin-Che for discussing this study idea, and Chang An-Ray for data analysis.

References

1. 1. Johnson, D. W., Johnson, R., Houbec, E.: Cooperation in the classroom (7th ed.). Edina, MN: Interaction Book Company. (2008)

2. Girvan M., Newman M. E. J.: Community structure in social and biological networks, in Proceedings of the National Academy of Sciences, 99 (12) pp.7821-7826. (2002)
3. Omonayajo, B., Al-Turjman, F., Cavus, N.: Interactive and Innovative Technologies for Smart Education. Computer Science and Information Systems, 19(3), pp.1549-1564. (2022)
4. Yu, J., Lin, J.: Data mining technology in the analysis of college students' psychological problems. Computer Science and Information Systems, 19(3), pp.1583-1596. (2022)
5. Dagienė, V., Jevsikova, T., Stupurienė, G., Juškevičienė, A.: Teaching Computational Thinking in Primary Schools: Worldwide Trends and Teachers' Attitudes. Computer Science and Information Systems, Vol. 19(1), pp.1-24. (2022)
6. Zhang G., Wang X., Zhao R.L., Wang C., Wang C.: Construction of Innovative Thinking Training System for Computer Majors under the Background of New Engineering Subject. Computer Science and Information Systems, 19(3), pp.1499-1516. (2022)
7. Dale, E.: Audio-Visual Methods in Teaching (3rd ed., p. 108). Holt, Rinehart & Winston, New York: Dryden Press. (1969)
8. Yang H.-C., Chang W.C.: Ubiquitous smartphone platform for K-7 students learning geography in Taiwan. Multimedia Tools Application 76, pp.11651–11668 (2017).
9. Chang W.-C.: Integrating Social Network Analysis, Content Analysis and Sequence Analysis with Cooperative Learning in the Programming Courses: A Case Study, International Journal of Engineering Education, Vol. 39 No. 2, pp.397-408. (2022)
10. Cheng S.-C., Cheng Y.-P., and Huang Y.-M., Enhancing Students' Learning Performance by Combining Flipped Learning and Online Formative Assessment Platform, International Journal of Engineering Education, Vol. 39 No. 2, pp.409–419. (2022)
11. Otte, E.; Rousseau, R.: Social network analysis: a powerful strategy, also for the information sciences". Journal of Information Science. 28(6), pp.441-453. (2002)
12. Milgram, S., The Small World Problem. Psychology Today, 2, pp.60-67. (1967)
13. Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks, Nature 393, pp.440-442. (1998)
14. Barnes, J.A.: Class and Committees in a Norwegian Island Parish. Human Relations, 7, pp.39-58. (1954)
15. Pattison P.: Algebraic Models for Social Network, Cambridge University Press, (1993)
16. Tseng, C. H.: The Type of Network Organization of MNC"s Subsidiary in Taiwan and the Management Mechanisms Adopted by MNC"s Headquarters, National Cheng-chi University, Unpublished doctoral thesis (1995)
17. Kelly T.L.: The selection of upper and lower groups for the validation of test items. Journal of Educational Psychology, 30, pp.17-24. (1939)
18. Weka official website at the University of Waikato in New Zealand, *https://www.cs.waikato.ac.nz/~ml/weka/*
19. Elbattah, M., Roushdy, M., Aref, M., Salem M.A.: Large-Scale Entity Clustering Based on Structural Similarity within Knowledge Graphs, Book chapter of Big Data Analytics: Tools, Technology for Effective Planning, pp.311-334. (2017)
20. Jeffery Chiang, *https://medium.com/analytics-vidhya/girvan-newman-the-clustering-technique-in-network-analysis-27fe6d665c92*, Last Accessed 02/07/2022.
21. Pfeiffer, S., Stefan F., Wolfgang E.: Automatic audio content analysis, Technical Reports 96 (1996).
22. Grimmer, J., Brandon M. S.: Text as data: The promise and pitfalls of automatic content analysis methods for political texts, Political Analysis 21(3), pp.267-297. (2013)
23. Nasukawa, Tetsuya, Jeonghee Yi.: Sentiment analysis: Capturing favorability using natural language processing, in Proceedings of the 2nd international conference on Knowledge capture. ACM (2003)

24. Sackett, G. P.: The lag sequential analysis of contingency and cyclicity in behavioral interaction research. In J. D. Osofsky (Ed.), Handbook of infant development (pp. 623-649). New York: Wiley. (1979)
25. Sackett, G. P.: Lag sequential analysis as a data reduction technique in social interaction research. In D. B. Sawin, R. C. Hawkins. (1980)
26. Faraone S.V., Dorfman D.D.: Lag sequential analysis: Robust Statistical Methods, Psychological Bulletin, 101(2), pp.312-323. (1987)
27. Gunter, P. L., Jack, S. L., Shores, R.E., Carrell, D.E., Flowers, J.A: Lag Sequential Analysis as a Tool for Functional Analysis of Student Disruptive Behavior in Classrooms, Journal of Emotional and Behavioral Disorders, SAGE Publications Inc, 1(3), pp.1063-4266. (1993)
28. Tlili A., Wang H.H., Gao B.J., Shi Y.H., Nian Z.Y., Looi C.-K., Huang R.H.: Impact of cultural diversity on students' learning behavioral patterns in open and online courses: a lag sequential analysis approach, Interactive Learning Environments, pp.1-20. (2021)
29. Sun, Z., Lin, CH., Lv, K., Song J.: Knowledge-construction behaviors in a mobile learning environment: a lag-sequential analysis of group differences. Education Tech Research Dev 69, pp.533–551 (2021).
30. Hou, H. T., Sung Y. T., & Chang, K. E.: Exploring the behavioral patterns of an online knowledge sharing discussion activity among teachers with problem-solving strategy. Teachers and Teachers Education, 25(1), pp.101-108. (2009)
31. Allison P. D., Liker, J.K.: Analyzing sequential categorical data on dyadic Interaction, Psychological Bulletin, 91, pp.393-403. (1982)

Wen-Chih Chang received B.S., M.S., and Ph.D. degrees in computer science and information engineering from Tamkang University, Taipei, Taiwan. He has been an Associate Professor in the International master program in Information Technology and Applications at the National Pingtung University, Taiwan. His current research interests include social network analysis and AI in e-learning.

Received: August 30, 2022; Accepted: September 19, 2023.