Design and Development of a Stepwise Learning Environment for Problem Posing of Arithmetic Word Problem

Yusuke HAYASHI[a*], Nagito YAMAMOTO[a], Susumu SHIMAKAWA[b] & Tsukasa HIRASHIMA[a]

[a]*Graduate School of Advanced Science and Engineering, Hiroshima University, Japan*

[b]*Faculty of Engineering, Hiroshima University, Japan*

*hayashi@lel.hitoshima-u.ac.jp

Abstract: In this paper, we focus on learning through problem-posing in arithmetic word problems and propose an environment for stepwise learning of problem-posing. As a foundation for stepwise learning, we have summarized the relationship between problem-posing tasks as a problem-posing task graph by setting difficulty levels based on the number of conditions that must be considered during problem-posing. This is based on a domain model that summarizes the structural conditions necessary for arithmetic word problems to be valid. Based on this, we designed and developed a stepwise problem-posing learning environment called Monsakun Step. To verify its effectiveness, we conducted an experiment with university students. The results suggest that using Monsakun Step may be particularly effective for learners with a low structural understanding of arithmetic word problems.

Keywords: Problem-posing, Arithmetic word problems, Stepwise Learning, Domain modeling

1. Introduction

One of the important aspects in arithmetic and mathematics education is "mathematical modeling" (Sole, 2013). This involves converting problems from the "real world" into "mathematical problems" by modeling them as "mathematical models" in the "world of mathematics," and then seeking solutions to real-world problems by deriving "mathematical conclusions."

To represent real-world problems as "mathematical models," skills related to formulation (simplification, idealization, approximation, setting assumptions, symbolization, formalization) are required. To process "mathematical models" and obtain "mathematical conclusions," skills related to mathematical operations (mathematical theories and methods) are needed. To treat "mathematical conclusions" as solutions in the real world, the ability to interpret, evaluate, and compare these solutions is necessary. In this way, "mathematical modeling" aims to teach the utility of mathematics, the relationship between real phenomena and mathematics, and methods for solving new problems by moving back and forth between the "real world" and the "world of mathematics." Based on this concept, various efforts are being made in arithmetic and mathematics education to support understanding the relationship between the "real world" and the "world of mathematics." These include considerations on the process of generating mathematical models from real problems, analysis of conditions for teaching materials where mathematical modeling occurs, and implementation of practical lessons aimed at fostering mathematical modeling based on analysis.

In this research, we position arithmetic word problems as one method for learning mathematical modeling. There are both pros and cons to this within arithmetic and mathematics education (Verschaffel et al., 2020). Kaiser (2017) states that the characteristic of mathematical modeling lies in its authenticity, and that arithmetic word problems, for example, are problems that are overly simplified by abstracting real problems, and thus are separate from mathematical modeling. On the other hand, Verschaffel et al. *(2020) state that while there are indeed differences between problems set in the context of schools and authentic real-world modeling problems faced outside of school, word problems that are appropriately designed and handled in the context of mathematics education can function as accurate and valuable "simulations" of authentic mathematical modeling problems that may be encountered in real life. Here, following Verschaffel et al.'s thinking, we position arithmetic word problems as one step in learning mathematical modeling.

This research focuses on the formulation step for representing "real world" problems as "mathematical models," particularly on formulation that enables mathematical operations. To represent a "real world" problem as a "mathematical model" capable of obtaining mathematical conclusions, not only must the "real world" and "mathematical model" correspond, but the "mathematical model" must also be able to derive a solution as a mathematical conclusion by applying mathematical operations. In arithmetic word problems, the "real world" is described in a simplified manner, but the goal is to understand how elements obtained from this simplified description correspond to a "mathematical model" that can lead to a solution as a mathematical conclusion.

As a method for this, this research focuses on the creation of arithmetic word problems. Creating arithmetic word problems (= problem-posing learning) is more effective than solving them. Problem-posing learning is suggested to be effective in making learners conscious of the application conditions of solution methods (arithmetic word problems → mathematical expressions) and in mastering the use of solution methods based on appropriate understanding of these application conditions (solution method establishment). Therefore, it can provide support for understanding relationships between elements, which is also emphasized in mathematical modeling. Although problem-posing learning is considered an effective learning method for understanding relationships, it has been difficult to implement in educational settings due to the high burden on both learners and instructors.

The sentence integration type problem-posing learning environment "Monsakun" is designed to alleviate the difficulties of implementing problem-posing learning in educational settings and to support understanding of the structural relationship between problem statements, which are simplifications of the "real world," and mathematical expressions as "mathematical models" (Yamamoto et al., 2012)(Hayashi et.al, 2021).

In the sentence-integration-based problem-posing task within Monsakun, learners are provided with multiple sentences for constructing arithmetic word problems. Learners select the necessary sentences from the given options to pose the required arithmetic word problems. The arithmetic word problems created by the learners are automatically evaluated, and feedback is provided as to whether the problems meet the requirements. Learners continue to work on each problem-posing task until they answer correctly. Through the process of posing various arithmetic word problems, learners can progress by learning which combinations of sentences form valid arithmetic word problems and which combinations align with the given requirements.

Previous research has shown that as learners engage in problem-posing tasks, their mistakes decrease, and correct understanding is gradually formed. In other words, this learning environment enables learners to learn in a "heuristic" and "self-regulated" manner. However, detailed investigations into the learning process have revealed that some learners may not develop correct understanding despite making progress in the tasks (Hayashi et al., 2024). Therefore, it is suggested that a learning environment that enables learners to not only learn in "heuristic" and "self-regulated" manner, but also "stepwise" and "adaptively," is necessary.

In this study, we propose a stepwise problem-posing learning environment by structuring problem-posing tasks as problem-posing task graphs based on the required thinking process and deriving a series of learning paths from this structure.

2. Sentence Integration Type Problem-Posing Learning Environment "Monsakun"

The arithmetic word problem posing learning environment "Monsakun" is a system based on the Triplet-structure Model for problem-posing learning. The Triplet-structure Model is a framework for describing the relationship between the problem statement of arithmetic word problems and mathematical expressions. It breaks down the problem statement into single sentences for each quantity and correlates their semantic quantitative relationships with mathematical expressions.

Monsakun is a learning environment based on the Triplet-structure Model where learners can create problems by selecting and assembling single sentences, and receive immediate judgement and feedback. It was developed to resolve the difficulties of problem-posing learning and to enable problem posing on tablet or PC terminals through immediate judgement and feedback. Figure 1 show the screenshot of Monsakun. Learners pose problems with provided sentences without creating sentences by themselves. Figure 2 shows an example of feedback to learners. Monsakun provide feedback about the correctness of problems posed by learners with the types of failure.

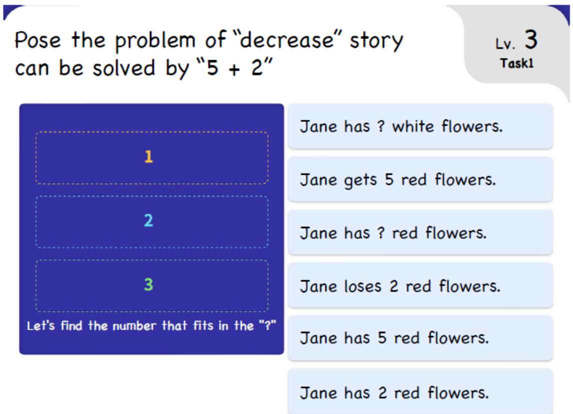

Figure 1. An example of a problem-posing task

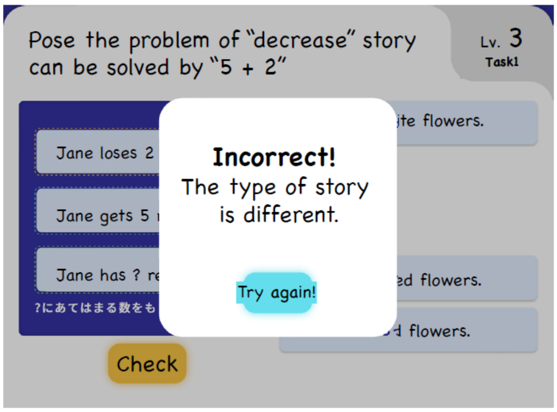

Figure 2. An example of feedback on a problem-posing task

Monsakun deals with unit arithmetic word problems and aims to promote understanding of the relationship between arithmetic word problems and mathematical expressions through sentence integration problem posing. A unit arithmetic word problem is one that sets one unknown number in a story where quantitative relationships can be expressed with a single operation. For example, "There are 3 apples. I bought 5 apples, so now there are 8" is a unit arithmetic story that can be expressed by the equation "3 + 5 = 8". When this is broken down into three sentences: "There are 3 apples.", "I bought 5 apples.", "There are 8 apples.", and the number of apples after the increase is set as the unknown, it becomes a unit arithmetic word problem.

In the Triplet-structure Model, unit arithmetic word problems are formulated as a combination of two types of sentences: existence sentences expressing the existence of quantities, and relational sentences expressing the relationship between two quantities. In the previous example, it can be composed of two existence sentences ("There are 3 apples", "There are 8 apples") and one relational sentence ("I bought 5 apples").

Sentence integration problem posing in Monsakun involves creating problems by combining existence and relational sentences expressed as single sentences. Each problem in Monsakun has a unique correct answer, allowing learners to understand the relationships of problems according to constraints.

Monsakun implements problem-posing learning activities with an awareness of quantitative relationships expressed in arithmetic word problems (real world). While there are claims that the participants dealt with in arithmetic word problems simplify the "real world" and do not make learners conscious of quantitative relationships, this research argues that through problem-posing learning, it makes learners conscious of the relationship between the "real world" and the "world of mathematics" dealt with in mathematical modeling by making them conscious of quantitative relationships.

This research aims to explain the relationship between "the specific situation shown by arithmetic word problems" and "the mathematical expressions derived from that situation". Monsakun promotes understanding of the relationship between the "real world" and the "world of mathematics" through learning in in "heuristic" and "self-regulated” manner”. In contrast, this research aims to create an environment where "relationships" can be learned "step by step" and "adaptively".

Monsakun presents conditions for creating arithmetic word problems and aims for learners to understand the relationships of problems that fit the constraints by finding specific relationships between quantities and mathematical expressions while heuristically learning the underlying principles. In this sense, Monsakun is an environment for learning "quantitative relationships" in arithmetic word problems in "heuristic" and "self-regulated manner".

While Monsakun supports promoting understanding of the relationship between arithmetic word problems and mathematical expressions by learning this relationship heuristically, the ability to grasp relationships is also important in mathematical modeling. Also, while Monsakun's learning policy is in self-regulated manner, appropriate support may be necessary depending on the learner's ability and learning situation.

Therefore, this research aims to make the "heuristic" relationships "step-by-step" and provide "adaptive" support for learners who find it difficult to learn in "self-regulated manner". The purpose of this research is to create a "stepwise" and "adaptive" exercise environment to support the promotion of understanding the relationship between the "real world" and the "world of mathematics" dealt with in mathematical modeling.

3. Structuring Arithmetic Word Problem-Posing Tasks

One of the important aspects in arithmetic and mathematics education is "mathematical modeling" (Sole, 2013). This involves converting problems from the "real world" into "mathematical problems" by modeling them as "mathematical models" in the "world of mathematics," and then seeking solutions to real-world problems by deriving "mathematical

3.1 Triplet-structure Model

Hirashima et al. propose the Triplet-structure Model, which structures arithmetic word problems as being composed of three sentences (Hirashima, et al., 2014). In the Triplet-structure Model, the components of arithmetic word problems are proposed as a combination of two types of sentences: existence sentences and relational sentences.

Existence sentences statically indicate the quantity of "things," such as "There are 3 apples" or "There are 5 children." Relational sentences connect two quantities and correspond to the types of stories. In addition, and subtraction, four types of stories are defined: "combination," "comparison," "increase," and "decrease" (Riley, et al. 1983). For example, in combination, it expresses the sum relationship of two different quantities, such as "There are 8 apples and oranges in total," while in decrease, it expresses the temporal change of a quantity, such as "I ate 2 apples."

The model demonstrates that by combining two existence sentences and one relational sentence, it's possible to compose a text that represents the equation 2 + 3 = 5, such as "There are 2 apples," "I bought 3 apples," "There are 5 apples." Furthermore, by making any of these numbers an unknown, it shows that arithmetic word problems can be constructed. For example, "There are 2 apples," "I bought 3 apples," "How many apples are there?" which can be solved with "2 + 3," or "How many apples were there?", "I bought 3 apples," "There are 5 apples," which can be solved with "5 - 2."

3.2 *Problem-Posing by Sentence Integration*

Hayashi et al. (2021) position problem-posing activities as a constraint satisfaction problem based on the three-sentence composition mode. In sentence integration type problem-posing in Monsakun, there are a total of five constraints (Hirashima, et al., 2014): three based on the Triplet-structure Model and two based on the task conditions. Problems must be created satisfying all five of these constraints.

First, based on the Triplet-structure Model, the constraints of "Sentence Structure," "Object," and "Quantity" must be met. To satisfy "Sentence Structure," two of the three chosen sentence cards must be existence sentences and one must be a relational sentence. To satisfy "Object," the object relationships of the selected sentence cards must be correctly structured. To satisfy "Quantity," the quantity relationships of the selected sentence cards must be correctly structured.

Additionally, based on the task conditions, the constraints of "expression" and “story” must be met. In Monsakun, as the task specifies the "expression" or "story type," there are conditions for "expression " and “story” corresponding to the task. To satisfy " expression," the condition of the task's expression must be met. For example, if the condition is to create a problem that can be calculated with "3 + 5 = ?", the equation established by the three sentence cards must be "3 + 5 = ?". In this case, if the sentence cards are arranged in the order "There are 2 apples," "I bought ? apples," "There are 5 apples," it would result in the expression "2 + ? = 5," which would be incorrect.

To satisfy "Story," one of the three sentence cards must be a relational sentence, and the story of this relational sentence must match the story specified in the task. For example, if the task is to create a "subtraction" story, arranging the sentence cards in the order "There are 5 oranges," "I bought 1 orange," "There are ? oranges" would create an "addition" story. Therefore, as the actual story created differs from the task's story, it would be incorrect.

From the above, the constraints that should be satisfied in problem-posing tasks for arithmetic word problems are the five constraints of "Sentence Structure," "Object," "Quantity," "Expression," and "Story." Understanding these constraints equates to understanding the relationship between arithmetic word problems and mathematical expressions.

3.3 *Problem-Posing by Sentence Integration*

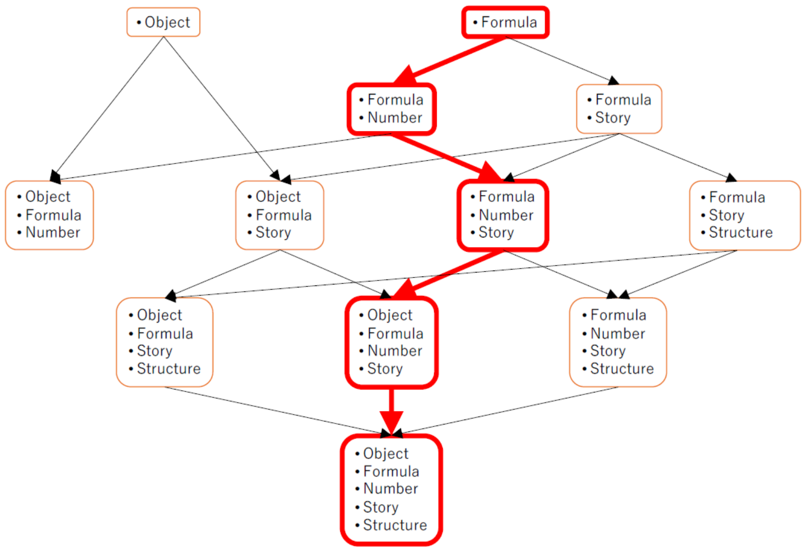

Figure 3. Problem-Posing Task Graph for Arithmetic Word Problems

We systematized problem-posing tasks and designed them to be stepwise and differential in terms of constraints. Figure 3 shows the Problem-Posing Task Graph for Arithmetic Word Problems. Each node represents the type of problem-posing task with the constraint to be considered and the arrows represent possible path among tasks. To systematize, they classified and organized problem-posing tasks according to constraints, considered the possibility of creating problem-posing tasks for each constraint, and designed them with learning sequences in mind based on the number and connections of constraints for creatable problem-posing tasks.

The design is stepwise, with constraints gradually increasing one by one, and differential in that if one tries to solve the next problem, they cannot proceed unless they have solved the problem with the prerequisite constraints. Specifically, when tackling a problem-posing task with the "Expression and Quantity" constraint, one cannot engage with the task without first addressing the "expression" constraint, which is a prerequisite.

While sentence integration type problem-posing in Monsakun requires considering all five constraints simultaneously, this approach allows for stepwise and differential consideration of constraints. Additionally, by explicitly stating which constraints should be considered, it becomes clearer which constraints should be focused on.

4. Stepwise Problem-Posing Learning Environment "Monsakun Step"

In this research, we designed and developed a stepwise problem-posing learning environment by selecting and implementing one path from the problem-posing task graph for arithmetic word problems. The problem-posing task graph for arithmetic word problems does not uniquely determine transitions between problem-posing tasks, resulting in various possible paths. This could potentially serve as a basis for adaptive path selection based on the learner's state of understanding. However, currently, it is difficult for learners or teachers to select appropriate problem-posing tasks because we have not established a technique to measure the learner's state of understanding. Therefore, we verify the effectiveness of a stepwise problem-posing learning environment by constructing a learning environment that selects one path from the problem-posing task graph. The path we used this time was selected with the assumption of progressing learning from basic elements, but it is thought that the learning effect may change depending on this selection, so further consideration is necessary in the future. The path emphasized by red in Figure 3 shows the one we selected this study. This path is not always the best one and it is necessary to compare with the others in the future work.

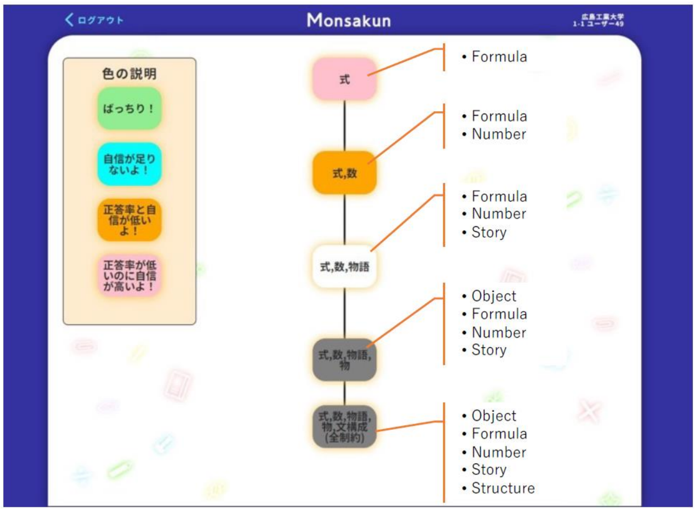

Figure 4. Problem-posing task selection screen

When learners use the problem-posing learning environment, it is displayed as shown in Figure 4. Due to space limitations in this paper, we omit the details, but in addition to judging correctness, we have made it possible to display the learner's state of understanding in a simplified manner by having them input their confidence in their answer at the time of their initial answer,. In the current setting, once a learner solves a group of problem-posing tasks, they can select the next level of problem-posing tasks. Each group of problem-posing tasks consists of 16 problems.

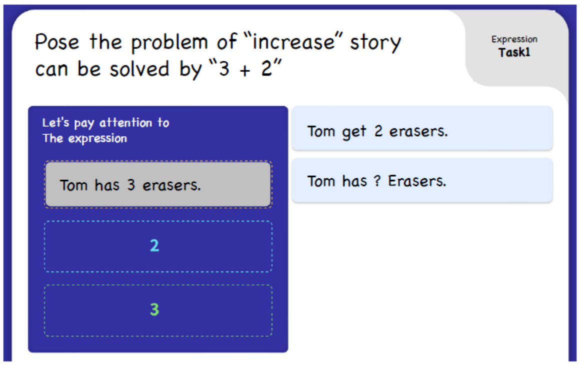

Figure 5. Example of an "expression" problem-posing task

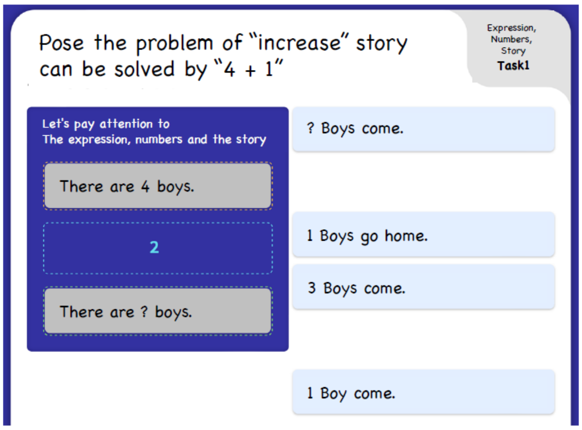

Figure 6. Example of an "equation / number / story" problem-posing task

Figures 5 and 6 show examples of problem-posing tasks. Figure 5 is an example of an "equation" problem-posing task. In this task, based on the constraints of the Triplet-structure Model, only whether the equation that can be correctly generated from the posed word problem matches the one specified in the problem-posing task is considered, while other constraints such as "number," "story," "object," and "sentence composition" are not to be mistaken. In this problem-posing task, the choice is which of the two sentences on the right to use as the second sentence. The correct answer is to use "I received 2 erasers" as the second sentence. If this sentence is used as the third sentence, it becomes a problem statement where it's unclear whether the number before receiving is 3 or ?, making it impossible to set up an equation, thus making it a problem-posing task related to the "equation" constraint. Figure 6 is an example of an "equation/number/story" problem-posing task. In this case, depending on which sentences are used, an answer that does not satisfy one or more of the "equation," "number," or "story" constraints can be created.

In Monsakun Step, by clarifying the constraints that need to be considered in each individual problem-posing task, learners can gradually learn the types and contents of constraints necessary for establishing arithmetic word problems that can be solved with a single addition or subtraction operation.

5. Preliminary Evaluation by University Students

A preliminary evaluation was conducted with university students to investigate whether they could explain the relationship between "the specific situation shown by arithmetic word problems" and "the mathematical expressions derived from that situation" through the use of Monsakun Step. The participants were 18 students from the information science department and graduate school of a private university in Hiroshima Prefecture. The survey procedure is as follows:

1. Pre-test (10 minutes)
2. Explanation of Monsakun (5 minutes)
3. Use of Monsakun (15 minutes)
4. Intermediate test (10 minutes)
5. Explanation time for Monsakun Step (5 minutes)
6. Use of Monsakun Step (25 minutes)
7. Post-test (10 minutes)
8. Questionnaire (5 minutes)

To measure the effect of the exercises using Monsakun and Monsakun Step, three tests (pre, intermediate, and post) were conducted. An example of the test problem is shown in Table 1. The content of each test was identical, consisting of 8 problems, each composed of sub-questions 1-3. Each sub-question is set up as follows:

Table 1. *Example of a test problem*

Problem1	The following arithmetic word problem is given: "There are 5 oranges. 3 pears were bought. There are ? oranges."
Sub-question 1	Can the above problem be calculated using "5 + 3"? • It can be calculated using "5 + 3" • It cannot be calculated using "5 + 3"
Sub-question 2	Please choose the reasons why it cannot be calculated (multiple selections allowed): ☐ The objects are incorrect ☐ The expression is incorrect ☐ The quantities are incorrect ☐ The story is incorrect ☐ The sentence structure is incorrect
Sub-question 3	Please explain in your own words the reason why the calculation cannot be performed.

Sub-question 1: Given an arithmetic word problem and a mathematical expression, choose whether it's possible to create the expression from the word problem
Sub-question 2: (Only for those who chose "not possible" in question 1) Check the constraints that are the cause
Sub-question 3: (Only for those who chose "not possible" in question 1) Explain verbalize) the cause in their own words

As it's possible to give multiple answers in sub-questions 2 and 3, there may be extra answers included. Therefore, the evaluation was done using the F-value (F1-score) based on precision and recall. For the evaluation of sub-question 3, the authors set standard descriptions for the content written by the participants and confirmed the presence or absence of these descriptions in each answer.

Figure 7 shows the results of sub-question 2, including the median and the first and third quartiles of each test, as well as the p-values and effect sizes. The p-values were calculated using the Brunner-Munzel test, adjusted by the Holm method. The effect size was calculated using r (rank-biserial correlation). Although there were significant differences at the 1% level between the pre- and the middle-test, there is not between the middle- and the post-test. From this, it can be said that there was no change in sub-question 2 even after using Monsakun Step.

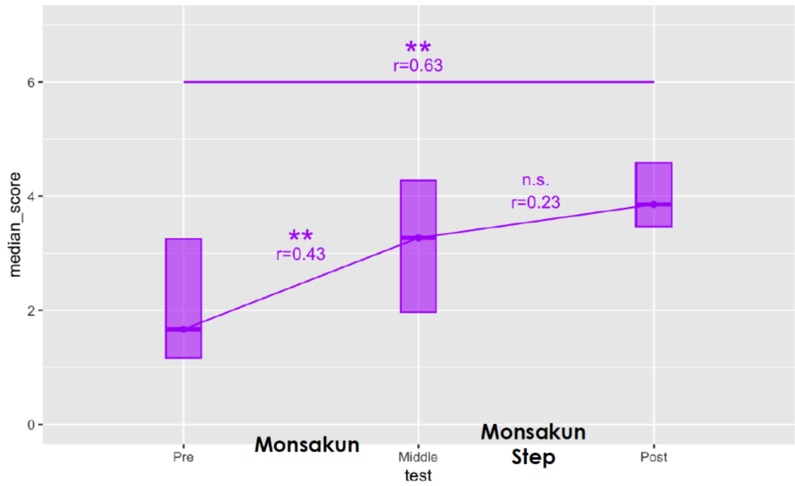

Figure 7. Transition of F-value (F1-score) in pre, intermediate, and post-tests for sub-question 2

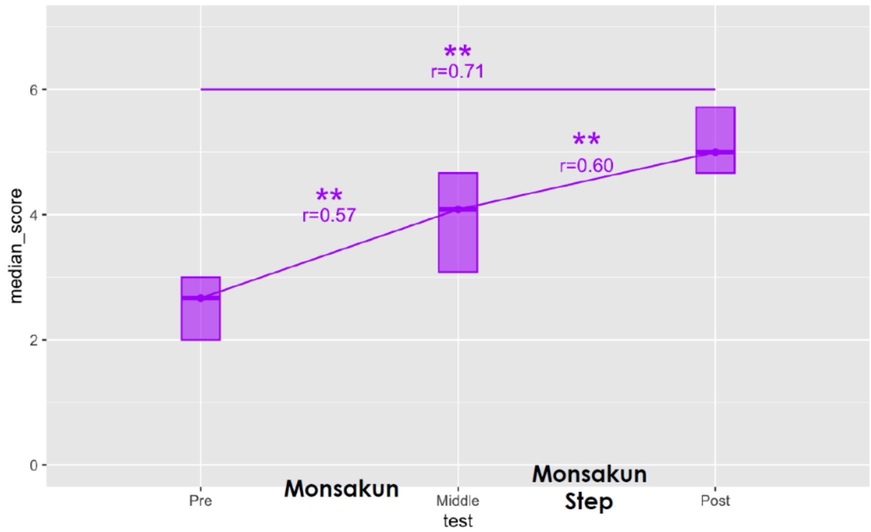

Figure 8. Transition of F-value (F1-score) in pre, intermediate, and post-tests for sub-question 3

Figure 8 shows the results of sub-question 3, including the median and the first and third quartiles of each test, as well as the p-values and effect sizes. The p-values were calculated using the same methods as for sub-question 2. There was a significant difference at the 1% level between middle- and the post-test as well as between the pre- and the middle-test. From this, it can be said that there was also change even after using Monsakun Step in sub-question 3.

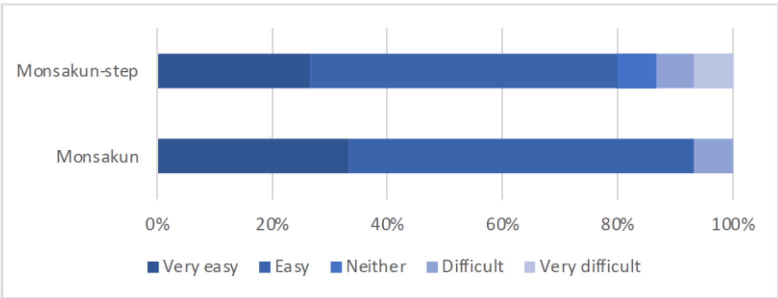

Figure 9. the results of the questionnaire

Figure 9 shows the results of the questionnaire. The questions were, "Was it difficult to pose problems in Monsakun?" and "Was it difficult to pose problems in Monsakun-Step?" The response trends were similar for both systems, with most participants answering that both were easy.

6. Conclusion

In this paper, we focused on formulation in mathematical modeling, particularly formulation that enables mathematical operations in the "world of mathematics," and proposed Monsakun Step, a stepwise exercise environment for problem-posing learning of arithmetic word problems as an exercise for this purpose. In designing Monsakun Step, based on the Triplet-structure Model, we structured sentence integration type problem-posing tasks as a problem-posing task graph, defining the relationship from problem-posing tasks with fewer elements to consider to those with more elements to consider, thereby defining the stages of problem-posing tasks. By traversing this problem-posing task graph, stepwise problem-posing exercises can be realized.

This time, we developed an experimental stepwise problem-posing learning environment by extracting one path from the problem-posing task graph, and conducted an experiment with university students as participants for a preliminary evaluation. As a result, it was suggested that the environment is effective in understanding the structure of arithmetic word problems, and it has a learning effect on aspects that cannot be fully learned with Monsakun alone.

Future tasks include confirming the effectiveness of Monsakun Step through verification of learning effects with more participants, as well as considering the construction of an adaptive problem-posing learning environment using the problem-posing task graph. In addition to the problem-posing task graph, we want to consider estimating the learner's understanding status (construction of a learner model) and proposing adaptive problem-posing tasks based on the combination of the problem-posing task graph and the learner's understanding status.

Acknowledgements

We would like to express our gratitude to Prof. Shimpei Matsumoto of the Faculty of Information Sciences at Hiroshima Institute of Technology, and to the students of the Faculty of Information Sciences and the Graduate School of Engineering at Hiroshima Institute of Technology for their cooperation in this experiment.

References

Hayashi, Y., Tsudaka, N., Iwai, K., & Hirashima, T. (2021). *Problem posing with representation conversion model for learning the condition of addition and subtraction word problems.* Proc. of ICCE2021, 2, 60-69.

Hayashi, Y., Ohsaki, A., Yamamoto, T., Hirashima, T. (2024) Finding and Understanding an Impasse in Learning by Assembling with Epistemic Network Analysis. In Companion Proceedings 14th International Conference on Learning Analytics & Knowledge (LAK24), 284-286.

Hirashima, T., Yamamoto, S. and Hayashi, Y. (2014) Triplet Structure Model of Arithmetical Word Problems for Learning by Problem-Posing," *Proceedings of HCI international 2014 Part II*, pp. 42–50 (2014)

Kaiser, G (2017) The teaching and learning of mathematical modeling. In J. Cai (Ed.), Compendium for research in mathematics education, pp. 267–291. Reston, VA: National Council of Teachers of Mathematics

Riley, M. S, Greeno, J. G, Heller, J. I. (1983) Development of children's problem-solving ability in arithmetic, H. P. Ginsburg (Ed.). (1983) The development of mathematical thinking. New York: Academic Press, 153-196

Sole, M. (2013). A primer for mathematical modeling. *Journal of Mathematics Education at Teachers College*, 4, 44–49.

Verschaffel, L., Schukajlow, S., Star, J. & Van Dooren, W. (2020) Word problems in mathematics education: a survey. ZDM 52, 1–16

Yamamoto, S., Kanbe, T., Yoshida, Y., Maeda, K., & Hirashima, T. (2012) A case study of learning by problem-posing in introductory phase of arithmetic word problems. In Proceedings of the 20th International Conference on Computers in Education, 25-32.