Web-based Application for Visual Representation of Learners' Problem-Posing Learning Pattern

Ahmad Afif Supianto1, Satrio Agung Wicaksono2, Fitra A. Bachtiar3, Admaja Dwi Herlambang4, Yusuke Hayashi5, Tsukasa Hirashima6

1,2,3,4 Faculty of Computer Science, Brawijaya University Malang, Indonesia
5,6 Department of Information Engineering, Graduate School of Engineering, Hiroshima University, 1-4-1 Kagamiyama, Higashi-hiroshima, Hiroshima 739-8527, Japan

{afif.supianto, satrio, fitra.bachtiar, herlambang}@ub.ac.id, {hayashi, tsukasa}@lel.hiroshima-u.ac.jp

Received 04 November 2018; accepted 27 June 2019

Abstract. The analysis of learner’s learning process on an interactive learning media tends to involve a huge amount of data gathering. The analysis aims to explore patterns and relationship of such data to understand the learning experience and identifying learner’s trap states in the learning process. Such data analysis tends to be a nuisance for stakeholders such as class instructors (i.e., teachers) and educational researchers. This research suggests the development of a web-based software application that utilizes the use of visual artifacts as an approach of educational data mining to analyze learner’s learning process in an interactive learning media. The developed software aims to visualize the learner’s activity sequence, furthermore, to identify learning paths toward bottleneck states due to lack of understanding of a problem given. Such information is then passed to the instructors as key information to create proper feedback to the learners based on its results. As a case study, this research uses the data log of Monsakun, a digital learning environment that focuses on exercising arithmetic using story-based question by using a problem-posing approach with the integration of mathematical sentences. An investigation is conducted to the data and the results Furthermore, the result of investigations shows that there are four groups of learning trajectory pattern, they are smart patterns, adventure patterns, peer patterns, and cyclic patterns. The application provides visual information in regard to learners' problem-posing sequence activities, and through it, feedback to learners is expected to be proper according to the learning patterns found.

Keywords: web-based application, visualizations, learning pattern, problem-posing process

1 Introduction

Problem-posing is defined as an act of building a problem and new questions that aims to explore a particular condition or to re-formulate a problem based on existing problems [1]. The act of allowing students and learners to propose their own questions promotes a more flexible thought, improves problem-solving skill, expands their perception regarding mathematics, and consolidate basic concepts [2,3].
Technology-based approach has been made to create a practical implementation of a learning method using problem-posing approaches, primarily to determine the impact of student’s self-assessment and its mathematical creativity [4], alongside with that, technological approach also allows the development of automatic diagnostic function which enables assessment and evaluations while also giving feedback for each compiled question [5]. A diagnostic evaluation method is a form of an automated agent-assessment. Furthermore, an interactive learning system called Monsakun has been developed using agent-assessment focusing on the basic arithmetic of addition and subtraction in the form of story-based questions [6]. Although the usage of Monsakun has been proved to be useful for learning arithmetics using the problem-posing approach in classes [7,8,9], it is essential to investigate the validity of the story creation process on Monsakun’s story-based questions.

Several kinds of research have been explicitly proposed to discuss the learning activity analysis using Monsakun. Hirashima et al. analyze whether the learners can form a story-based question based on the discussion related to the number of questions formulated by the learner [6]. Hirashima and Kurayama analyze the learning effect of Monsakun by comparing the problem-solving score using pre- and post-test approach from the paper test [10]. Hasanah et al. examine a more advanced part by researching the learning and thinking process of the learner based on their first sentence choice in Monsakun via a Binomial test [11]. The result shows that the selection of the first sentence changes based on the learner’s experience. In regards to the question formulation process in Monsakun, further analysis has been conducted on the entire process of generating the problems. The result shows an improving effect of how the learners pose the problems with higher validity [12] and avoid as many mistakes as possible [13].

Although multiple types of research have analyzed the log data from Monsakun, a software to explore and monitor the learning activity based on Monsakun log data has yet been created. Moreover, the data monitored are visualized into meaningful and informative visualizations. Datalog visualization is a useful approach to interpret the learner’s thinking process and is intended to improve cognition regarding the data [14]. Visualization can be fully be used to understand the log data step-by-step that is created in a digital learning environment – in this case, Monsakun. Shneiderman claims that the integration of data mining and data visualization on tools development allows for a more thorough and effective exploration towards hidden data implications [15].

The information regarding the learner’s learning process in an interactive learning application might be an integral part that can support their learning process, for example when they reached a trap condition [16]. By understanding the trap situation, proper feedback can be given to solve such a problem, and this can be improved further with a visual representation. The visualization is expected to provide a better description regarding the learner activity and their process of reaching a particular state. Moreover, the visualization approach on Monsakun can also be applied to other fields other than arithmetic and in any interactive learning applications.
2 Activities in a Problem-Posing Learning System

Monsakun has been developed as an interactive digital learning media to learn with problem-posing approach via sentence integration in the form of triplet structure model [17]. This model integrates story-based questions along with arithmetical problems by using three main components of numbers: the first operand, the second operand, and the result. The interface of Monsakun consists of three main parts: question-arranging area, sentence cards, and diagnosis button, as depicted in Fig. 1. Monsakun provides the learner with exercise to face the questions, and the learner is required to arrange the problem to meet with a particular requirement.

The question-arranging area on the left side of the interface displays the “requirement” and “card slots.” The requirement consists of two critical aspects, which are the type of story and the mathematical formula required to be formed on such problems. The learner can construct story-based problems by inputting the sentence into the card slots in the bottom of the area. Sentence cards are displayed on the right side of the interface. The learner can move the sentence cards by dragging and dropping it into the card slots. There are more than three cards provided; thus not all sentence cards are required to be used on forming such problems based on the requirement. The unused sentence cards are meant to be a decoy to test the learner’s understanding. The last component is the button located at the bottom of the screen, the diagnosis button. The diagnosis button is used to check the combination of the sentence cards that are arranged by the learners.

The learners can choose several cards and put them into the card slots, arrange them based on the requirement given. This flow depicts the main activity of the learners, consisting of dragging and dropping the sentence cards into the card slots based on the given requirement and checking them by using the diagnosis button. The Monsakun tracks the learner’s activity in the form of sentence combination on the card slots and stores the activity in log files. An example of Monsakun log data can be seen in Fig. 2(a). The log consists of level, assignment/question, set or remove actions, and sentence compositions. The stars (*) symbol in the composition shows that the card slot still not appointed yet.
3 Web-based Application

We develop a web-based application for tracking and mining pattern of arithmetic problem-posing learning processes that uses the data from Monsakun data log. The application converts the data log in Monsakun into malleable data and generates a graphical representation that can be understood by teachers to determine the learners' obstacles when they learn arithmetic word problems using Monsakun. The system architecture is depicted in Fig. 3.

Fig. 2. (a) Monsakun log data. (b) Converted log data stored in DBMS.

Fig. 3. The application architecture.

Data Conversion Module collects the log files that represent learners' activity in Monsakun. The module is also used to converts the log file into MySQL database to attributes inside a table (see Fig. 2(b)). This step will create a database that can be easily queried. The datasets (tables) are required as an input of the next modules. The next modules focus on creating a graphical representation from the database into an easily understood data visualization. The modules are separated into three parts, which are:
Data Formulation Module: (a) extracts and/or filters data from the database and choose only the relevant data for a particular visualization, and (b) manipulates the selected data and generates the data into a specific format, such as statistics to add more meaningful information.

Data Visualizing Module: transforms the data that has been formulated into a geometrical representation with proper attributes such as color, thickness, and opacity. This step changes the tables from the previous module into a graphical format that is easily understood by a human. It is the core process of the visualization process.

Viewing Visualization Module: Generates the visualization in the form of a mapping process and specifies the parameters of the visualization, such as scale, position, and orientation. The users control the settings as they want to.

The application architecture is based on a web-based technology that is built upon three main components: the database component, the middle component, and the interface component, as depicted in Fig. 4. Each component has its own tasks that contribute to the application. Each component on a web-based technology requires specific software. The application uses a MySQL database as a data storage facility and is created by using PHP and JavaScript programming language. PHP is used to communicate with the server, while JavaScript is used to communicate with the users to increase interactivity. The software is run on a web server to control the system flow and a web browser to access such application.

Fig. 4. Web-based technology used on the application.

4 Visualizations

In this section, the graphical representation of the application is illustrated. Fig. 5 shows that the application web page is divided into two parts, which are the menu on the left (number 1), and the visualization display on the right (number 2). Fig. 5 also represents the welcome page of the application that is filled with the summary of the
data condition. The welcome page displays statistical data of the learners and their activity. The data shown on the welcome page may include aggregation data such as the total of learners involved with the exercise on Monsakun, the total of questions answered, and the total of action steps done by the learners. In addition to that, the data are also depicted in a chart form displaying the spread of the data regarding the learners that are able to finish the problem.

Fig. 5. The welcome page.

4.1 Activity Sequences

Fig. 6 depicts the total and steps sequence of how the learners in posing the problem. Graphically, the data are represented in the form of 3 indexes. The indexes represent the sentence card index that was placed on three different card slots. Index=0 indicated that the slot was empty and has not been filled yet. For instance, an index composition 240 depicts that the 2nd sentence card was put on the 1st slot and the 4th sentence card was put into the 2nd slot. We call such composition as a state.

The screen also shows different colors. Grey represents that all slots are yet to be filled, black means all slots are filled, but the diagnosis has not been done yet, red shows that the diagnosis has been made but there exists an error of the wrong answer, and green represents the diagnosis also has been made but with the correct answer. This visualization aims to give the teachers a viewing about the steps sequence done by the learners to form a problem along with its pattern on every single action, while the color aims to show the different actions of the learners.

4.2 Activity Tree

Fig. 7 depicts the learning path that is visualized into a tree structure. This visualization represents the learner’s learning process by using states (how they think from a state to another state). Each learner potentially has a very different learning approach. A
teacher can use this data to understand the learner’s tendency on the composition of the sentence structure, identifying the mistakes, the avoided arrangements, and the structure of how the learner finishes such problems. The information on this part can be used by teachers to create a proper feedback based on each student’s performance.

Fig. 6. Visualization of activity sequences.

Fig. 7. Visualization of learner’s learning trajectory.
This screen is also equipped with data filter facility that can narrow-sort learner’s, tasks, and specific trials. In term of the graphical aspect, the orientation can be changed into either landscape or portrait. The data filter also shows information regarding the sentence cards available. Sentence cards demonstrated on orange color represents the correct answer.

At the bottom of the screen, the visualization is represented into tree format. The structure begins with State 000 as its root. The next states are continued by each of the learner’s action. The formulation the structure is continued based on the sentence chosen and its following sentence, for instances, the flow may be depicted as 000 \( \rightarrow \) 001 \( \rightarrow \) 012 \( \rightarrow \) 124. The structure is shown in Fig. 7. Yellow circle represents that the state is formed by the learner, while the green line depicts their changes of actions from one state to another. The bigger the circle means the more that particular state is re-arranged. The thicker the green line shows a high frequency of such part of the track passed by.

5 Discussions

The results shown through visual representations of data indicate that at Level 1 to Level 4, the arrangement made by students does not indicate a serious problem for them. They are able to well-arranged sentence cards to answer the given problems. Although some students make arrangements with quite a lot of steps, in general, it can be solved quite well. Visual representations of sequences of steps for Level 1 to Level 4 are shown in Fig. 8(a)-(d), respectively.

![Fig. 8. Visual representations of sequences of steps for Level 1 to Level 4.](image-url)
In contrast with Level 1 to Level 4, Level 5 tends to be very challenging. Some compositions of the same sentence card arrangement are done repeatedly; in fact, some students often try almost all combinations of sentence card arrangements. Therefore, further investigation will be focused on Level 5. Visual representations of sequences of steps for Level 5 can be seen in Fig. 9. It happens at Level 5 because of the level being a reverse thinking problem. Assignments in the level have a structure where the unknown number in the formula is hidden, and it is one of the operands, instead of the result. For instance, make a word problem about "How many are there overall" that can be solved by "8 - 3". The learners should think and rearrange the formula so that the formula becomes "3 + ? = 8" or "? + 3 = 8".

![List of Sequence Actions](image)

Fig. 9. Visual representations of sequences of steps for Level 5.

According to the result of investigations, learners have different learning patterns. We classify them into four categories, they are smart patterns, adventure patterns, peer patterns, and cyclic patterns.

1. The smart pattern is a pattern where learners quickly understand the concept so that the steps taken to achieve the correct answer are few (see Fig. 10).
Learners in this pattern will quickly realize the mistake and immediately find the correct answer.

![Fig. 10. Visual representations of smart patterns.](image)

2. In contrast with the smart pattern, the adventure pattern implies that learners try almost all possible composition of the sentence card arrangement. It can be seen that almost all the compositions tried to be compiled, but the trials produced wrong answers (see Fig. 11). After almost all compositions have been explored, in the end, the learners with this pattern finally get the correct answer.

3. The next pattern is a peer pattern. This pattern is identified from the existence of two compositions that are always a special concern of learners. The two compositions of the sentence card arrangement are repeated many times so that there seems to be a significant difference in number with the other compositions (see Fig. 12).

![Fig. 11. Visual representations of adventure patterns.](image)
4. The last pattern is a cyclic pattern. The cyclic pattern is a pattern in which learners are trapped in a number of certain compositions. Learners feel that the composition has done will deliver to the correct answer, however, the learners always repeat several compositions of the same sentence card arrangement repeatedly (see Fig. 13).

![Visual representations of peer patterns.](image1)

![Visual representations of cyclic patterns.](image2)

**Fig. 12.** Visual representations of peer patterns.

**Fig. 13.** Visual representations of cyclic patterns.

### 6 Conclusions and Future Works

In this paper, the development of a web-based application intended to give a proper insight for teachers on the class as well as researchers on understanding the learners thinking process. It visualizes the activity pattern of the learner on interactive arithmetic learning application using a problem-posing approach. This research
concludes that the system depicted in this paper is able to represent the sequence and pattern of learner’s actions into a meaningful visualization. Such visualization shows proper information regarding the learning condition and learning process of the learners. The visualization also aims to be one of the considerations to give appropriate feedback towards a specific kind of learners based on their learning patterns.

Future works of this research may include the visualization of activity pattern from the entire class perspective instead of individually, visualization of the learner’s trap states based on the misunderstanding of the problem concept, and visualization of track changes in multiple states. Furthermore, the implementation of data mining techniques such as pattern mining may unveil more information regarding the activity pattern of a learner in its learning activities.

Acknowledgment

This work was supported by DIPA FILKOM UB Grant Number 1999/UN10.36/PG/2018.

References


