Development of an Analyzing System for Student's Learning Characteristics by Visualization of Learning History

Takeshi Morishita
Institute of Education
Shinshu University
Japan
morisita@shinshu-u.ac.jp

Takashi Yokoyama
Graduate School of Science and Technology
Shinshu University
Japan
16w2077k@shinshu-u.ac.jp

Masaaki Niimura
Institute of Engineering
Shinshu University
Japan
niimura@cs.shinshu-u.ac.jp

Hisayoshi Kunimune
Faculty of Information and Computer Science
Chiba Institute of Technology
Japan
kunimune@net.it-chiba.ac.jp

Yoshinori Higashibara
Institute of Education
Shinshu University
Japan
higashi@shinshu-u.ac.jp

Abstract: We have developed a system for visualizing students' learning characteristics within a small learning group based on the learning data accumulated in the individual learning support system. In this paper, we developed a program with both a server side and a client side in order to implement it as a Web application. This makes it possible for the teachers to display a list of learning histories for all students in the class, analyze and understand the learning characteristics of any student in the class.

Introduction

The Ministry of Economy, Trade and Industry (METI) "New Industrial Structure Vision – Interim Arrangement" indicates that "core technology in the fourth Industrial Revolution is common basic technology for innovation in all industries and it is possible to completely fulfill new needs by linking technological innovation and business models in various fields." It is suggested that, in the education field, particularly, Adaptive Learning is enhanced to provide learning materials according to proficiency level through use of AI (Artificial Intelligence) and collaboration with school education is promoted. Adaptive Learning adjusts and provides learning content and learning levels according to each student's progression. Skilled educators are doing this both consciously and subconsciously; however, by utilizing ICT (Information and Communication Technology), everyone will be able to optimize learning content and learning levels. Adaptive Learning records, analyzes, and reflects each student's learning progression to separate learning contents, optimizing and presenting them. It also allows to match students who are having difficulties at similar places, leading them toward a solution through mutual and group learning. With the advancement of Adaptive Learning, Article 30, Paragraph 2 of the School Education Act "Acquisition of Foundational, Basic Knowledge and Skills," is effectively implemented, and is expected to make learning possible according to individual levels, including each of the student's proficiency, learning difficulties, and their field of strengths.

Furthermore, in Japan, the introduction of CBT (Computer Based Testing) is being considered for university entrance examination reform. With CBT, large amounts of response data can be collected and accumulated for each test item. Through big data analysis including life logs, efficient and effective learning according to students' proficiency levels and environments may be provided. Specifically, Baker & Yacef (2009) and Siemens *et al.* (2011) have proposed methods and techniques using learning analysis and various higher education institutions have report cases where academic grades and withdrawal rates have improved. However, "developing a comprehensive data set analysis method for learning environment optimization" and "focusing on the learner's viewpoint" are challenges and development of a framework with learner-focused characteristics and data collaboration is required.

On the other hand, Japan's K-12 learning group has less than 100 people per grade, and it is difficult to collect diverse data from the learning group and extract rules for individual students. By standardizing nationwide data gathered in a national academic achievement/learning status survey, it is possible to extract rules for individual students and learning groups. However, factors influence students' learning environments such as regionality and sociality, showing difference by prefecture, municipality, and school. Since it is extremely difficult to extract rules for individual student's learning activities by standardized indicators from nationwide survey results, it is necessary to realize a development of a method where "a cycle is generated where even a small amount of data can be visualized to create a model which can be adapted to other contexts" (Yamakawa 2015).

Research Purpose

In response to this problem, we have conducted quantitative and qualitative analysis from small amounts of data and clarified individual learning characteristics and group characteristics (Morishita & Higashibara 2016). Based on these findings, we believed it was possible to visualize historical learning data from a small amount of data and to develop a system for understanding the rules of these students. Therefore, this study is aimed to develop a system for visualizing students' learning characteristics within a small learning group based on the learning data accumulated in the individual learning support system.

System Development Summary

In this study, we developed a program with both a server side and a client side in order to implement it as a Web application. The server-side program has the following three roles:

1. Communication with individual learning support systems, collection of learning history
2. Analysis of received learning history
3. Providing client-specified data

This paper uses learning data accumulated through the existing individual learning support system and describes the role of providing the client-specified data.

Server-Side Program

Python 3 and Flask for framework are used for implementing the server side. Python is used for analysis of received learning history because it provides the potential for analysis through machine learning in the future. Python has many libraries and knowledge required for machine learning and it was considered to have less technical obstacles than other programming languages.

"Django," "Bottle," and "Flask" were candidates when selecting Web application framework. After examining the functions and code amounts required on the server side, we decided to use Flask. Django is the most famous within the Python framework and there seems to be a lot of related information. However, for server-side processing, it is simple process where request from the client is issued by SQL (Structured Query Language) to the database, the results returned to the client with JSON (JavaScript Object Notation). Therefore, we thought django, with its many functions, was too specific and we decided not to use it. Bottle and Flask are high profile for lightweight framework. Since differences in functions are not recognized, we focused on the numbers for Watch, Star, and Fork in Github's repository and compared development activities. As of August 10, 2017, Bottle had Watch: 291, Star: 4706, and Fork: 949. Flask, on the other hand, had Watch: 1777, Star: 28903, and Fork: 9118. Therefore, we decided to use flask because future development of Flask is expected.

REST (REpresentational State Transfer) API was implemented to provide data specified by the client. At this time, we receive data read requests from clients; however, we decided not to receive generated, updated, and deleted requests. Provided data was "class information," "student information," "course information," and "learning history." For each set of data, we provide the information in Table 1 and we made it possible to determine the information conditions acquired through URI (Uniform Resource Identifier).

URI Example:

- When acquiring optional lesson information: http://localhost/material/[teaching material code]
- When acquiring all student information for ABC class: http:// localhost/student?class=ABC

Table 1: Major Configuration Data Set

Category	Data
Class	Class Name
Learner	Class Name, Login ID, Learner Name
Unit	Unit Code, Unit Name
Lesson	Class Name, Unit Code, Teaching Material Code, Difficulty Level
Learning History	Class Name, Login ID, Teaching Material Code, Learning Time, Action Count, Current Screen, Current Screen Group, Line, Next Screen, Next Screen Group, Property, Time (Sec.), Errata, Question Number, Answer Field Number, Goal A, Goal B, Answer

Client-Side Program

The client-side program has the following two roles:

- Display a line graph of a student's learning history
- Display student activity content at specific learning points

The JavaScript D3.js library was used for building the client-side program. Since D3.js is good for SVG (Scalable Vector Graphics) operation, we thought it was suitable for this study. As with the server-side program, we considered the use of framework; however, since there were few page transitions and limited functions required on the server side, we determined that the use of framework is unsuitable.

Figure 1 is an image of the client-side program. The graph's horizontal axis shows time and the vertical axis shows the teaching materials code and learning progress. When the teacher specifies a class, the client-side program acquires and displays student information list of the relevant class from the server. Then, when multiple students are selected randomly, the learning history of such students will be displayed on the graph. It is also possible for the teachers to display a list of learning histories for all students in the class. This makes it possible for the teacher to analyze and understand the learning characteristics of any student in the class. Furthermore, if you hover the mouse over an optional student's learning point in the class, you can confirm the details of the student's learning content.

Conclusion

This paper explained this development from the client side, which is the focus of the system operation, within a system that visualizes student learning characteristics within a small learning group. This makes it possible for the teachers to display a list of learning histories for all students in the class, analyze and understand the learning characteristics of any student in the class. On the other hand, it is possible to automate the analysis of students' learning characteristics. However, in order to grow in teacher's ability to understand students, we intentionally decided that teacher analyzed students' learning characteristics. Future challenges include testing and evaluating the developed system within real school education.

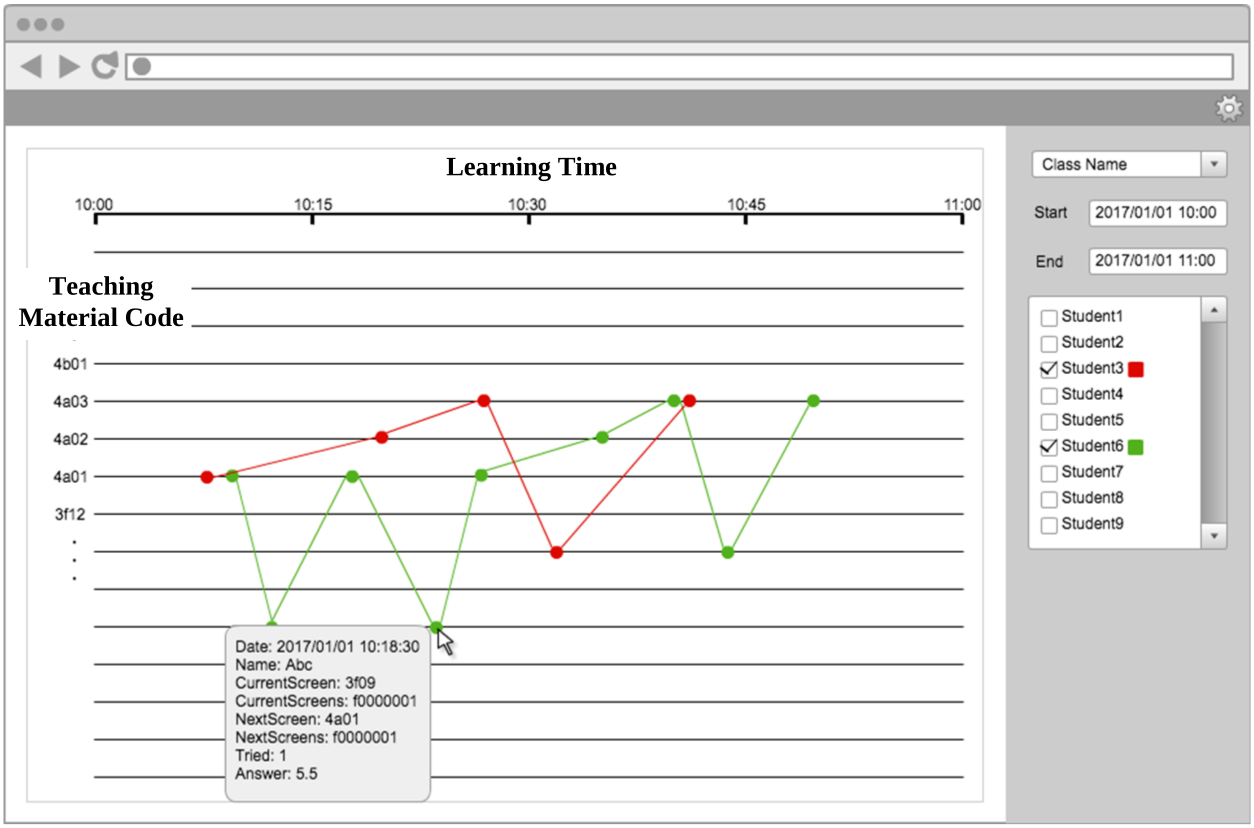

Figure 1: An Image of the Client-Side Program

Acknowledgements

We would like to thank TEK Software Co., Ltd. – http://tek.jp/. This study was supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grant-in-Aid for Young Scientists (A) JP17H04707, Grant-in-Aid for Scientific Research (B) JP26282050 and Grant-in-Aid for Scientific Research (C) JP 15K01023.

References

R. Baker and K. Yacef (2009). The State of Educational Data Mining in 2009: A Review and Future Visions. *Journal of Educational Data Mining*, Vol. 1, No. 1 (pp.3-16).

T. Morishita and Y. Higashibara (2016). Developing a System to Report Levels of Understanding from Educational Data. *In Proceedings of Global Learn-Global Conference on Learning and Technology* (pp. 310-315). Limerick, Ireland: Association for the Advancement of Computing in Education (AACE).

G. Siemens, D. Gasevic, C. Haythornthwaite, S. Dawson, S. B. Shum, R. Ferguson , E. Duval, K. Verbert and R. S. J. d. Baker (2011). Open Learning Analytics: an integrated & modularized platform. http://www.elearnspace.org/blog/wp-content/uploads/2016/02/ProposalLearningAnalyticsModel_SoLAR.pdf (accessed 2017.08.22)

O. Yamakawa (2015). Possibility of Learning Analytics beyond the Borders of Organizations –A Critical Review on the Possiblity. *Computer and Education*, Vol. 38 (pp.55-61): Community for Innovation of Education and learning through Computers and communication network (CIEC). in Japanese.