International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:10, No:11, 2016

Ontology-Navigated Tutoring System for Flipped-Mastery Model

Masao Okabe

***Abstract*—**Nowadays, in Japan, variety of students get into a university and one of the main roles of introductory courses for freshmen is to make such students well prepared for subsequent intermediate courses. For that purpose, the flipped-mastery model is not enough because videos usually used in a flipped classroom is not adaptive and does not fit all freshmen with different academic performances. This paper proposes an ontology-navigated tutoring system called EduGraph. Using EduGraph, students can prepare for and review a class, in a more flexibly personalizable way than by videos. Structuralizing learning materials by its ontology, EduGraph also helps students integrate what they learn as knowledge, and makes learning materials sharable. EduGraph was used for an introductory course for freshmen. This application suggests that EduGraph is effective.

***Keywords*—**Adaptive e-learning, flipped classroom, mastery learning, ontology.

I. Introduction

NOWADAYS, in Japan, more than half of students go on to higher education. That means that academic aptitude and achievement of freshmen vary very much. This is especially true for universities located in other areas than around Tokyo, such as Tohoku Institute of Technology, because there are not many universities and students do not have many choices. One of the main roles of introductory courses for freshmen is, hence, to make all the students with different academic performances well prepared for subsequent intermediate courses.

According to mastery learning [1], it can be done if each student can have as much learning time as he or she needs. But, under an actual educational environment with limited time and resources, it is still practically open.

Recently, a flipped classroom is becoming popular. In a flipped classroom, learning materials are provided prior to a class, usually in videos, which may be prepared by teachers, or, from MOOCs, and students take video lessons before a class, and in a classroom, a teacher spends more time for higher-order thinking skills or students' individual needs. Especially, the flipped-mastery model [2] intends to achieve mastery learning using flipped classroom techniques. However, videos are not adaptive and it is difficult to fit variety of students. In the flipped-mastery model, students are to take responsibility for their own learning. They learn with their own pace in a classroom and outside a classroom.

The flipped-mastery model may be fine for mastery learning on what they learn with their own pace. But it is difficult to apply it to freshmen who are expected to achieve mastery on what intermediate courses pre-require.

A flipped classroom is considered as a kind of blended learning. There are a lot of research outcomes of blended learning, especially with adaptive e-learning systems. Unfortunately, the flipped-mastery model does not seem to use these outcomes. If they use them, it might be more powerful and practical.

Using the research outcomes of adaptive e-learning systems, this paper proposes an ontology-navigated tutoring system called EduGpaph, which supports the flipped-mastery model under the situation that students with different academic performances need to achieve mastery on the same subjects.

The structure of this paper is as follows. Section II reviews related works. Section III presents the proposal. Section IV is a case study. Finally, Section V summarizes the proposal and points out some future works.

II. Related Works

A. Adaptive e-Learning and Ontology

Around 2003, with adaptive e-learning systems, blended learning began to gather attention as a practical solution for mastery learning [3]. Adaptive e-learning systems themselves have a longer history, and appeared soon after Web emerged [4]. Generally speaking, an adaptive e-learning system consists of three main models, a content model, a learner model, and an instruction model, and using these models an adaptive engine make adaptation [5]. Among these models, a content model is essential because even if a learner himself or herself makes adaptation without an adaptive engine, a content model is necessary. A content model or a similar idea was originally proposed by Novak [6], [7] as a concept mapping, and has long been recognized as effective for a learner's cognition. Nowadays, it is called an ontology or a domain ontology. There are many proposals of ontology-driven e-learning or ontology-based educational design. Some examples are [8]–[10]. Unfortunately, these ontologies were developed independently and even worse they might be inconsistent each other. From the point of education, when a learner understands new concepts, it is important to integrate them as knowledge. Ontologies can have an important role for this if they are consistently developed and can be uniformly integrated. But, it is not true.

B. Upper Ontology

Nowadays, ontologies are widely used in many fields, not limited to in education. The situation is similar to in education. Many ontologies are developed, but are probably inconsistent.

Masao Okabe is with the Tohoku Institute of Technology, Sendai-shi, Miyagi-ken, 982-8588 Japan (phone: +81-22-304-5540; fax: +81-22-304-5504; e-mail: okabemasao@ tohtech.ac.jp).

To resolve this situation, several upper ontologies are developed [11]–[15].

An upper ontology is a top-level domain-neutral ontology to be a basis of more specific domain ontologies. If domain ontologies are developed based on the same upper ontology, they are expected to be consistent each other and to be integrated based on the upper ontology. Unfortunately, there is still no consensus on what an upper ontology should be used. But, apparently, BFO [11] is the most widely used upper ontology. It was originally developed as a basis of biomedical ontologies, but now is a basis of many ontologies and aim to promote consistency in description of scientific data and then to promote interoperability of scientific data in electric format among computers [16].

III. EduGraph

A. Overview

One of the reasons why the flipped-mastery model does not use the research outcomes of blended learning with adaptive e-learning systems might be that they are too sophisticated to be implemented to an ordinary classroom.

In practice, many teachers use their own slides. It might be better for students to do self-study outside a classroom, using these slides, instead of a sophisticated e-learning system. In addition, if slides created by different teachers can be shared, they can help both teachers and students.

EduGraph is a fairly simple ontology-navigated tutoring system, which can be used both by a teacher in a classroom and by students themselves outside a classroom, using their own PCs or smartphones. Fig. 1 shows a high-level overview of EduGraph.

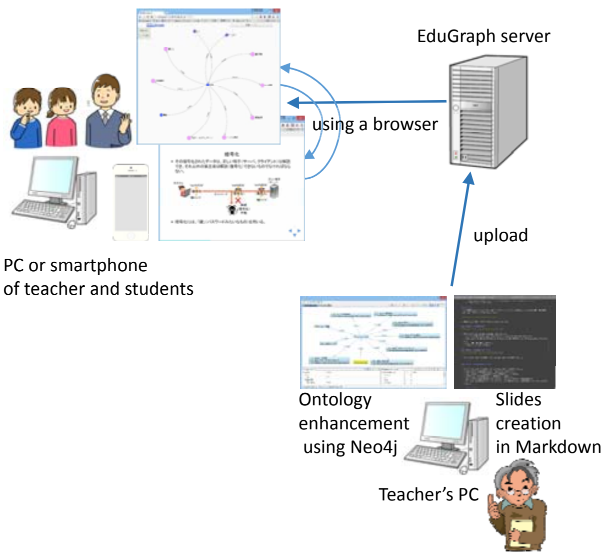


Fig. 1 Overview of EduGraph

EduGraph mainly consists of two parts. One is a slide part and the other is an ontology part. In the slide part, a teacher can easily create slides in Markdown [17], a very simple markup language and upload them on the Web. Both a teacher and students use them via a browser just same as usual slides. In the ontology part, each learning item on slides is classified by EduGraph ontology. EduGraph ontology has two important roles. One is to navigate a student what to learn next, and the other is to integrate all the learning items on different slides. A learning item in EduGraph ontology and a slide that explains it are mutually linked. Navigated by EduGraph ontology, students easily access a slide that he or she need to learn next, independent from actual slide sequences.

EduGraph ontology is a kind of content model. EduGraph has no learner model nor no instruction model, because EduGraph does not intend to make adaptation automatically. The ontology simply suggests what should be learned next from several points of view. Adaptation is left to a learner's own interest and will. EduGraph does not do reasoning for adaptation, and therefore, its structure is simple and can be represented fully by a graph. A teacher can expand EduGraph ontology easily using Neo4j [18], a graph database. But, to support integration of various slides, it needs to be based on a well-designed upper ontology.

B. EduGraph Ontology

1. Upper Ontology

This subsection presents an upper ontology of EduGraph ontology, in contrast to BFO. All the descriptions on BFO in this subsection are based on [16], [19].

Originally, EduGraph ontology was planned to be developed based on BFO. But, there are several reasons that BFO is not suitable for an upper ontology of EduGraph ontology.

Firstly, objectives are different. The objectives of BFO are interoperability of scientific data in electric format among computers. For that purpose, BFO needs to be a basis of some kind of reasoning. On the other hand, the objectives of EduGraph ontology are to navigate learners and EduGraph ontology need not be a basis of reasoning because automatic adaptation is out of the scope.

Secondly, users are different. Users of BFO are scientists who may or can have technical background of ontologies. On the other hand, users of EduGraph ontology are learners, more specifically, freshmen of universities, who cannot be expected to have technical background of ontologies.

Thirdly, domains are different. BFO now intends to be a domain-neutral upper ontology. But, BFO seems to be slightly influenced by its origin as a basis of biomedical ontologies. For example, there is a class called "Generically dependent continuant" (see Fig. 2), which is very generic and difficult to understand but that has no subclasses probably because it is not important to biomedical ontologies. Therefore, EduGraph ontology has its own upper ontology. It is similar to BFO, but is somewhat different because of the reasons above. Basic structure of the upper ontology of EduGraph ontology, in contrast to BFO, is as follows;

a. Universals and Particulars

One of the top-level dichotomies is universals and particulars. BFO does not have this top level dichotomy and only categorizes universals. Particulars are out of the scope of

BFO. This is because BFO is to provide universals that can classify particulars for its objectives.

EduGraph ontology does not have this top level dichotomy either and mainly structuralizes universals because usually a learning item is a general thing, that is, a universal. But, sometimes, particulars are important because some examples (particulars) help to understand a general thing (a universal). In that case, EduGraph classifies the particulars under the universal as its instances.

b. Top Dichotomy

Although treatment of particulars is different between BFO and EduGraph ontology, they have the same top dichotomy. That is, continuants and occurents (see Figs. 2 and 3). Continuants are entities that continue or persist through time, and occurents are entities that occur or happen. Just below “Universal,” BFO has two disjoint classes called “Continuant” and “Occurent.” But, freshmen of universities may not be familiar with these words, and instead of them, EduGraph uses more popular words. That is, EduGraph ontology has two disjoint classes called “Mono” and “Koto,” both of which are very popular Japanese words and the former of which means a static thing and the latter of which means a dynamic thing. Hereafter, instead of “mono” and “koto,” the English words “static thing” and “dynamic thing” are used in this paper.

c. Structure of Static Thing

Upper part of class hierarchy of “Continuant” of BFO is shown at Fig. 2. “Continuant” has three direct subclasses called “Independent continuant,” “Generically dependent continuant” and “Specifically dependent continuant.” “Independent continuant” is a class of entities that can exist independently, whether tangible or not. Generally speaking, “Dependent continuant” is a class of entities that can only exist dependent on another entity. BFO distinguishes "Specifically dependent continuant" and “Generically dependent continuant.” “Specifically dependent continuant” is a class of entities that exist dependent on another entity which cannot be changed, and its typical subclass is “Quality” as shown in Fig. 2.

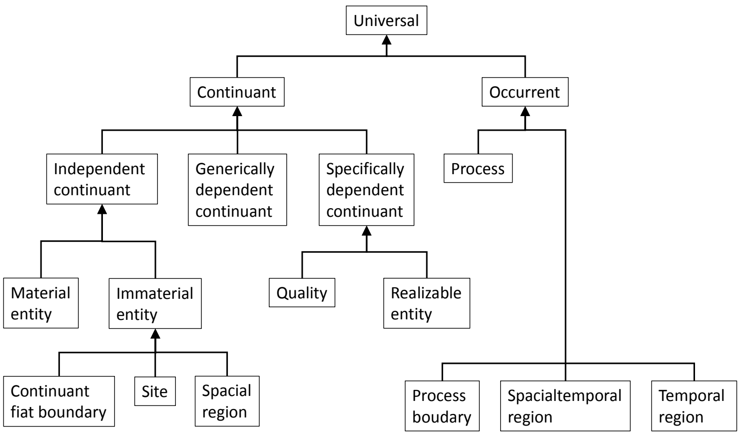


Fig. 2 Upper part of BFO

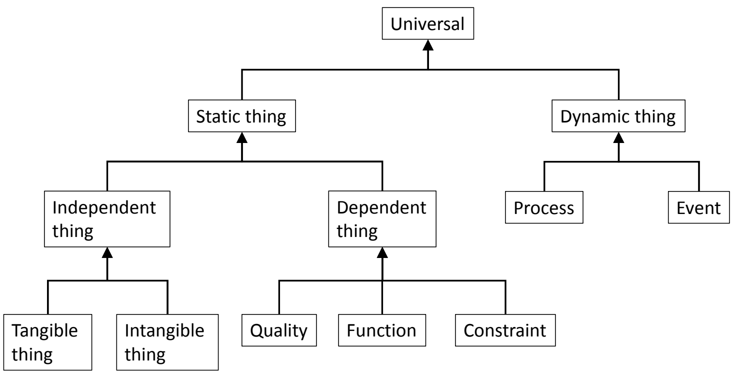


Fig. 3 Upper ontology of EduGraph ontology

“Generically dependent continuant” is a class of entities which can only exist dependent on another entity, which can migrate one entity to another. “Information” is a typical subclass of this, although BFO itself does not have it. An information artifact needs to be stored in some device, say a CD. Then, this information artifact is dependent on this CD. This information artifact can be copied to a USB memory. Then, this very same information artifact is dependent on this USB memory. So, this information artifact has to be depend on a CD or a USB memory, but what this information artifact is dependent on migrates from a CD to a USB memory. So, if the information signals stored in a CD and in a USB memory are distinguished, they are specifically dependent on a CD or a USB memory. If these information signals are recognized as one information artifact semantically, then this information artifact is generically dependent on a CD or a USB memory. Intuitively saying, so-called an abstract thing has many representations, and each representation is a specifically dependent continuant and this abstract thing is a generically dependent continuant. Thus, an abstract thing and its representations are strictly distinguished in BFO. But, from the point of education, it is rarely necessary to distinguish them, because, generally speaking, what should be learned is not a representation but its meaning. Hence, in EduGraph ontology, "Specifically dependent continuant" and "Generically dependent continuant" are merged to just "Dependent thing" and the difference between a representation and its semantics are ignored. Thus, in EduGraph ontology, "Static thing" has two direct subclasses called "Independent thing" and "Dependent thing."

In BFO, "Independent entity" has two direct subclasses called "Material entity" and "Immaterial entity." The interesting thing is that "Immaterial entity" includes only boundary, region and site which is space that can contain material entities and that so-called an abstract thing or a conceptual thing is not an immaterial entity. This is partially because, philosophically, BFO stands on realism and not conceptualism, and partially because BFO classifies so-called an abstract thing or a conceptual thing as a generically dependent continuant.

Boundary, region and site are important in biology, but from the point of education, most of things to be learned are abstract or conceptual, and compared to them, boundary, region and site are not so important. Therefore, in EduGraph ontology, "Independent thing" has two direct subclasses called "Tangible thing" and "Intangible thing," and "Tangible thing" is almost the same as "Material entity," but "Intangible thing" is much broader than "Immaterial entity" in BFO and includes some portion of "Generically dependent continuant" in BFO.

d. Structure of Dynamic Thing

In BFO, "Occurrent" also has "Process boundary," "Spatiotemporal region" and "Temporal region" as its direct subclasses. They are similar to "Immaterial entity" in "Independent continuant" and are not so important from the point of education. In EduGraph ontology, "Dynamic thing" does not have such direct subclasses.

There are several more simplifications and modifications, and in consequence, class hierarchy of the upper ontology of EduGraph ontology is as shown at Fig. 3.

2. Relationship

As a relationship type name to represent the class hierarchy above, EduGraph uses "sub item" (translated from Japanese). EduGraph avoids to use such words as "is_a" or "rdfs: subClassOf" because users of EduGraph ontology may not be familiar with these words. EduGraph has only a few more relationship types and uses very common words for their names so that even users with no background on ontologies can easily use it. EduGraph does not have a relationship type such as "prerequisite" because EduGraph ontology does not intend to specify learning sequence, but intends to give a big picture of what to learn and loosely suggests what to learn next, leaving a room for a learner to follow his or her own interest and will.

The names and meanings of relationship types of EduGraph ontology are summarized at Table I. Very roughly speaking, if a learner does not understand a learning item, he or she should follow a "part" relationship or an "example" relationship. If a learner understand a learning item and wants to know more, he or she should follow a "related item" relationship or the inverse of a "sub item" relationship.

TABLE I
RELATIONSHIP TYPES OF EDUGRAPH ONTOLOGY

Relationship type	Notation	Meaning
sub item	A - sub item -> B	B is a subclass of A.
example	A - example -> B	B is an example (an instance) of A.
part	A - part -> B	B is a part of A.
related item	A - related item -> B	B is related to A.

IV. CASE STUDY

Some introductory course materials for EduGraph were developed at Department of Management and Communication, Tohoku Institute of Technology, and they were actually used in some classes in 2015. This section presents a tentative evaluation of EduGraph based on its application to "Introduction to Information Technology," one of the courses in which EduGraph was actually used.

A. Course Explanation

"Introduction to Information Technology" is an introductory course for freshmen, and covers basics on hardware, software, network, and information security, all of which are necessary to understand the subsequent courses on information technology. Its coverage and level is almost the same as the technology part of "Information Technology Passport Examination," which is a very popular introductory qualifying examination of information technology in Japan. Hence, the contents of this course are fairly standardized and are necessary for subsequent courses on information technology.

In Japan, some high schools provide a similar course and some not. Hence, students' backgrounds for this course differ, but its goal is the same.

B. How EduGraph Was Applied

This course was taught at a classroom equipped with a computer for each student. At the kickoff of the course, all the learning materials structured by EduGraph ontology were provided to all students.

For each class, students were requested to prepare and review using EduGraph before the class and after the class. Students are also requested to submit a minute paper of their self-studies.

In a classroom, the learning materials developed by EduGraph were used usually as ordinal slides, but more time was spent to explain topics on which students raised questions and in which students expressed a special interest at the minute papers.

Out of a classroom, students look at and study them by their own smartphones or PCs, sometimes as ordinary slides and sometimes navigated by EduGraph ontology. Fig. 4 is some screen shots of EduGraph ontology and slides navigated by it.

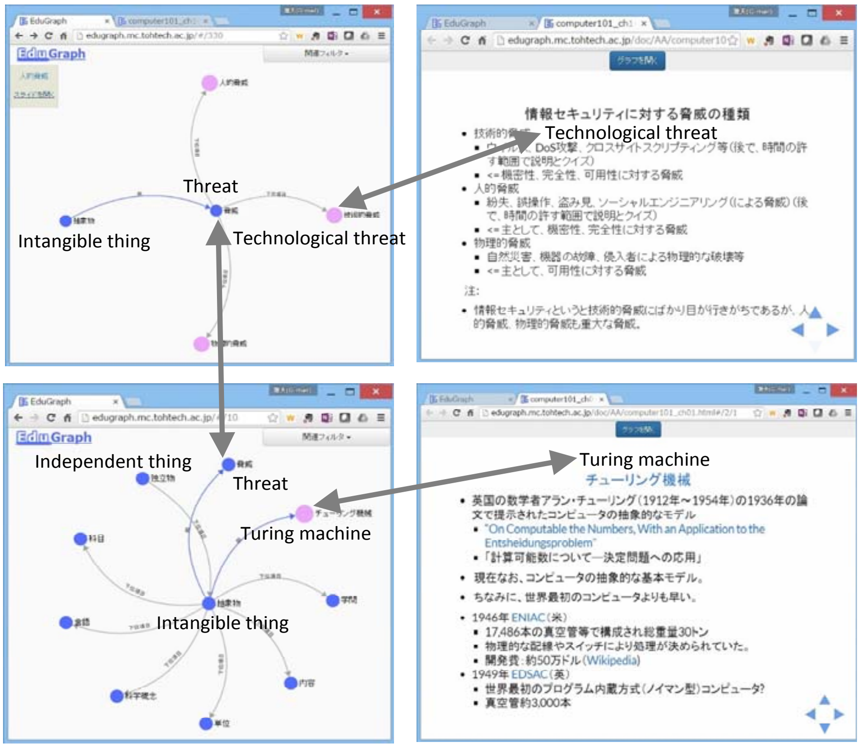


Fig. 4 Screen shots of EduGraph

C. Tentative Evaluation

Firstly, the comparison to the same course in 2014 is presented based on scores of their mid-term exams.

The course materials of 2014 were prepared by PowerPoint, but the contents were almost the same as the ones in 2015. Actually, the course materials of 2015 were converted from the ones of 2014 to the Markdown format and were structured by EduGraph ontology.

The reason why, instead the final exams, the midterm exams were used for the comparison is that both in 2014 and 2015 the midterm exams were scheduled just before the final exams and that the final exams were a little bit unordinary because most of the questions are almost the same as the questions of the midterm exams that many students failed.

The questions of the mid-term exams in 2014 and 2015 are different, but their levels are the same because both are very much similar to the questions of the technology part of "Information Technology Passport Examination" and most of the questions are 4 answer choice question.

A stringent comparison is impossible because students in 2014 and 2015 are different, but some suggestions may be got. Table II shows some statistics. The average score in 2015 is 7 points higher than that in 2014. The standard deviation in 2015 is slightly bigger than that in 2014.

Fig. 5 is the histogram of scores with 10 points band width. It shows that in 2015, compared to in 2014, more students have higher scores.

Fig. 6 is the histogram of scores with 25 points band width, and is coarser, but is more suggestive. Since most of the questions are 4 answer choice questions, scores less than or equal to 25 means that they do not learn at all. Regrettably, about 10% are such students both in 2014 and in 2015. Except them, the ratio of poor students (scores are between 26 and 50) are more than half in 2014, but less than 40% in 2015. The ratio of good students (scores are more than 75) are significantly increased in 2015, compared to in 2014.

TABLE II
STATISTICS ON SCORES OF MID-TERM EXAMS

	2015	2014	Difference (2015-2014)
Number of Students [a]	66	60	6
Mean	51.4	44.4	7.0
Standard Deviation	20.2	18.1	2.1
First Quartile	36.0	31.8	4.3
Median	51.0	39.5	11.5
Third Quartile	67.5	56.3	11.3

[a] Students who did not take a mid-term exam are eliminated.

International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:10, No:11, 2016

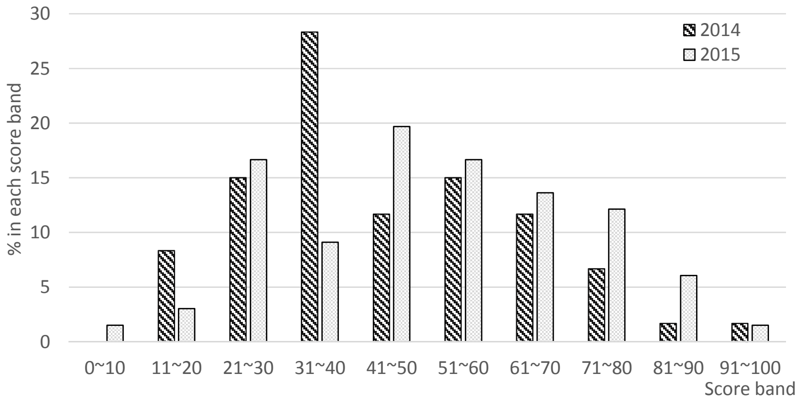


Fig. 5 Mid-term exam score histogram with 10 points band width

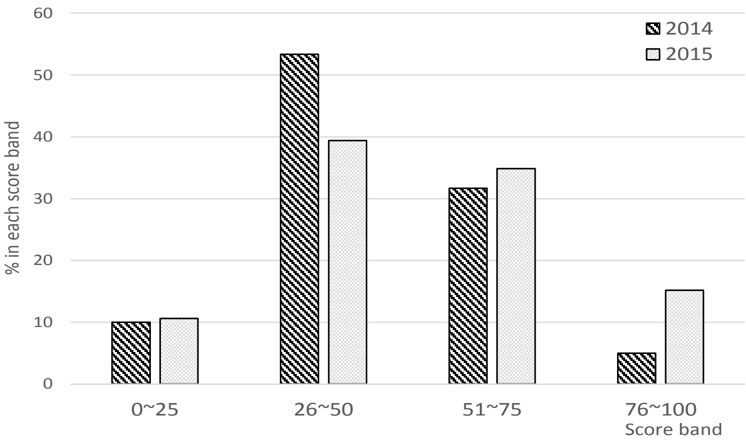


Fig. 6 Mid-term exam score histogram with 25 points band width

Secondly, the relation between score growth and total amount of time students spent studying outside a classroom using EduGraph is analyzed.

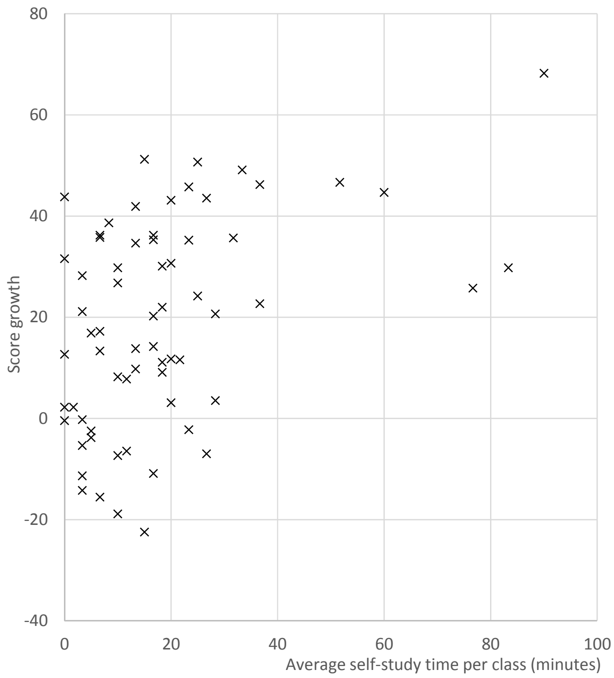
Score growth is simply calculated by subtracting an assessment exam score from a mid-term exam score. An assessment exam here is an exam that was conducted at the first class to know what students knew and what not. The exam consists of 4 answer choice questions, similar to the mid-term exam. The level is the same as the mid-term exam, but the coverage is a little bit broader than that of the mid-term exam. Hence, to calculate score growth, only questions within the coverage of the mid-term exam were used and their score were summed up to 100 if perfect. The reason why a mid-term exam was used rather than a final exam was the same as stated before.

The total amount of time spent studying outside a classroom using EduGraph was calculated from minute papers of self-study submitted before the mid-term exam. On the minute paper, there are 4 choices each for preparation and review. Table III shows how each choice is converted to minutes for summing up. Unfortunately, students do not always submit a minute paper of self-study. Its submission ratio is 50.5%, almost half. If a student did not submit a minute paper of self-study, it was unknown how much time spent on self-study, but it was counted as 0 minute.

TABLE III
CONVERSION FROM A CHOICE TO MINUTES

Choice	Minutes
None	0
Less than 30 minutes	15
More than 30 and less than 60minute	45
More than 1hour	75

Fig. 7 is a scatter plot of score growth versus average self-study time per class. Their correlation coefficient is 0.44 ($p=0.0002 < 0.05$). They have weak but significant positive correlation. Average self-study time per class less than 20 minutes looks little correlated with score growth. Actually, their correlation coefficient is 0.16. This is partially because there are several students who submitted no or only a few minute papers and who had good score growth. If the submission ratio of a minute paper had been better, stronger correlation had been expected.



Note: Students who did not take a assesment or mid-term exam are eliminated.

Fig. 7 Scatter plot of score growth versus average self-study time per class

V. Summary and Future Works

This paper has proposed EduGraph, an ontology-navigated tutoring system for the flipped-mastery model, especially for the situation that students with different academic performances need to achieve mastery on the same subjects.

Similar to preparing slides by PowerPoint, a teacher can prepare learning materials by EduGraph. The learning materials developed by EduGraph can be used both by a teacher in a classroom and by students outside a classroom. Students can use them easily, by their own smartphones or PCs. Using EduGraph, navigated by EduGraph ontology, students can prepare for and review a class, in a more flexibly personalizable way than by videos usually used for a homework of a flipped classroom. In addition to that, since EduGraph ontology is based on a well-designed upper ontology, learning materials structured by EduGraph ontology help students integrate what they learn at different classes and also help teachers share learning materials.

A case study suggests that EduGraph is effective. But, the case study also reveals its limitation. Good students learn better using EduGraph, but students who will not study do not study, even with EduGraph. This is not an issue resolved by EduGraph alone. In addition to EduGraph, it is always necessary to motivate students to learn, although it is not an easy task to motivate all of students. Also, this case study does not show that learning materials structured by EduGraph ontology help students integrate what they learn and teachers share learning materials. To confirm this, much more learning materials need to be developed by EduGraph and structured by EduGraph ontology. Learning materials for EduGraph will be continued to be developed, and EduGraph ontology will be continued to be enhanced to make them well-structured.

Acknowledgment

A part of this research is supported by the Japan Society for the Promotion of Science KAKENHI Grant Number 15K12423.

References

[1] B. S. Bloom, “Learning for Mastery. Instruction,” UCLA CSEIP Evaluation comment. Vol1(2), 1968.
[2] J. Bergmann, A. Sams, *Flip Your Classrooms.* Virginia, U.S.A: International Society for Technology in Education, 2012, ch.5, 6, 7.
[3] H. Singh, “Building Effective Blended Program,” Issue of Educational Technology, vol.43, no. 6, pp.51-54, December 2003.
[4] P. Brusilovsky, J. Ekulund, E. Schwarz, “Web-based Education for all: A Tool for Development Adaptive Courseware,” in Proc. 7th International World Wide Web Conference, 1998, pp.291-300.
[5] V. Shute, B. Towle, “Adaptive E-Learning,” Educational Psychologist, vol. 38, no.2 pp.1052-114, 2003.
[6] J. D. Novak, “Applying psychology and philosophy to the improvement of laboratory teaching,” The American Biology Teacher, vol. 48, no. 8, pp.466-474, 1979.
[7] J. D. Novak, “Concept mapping: a useful tool for science education,” Journal of Reserch in Science Teaching, vol. 27, no. 10 pp.937-949, 1990.
[8] C. Snae, M. Brueckner, “Ontology-driven e-learning system based on roles and activities for Thai learning environment,” Interdisciplinary Journal of Knowledge and Learning Objects, vol. 3, pp. 1-17, 2007.
[9] M. Gaeta, F. Orciuoli, and P. Ritrovate, “Advanced ontology management system for personalized e-learning,” Knowledge-Based System, vol. 22, pp.292-301, 2009.
[10] E. L. Baker “Ontology-based educational design: seeing is believing,” Resource Paper No.13, National Center for Research, Graduate School of Education, University of California, Los Angeles, CA, 2012.
[11] BFO Basic Formal Ontology, at http://ifomis.uni-saarland.de/bfo/ (Accessed 2016-02-10).
[12] DOLCE: a Descriptive Ontology for Linguistic and Cognitive Engineering, at http://www.loa.istc.cnr.it/old/DOLCE.html (Accessed 2016-02-15).
[13] Suggested Upper Merged Ontology (SUMO) at http://www.adampease.org/OP/ (Accessed 2016-03-05).
[14] J.F. Sowa, Top-Level Categories at http://www.jfsowa.com/ontology/toplevel.htm (Accessed 2016-01-21).
[15] R. Mizoguchi, “YAMATO: Yet Another More Advanced Top-level Ontology,” available at http://download.hozo.jp/onto_library/YAMATO101216.pdf (Accessed 2015-11-10).
[16] R. Apr, B. Smith, A. D. Spear, Building Ontologies with Basic Formal Ontology, The MIT Press, MA, 2015.
[17] Markdown Here at http://markdown-here.com/ (Accessed 2016-04-10).
[18] Neo4j at http://neo4j.com/ (Accessed 2016-04-10).
[19] B. Smith, Basic Formal Ontology 2.0 Specifications and User’s Guide, available at https://github.com/BFO-ontology/BFO/raw/master/docs/bfo2-reference/BFO2-Reference.pdf (Accessed 2015-12-10).