Course Content Analysis: An Initiative Step toward Learning Object Recommendation Systems for MOOC Learners

Yiling Dai
Graduate School of Informatics
Kyoto University
Yoshida-Honmachi, Sakyo-ku
Kyoto, Japan
daiyiling@db.soc.i.kyoto-u.ac.jp

Yasuhito Asano
Graduate School of Informatics
Kyoto University
Yoshida-Honmachi, Sakyo-ku
Kyoto, Japan
asano@i.kyoto-u.ac.jp

Masatoshi Yoshikawa
Graduate School of Informatics
Kyoto University
Yoshida-Honmachi, Sakyo-ku
Kyoto, Japan
yoshikawa@i.kyoto-u.ac.jp

ABSTRACT

With the accelerating development of open education, low-cost online learning resources, such as Massive Open Online Courses (MOOCs), are reaching a wide audience around the world. However, when faced with these appealing but overwhelming learning resources, learners are prone making rash learning decisions, which may be either excessive or insufficient to their learning capacities. To avoid the mismatch between learners and learning objects, we propose a supporting system that recommends a personalized path of learning objects for a given learner. In realizing this system, a domain knowledge structure is necessary to connect learners' information and learning objects. As an initiative step, we employ the Labeled Latent Dirichlet Allocation method to predict how the content of a course is distributed over different categories in the domain. We conduct experiments by utilizing course syllabi as course content, and curriculum guidelines as domain knowledge. The predicting performance is improved when involving external texts related to the concerned domain knowledge unit.

1. INTRODUCTION

Nowadays, pedagogically condensed free online resources are playing an increasingly more important role of facilitating self-learning. Among those resources, Massive Open Online Courses (MOOCs) are engendering a revolutionary change in higher education by distributing digital versions of university courses to everyone at a relatively low cost. Courses about Computer Science on Edx (one of the largest MOOC platforms), reached over 600,000 listeners during the period from 2012 autumn to 2014 summer [6], which hardly ever occurs on real campuses. However, compared with their popularity among audience, the low completion rate of courses (e.g, 7% of the MOOCs on Edx mentioned above) begs the question—how many learners have truly benefited from receiving MOOCs? It appears that MOOCs have a way to go to achieve its original goal of making education accessible to everyone.

Rather than not being able to receive traditional education, many users utilize MOOCs out of pure curiosity toward subjects, or to complement their academic lives or career development [2]. In addition, the occupations of MOOC users are diverse, from students, writers, and engineers to housewives [2]. This type of utilization of MOOCs sets a higher requirement in terms of learner's self-motivation and self-regulation. Consequently, many users have reflected that they did not have sufficient spare time to catch on to the process of MOOCs, or simply became stuck on the overwhelming learning contents [2].

An intuitive question concerning that how we can help to maintain this precious enthusiasm of refreshing one's knowledge, motives this paper. We hold the view that finding the "just right" learning objects for respective individuals paves the way toward a successful learning experience. This belief is also in agreement with the opinion of [4], which underlines the importance of personalization, especially in the context of online learning. Specifically, "just right" means that the learning objects fit both the learning objective and learning ability of a given learner. In the context of self-learning, where more flexibility is given to a learner for him to decide what to learn, the adaptation to learning objectives deserve greater investigation than before. Concerning the method used to accomplish personalization in learning, previous studies have shown a trend of utilizing expert manpower or learner performance data to extract internal relationships among knowledge itself and external relationships between knowledge and learner mastery, which may not work when promoting personalized learning on a massive scale.

In this paper, we propose the idea of a novel supporting system that automatically recommends an appropriate set of learning objects with cues of learning priority to a given learner. This system is expected to outperform existing adaptive learning systems on addressing heterogeneous course materials automatically and on adapting learning objects to learners before they start to learn. As an initiative task, a course content analysis is conducted to crystallize the realization of the supporting system. We employ the Labeled Latent Dirichlet Allocation method to predict how the content of a course is distributed over different domain knowledge categories. Course syllabus texts are utilized as course content, and the knowledge listed in curriculum guidelines are utilized as domain knowledge. To improve the accuracy of predictions, we extend the content of the curriculum guideline by integrating external texts retrieved from search engines.

The remainder of this paper is structured as follows: Section 2 summarizes related work with regard to personalized learning and knowledge representation. In section 3, an illustration and the framework of the supporting system are sketched. Then, we present the results and observations of a course content analysis. Finally, we discuss on future work.

2. RELATED WORK

2.1 Personalized learning

What we call personalized learning is named differently in previous studies, e.g., adaptive learning/education, individualized learning/education, and intelligent tutoring systems; however, they all share the main concern of adapting learning materials to individual learners. In this paper, we adopt the phrase "personalized learning" to capture all these related studies and use "personalize", "individualize", "adapt" interchangeably.

Personalized learning is described as "learning tailored to the specific requirements and preferences of the individual" in [11]. Although not forming a fixed definition of personalized learning, many studies attempt to adapt learning to specific learners. [4] demonstrated a hypermedia textbook that can provide direct guidance and adaptive navigation support to learners. Similarly, [15] developed a topic-based adaptive learning system that directs the learner to the appropriate learning object by providing navigational cues. Moreover, [16] broadened the adaptation from a single source of personalization information to learning achievements and learning styles at one time. [8] presented an e-learning system that recommends learning items by detecting frequent learning sequences and similar learners. [9] proposed another approach of generating adaptive course content using concept filters.

A shared architecture of a personalized learning system that can be observed consists of three parts: Domain model, Learner model and Adaptation model. The domain model constructs all the knowledge units of learning materials in a common space, and its complexity varies based on the application contexts. The learner model is a projection of a learner's learning state (i.e., mastery level of knowledge, learning objective, and learning style) onto the structure of knowledge that is defined in the domain model. The adaptation model functions as a recommend of the next learning target basing on the updated learner state. This adaptation in learning environments occurs at different levels. [11] categorized this adaptation as follows: Adaptive Interaction, which occurs during the interactions between learners and the system; Adaptive Course Delivery, which intends to tailor learning materials to a given learner; Content Discovery and Assembly, which involve the collecting of learning materials from potential sources or repositories; Adaptive Collaboration Support, which supports communication in the learning process.

In the context of self-learning, "why I want to learn", "what I want to learn", "what outcomes I am expecting", things usually being told to the learner by the curriculum, must be determined by the learner himself. As a result, we consider that the information-seeking phase before starting to learn becomes a key to a successful learning experience. We provide a learning object recommendation system that the learners can resort to when they are faced with overwhelming learning resources. Compared with a branch of studies [10, 1, 19] that implement the adaptation by redirecting the learner to an optimal learning path using tracked learner performance, our approach focuses on a more macro level of adaptation, which occurs beforehand and addresses the learning object with a larger granularity (i.e., a lecture). According to [11]'s categorization of adaptation, our system stands in an overlapping area of Adaptive Course Delivery and Content Discovery and Assembly, thereby distinguishing itself from other adaptive learning/tutoring systems.

2.2 Automatic domain representation

The construction of domain knowledge is a key step in accommodating a personalized learning system. However, previous studies [4, 15, 16, 8, 9, 10, 1, 19] show a substantial reliance on expert efforts, whose systems require the instructors to define strictly structured course materials for the concerned system. This is so time-consuming and platform dependent that it is unsuitable when addressing a large amount of distributed learning materials. An automatic and interoperable knowledge representation and assemble are thus desired.

In the context of learning, knowledge representation refers to the process of editing knowledge in a more visually sound and retrievable manner based on its hierarchical or dependent relationships. Previous studies relating to this concept can be divided into two types according to their approaches, and we name them prior approaches and post approaches. A prior approach means extracting the relationships between knowledge units based on the structure defined by the instructor. For example, [3] utilized the content and structure of a textbook to extract the relationships between concepts based on their co-occurrence conditions. [5] exploited the extraction of prerequisite relationships of learning objects by conducting semantic analysis on Wikipedia articles. Regarding the post approach, in which the structure of knowledge is modified by the learner reactions on these learning objects, [17] and [18] attempted to detect prerequisite relationships between knowledge units by utilizing a considerable amount of learner achievement data. Their studies are based on the rationale that knowledge units that are statistically "always" mistaken by the learners should be learned before the ones that are not so.

In this paper, we emphasize the preprocess of learning (i.e., seeking information and making a learning plan), which occurs before a substantial amount of learner performance data

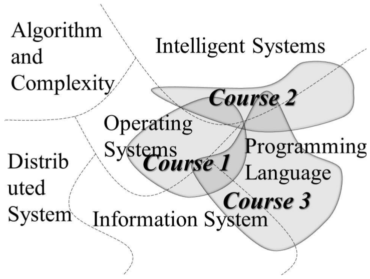

Figure 1: Illustration of our supporting system—the course map

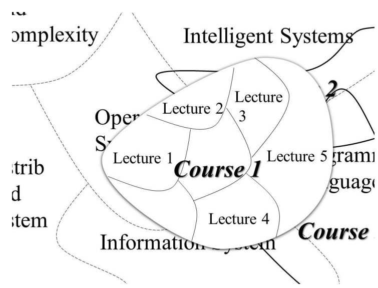

Figure 2: Illustration of our supporting system—the detailed course information

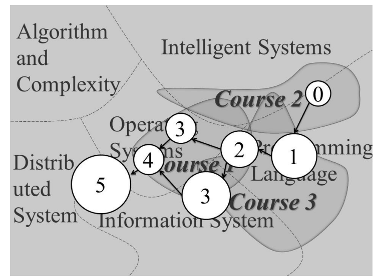

Figure 3: Illustration of our supporting system—a learning path

are available. Thus, our research falls into the category of prior approaches. Previous studies [3, 5] have employed various Natural Language Processing techniques to extract relationships between knowledge units. However, the results remain modest in addressing heterogeneous learning materials at scale; a proliferation of this stream of research is needed.

3. OUR SUPPORTING SYSTEM

As discussed in the previous section, in the context of self-learning, support for a learner determining what to learn and how to learn is sensible. Except for a learner's learning ability, which has received a fair discussion in previous research, we consider the estimation of the learner's learning objective. Regarding the level of personalization in this learning environment, we highlight the phase of assembling learning materials from distributed learning resources. As a consequence, we suppose that learners will benefit from our system before they enter the real learning process when offered a tailored path of learning objects that fits their learning needs and ability.

3.1 An illustration of the system

To explain our supporting system more vividly, we present an illustration of a final usage of the system. The target user of our system will not be constrained to a specific group of learners; however, the learners who will benefit the most from our system are those who are planning to challenge some unfamiliar subject. Then, we can imagine a virtual learner, a college student majoring in social science, who is wondering how data mining techniques will assist in analyzing his collected data.

First, he may simply input a keyword "data mining". Instead of returning a ranked list of relevant courses, which is normal in existing MOOC search engines, our system will answer the query dynamically by starting with a map of relevant courses to that query. As shown in Figure 1, the shapes circled using a dotted line with titles (e.g., "Intelligent Systems") on them refer to the predefined structure of the domain knowledge. In addition, the shape circled using a solid line represent a course that contains the knowledge in that place.

Then, the learner responds to the first reply differently. He may want to obtain details of some highly similar courses or seek a more holistic view of this domain to determine what these courses mean to his learning task. If the learner chooses to zoom in to course 1, then he will obtain a detailed view of the content of course 1. As shown in Figure 2, the topics covered in course 1 will be shown in the unit of a lecture.

We suppose that the learner will not be satisfied until he can make a confident decision on what and how to learn. Therefore, he will continue interacting with our system, during which time his learning characteristics will be recorded. Finally, the recorded learner information will be used to recommend a tailored learning path for the learner (see Figure 3). The path consists of a set of learning objects that are chained according to the dependent relationships between the knowledge they cover. For well-prepared learners, the path will exclude materials he already knows and will cover a narrowed down knowledge set in the depth. For novice learners, in this case, the path will cover a wider range of knowledge and will start from the very simple knowledge units.

3.2 The architecture of the system

To realize the system illustrated above, the architecture is threefold—domain model, learner model, and personalization model. The domain model conducts the task of locating the learning objects of courses in the knowledge structure of the domain. The learner model tracks learner information about his learning objective, background knowledge, and learning preferences according to the knowledge structure. The personalization model specifies the appropriate learning objects based on predefined criteria. Among them, the construction of domain knowledge and the mapping of course content determine how to estimate learner information and what learning objects to recommend. Thus, it is reasonable

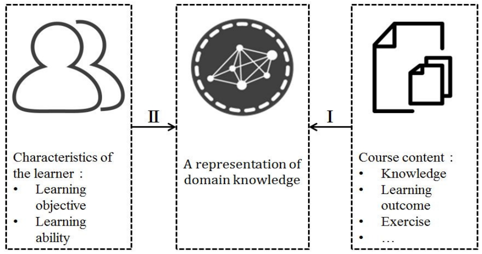

Figure 4: The architecture of proposed system

to exploit the domain model as a primary task. The following part of this paper describes a course content analysis and discusses its potential for equipping the domain model.

4. COURSE CONTENT ANALYSIS

4.1 Overview

As a primary task for matching course contents to a domain knowledge base, we extracted knowledge coverage of a given course by projecting its syllabus text onto a curricular guideline in the domain. A syllabus functions as a summary of the course content, which makes it suitable for our method. In addition, a curricular guideline generally contains important topics in the domain, which can be utilized as a reference of the domain knowledge. Specifically, we utilized the curriculum guideline *Computer Science Curricula 2013* (CS2013) [14] published by IEEE-CS and ACM, which attempts to provide instructional cues of knowledge that should be included in an undergraduate program. In CS2013, both classic and frontier topics in this domain are described in *Body of Knowledge* (BoK). BoK is compiled in a hierarchical structure wherein the smallest granularity of knowledge is a *topic*, and each *topic* belongs to a *Knowledge Unit* (KU), and each KU further belongs to a *Knowledge Area* (KA). In total, 18 KAs and 163 KUs are formed to categorize knowledge in the domain of Computer Science. A simplified example of KA-KU-Topic knowledge structure in CS2013 is shown in Table 1.

This semi-structured BoK has been used to analyze the curricula of different educational institutions [7, 13]. In an attempt to obtain an overall picture of Informatics programs in Japan, [7] conducted a judgement of knowledge coverage on syllabi by referring to curriculum guidelines. [13] employed a supervised Latent Dirichlet Allocation (LDA) method to extract KA coverages of a course using the text of its syllabus. From the above studies, it is reasonable to use curriculum guidelines as a knowledge base to form predictions of course knowledge coverage in an automated manner. However, it is not sufficient to recommend learning objects when solely using the knowledge coverage of a course at the level of KA. Therefore, we attempt to extract knowledge coverage of a course at a further fragmented level—KU in this case.

We adopt the topic model, Labeled Latent Dirichlet Allocation (Labeled LDA) to extract the knowledge coverage. Labeled LDA is designed to specify multiple dimensions of a given text that correspond to manually labeled tags [12]. In CS2013, exemplar courses with knowledge distribution information show that a course generally contains knowledge

Table 1: KA-KU-Topic knowledge structure in CS2013 [14]

KA	KU	Topics
Algorithms and Complexity (AL)	Basic Analysis	•Big O notation •...
	Algorithm Strategies	•Greedy algorithms •...
	...	•...

Table 2: An example of syllabus information in CS2013 [14]

What is covered in the course?
• The modeling process • Two system dynamic tool tutorials • Computational error • ...

from more than one KA or KU. Therefor, this method is suited when addressing a syllabus text that is labeled with multiple predefined tags—KA/KU in this case.

Considering that topics listed in BoK are highly compact representations of knowledge, we resort to external texts to complement the content of BoK. Specifically, we integrated snippet information retrieved from queries of a KU to improve the accuracy of predictions .

4.2 Dataset

81 exemplar courses, whose course information and knowledge distributions are assigned by the course instructor, are included. As shown in Table 2, the answer to the question "What is covered in the course?" is viewed as the syllabus information of a course. In addition, the information offered by the instructor on how the lecture hours of a course are allocated to each KA and KU is referred to as the ground truth of our method (e.g., 35.5 hours in CN, 3 hours in IS,...). After excluding malformed course information, 73 exemplar courses were used in the course content analysis.

Regarding the external texts, we threw 3 types of queries to retrieve snippet texts of websites from Google Custom Search API. The queries are formed by using: (1)KU title alone, (2) KA and KU title, (3) KU title and its top 3 representative terms (chosen by their tf-idf values, which represent an effective as an indicator of the importance of a term over a set of documents). 10 snippet texts were complemented to the content of each KU.

4.3 Procedures

4.3.1 Training set

As a trial analysis, we exploit the predictability of curriculum guidelines by conducting experiments with different training sets. Among all the experiments, 30 exemplar course syllabi were chosen randomly as the testing set. Concerning the training set, we set 2 variables, forming 8 patterns, to improve the accuracy of predictions. The first variable denotes whether manually labeled syllabus texts are used in the training set or BoK texts alone are used. The

Table 3: Experiment id

	BoK	BoK_Snippet1	BoK_Snippet2	BoK_Snippet3
BoK	KA-1-0	KA-1-1	KA-1-2	KA-1-3
BoK+Course Syllabus	KA-2-0	KA-2-1	KA-2-2	KA-2-3

second denotes what type of snippet texts are used, with "0" denoting using BoK texts alone.

Table 3 presents the naming of the experiments according to their content of the training set. The names of experiments for the prediction of KU knowledge coverage follow the same naming scheme. We conduct all 8 experiments on predicting knowledge coverage at the level of both KA and KU, and we add "KA" or "KU" to the experiment id to indicate the different targets.

4.3.2 *Evaluation*

To evaluate the predicted probabilities over KAs/KUs of a syllabus, we apply the Normalized Discounted Cumulative Gain (nDCG), which is used to evaluate the relevance of a document rank to a given query in classic Information Retrieval (IR). We choose the nDCG because it addresses relevance as a non-binary value, which is better suited to our case where the relevance of a document corresponds to lecture hours. For each course, we compare the ranked list of KAs/KUs that is predicted by our method, with the ranked list of KAs/KUs that is allocated by the course instructor. The computation is conducted using the following equations :

$$\begin{cases} G_c[i] = rel_c[i] \\ DCG_c[k] = \sum_{i=1}^{k} \frac{G_c[i]}{\log_2 i+1} \\ nDCG_c[k] = \frac{DCG[k]}{IDCG[k]} \end{cases} \tag{1}$$

Here, $rel_c[i]$ denotes the lecture hours allocated to the i$^{\text{th}}$ KA/KU for a given course; DCG denotes the discounted cumulative gain of the ranked KA/KU list that is predicted by our method, and IDCG denotes the one of the ranked KA/KU list assigned by the course instructor.

4.4 Results

We utilized the *Stanford Topic Modeling Toolbox* to compute the KA/KU distributions of a syllabus and the Python library *Scikit Learn* to compute the tf-idf value of each term appearing in a BoK. Other data processes, such as the computation of the nDCG, are implemented in Python. Concerning the most representative terms for each KU, we chose the top three terms from a vocabulary of 2486 non-stopword terms. Because the average number of KAs that a course covers assigned by the instructor is 2.67, being 9.04 for KU, we focus on the nDCG value of $k = 3$ for KA, of $k = 9$ for KU. The results for each experiment are shown in Figure 5.

4.5 Discussion

As observed in Figure 5, all the nDCG values of the experiments with a training set containing BoK texts alone are higher than those with a training set consisting of both BoK texts and exemplar course syllabus texts. In our data set, all the BoK texts are annotated with one label, whereas exemplar course syllabus texts are annotated with multiple

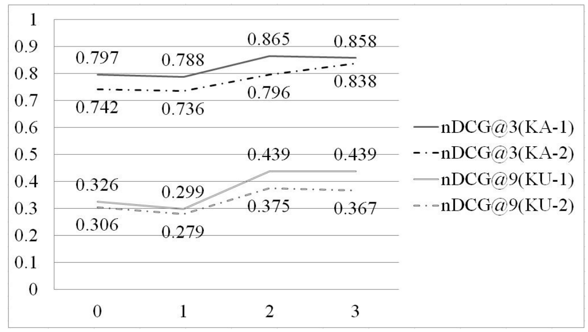

Figure 5: The nDCG values of each experiment. The vertical axis denotes the value of nDCG, which varies from 0 to 1. The horizontal axis denotes the second variable with regard to the naming of the experiments—the type of snippet texts used in training set.

labels. This unbalanced number of labels in the training set may reduce the precision of prediction obtained using Labeled LDA. However, from a positive perspective, this result indicates the potential of only using pre-collected documents of domain knowledge instead of collecting annotated course syllabi when predicting the knowledge coverage of a given course.

Two types of snippet texts exhibit a positive effect on predicting KA/KU knowledge coverage. They are snippet texts queried from KU titles with their corresponding KA title and snippet texts queried from KU titles with their top 3 representative terms. For example, nDCG@3 of KA-1-2 and KA-1-3 are notably higher than those of KA-1-0. A similar trend can also be observed in the case of predicting KUs. In contrast, nDCG@3 of KA-1-1 are lower than those of KA-1-0, which indicates that the external texts obtained from the KU title query drag down the performance of our model. One possible reason that can be inferred is that a sole KU title can produce substantial noise when it is used without context. For example, "processing" has a much broader meaning than that in the context of "Computational Science". Other ambiguous KU titles, such as "Basic Logic" and "Data, Information, and Knowledge", are prone to increasing the prevalence of this type of mistake. Overall, queries consisting of KA titles and KU titles or KU titles and their keywords provide effective and relevant texts when predicting knowledge coverage.

To seek deeper factors that may contribute to the correctness of a prediction, we examined an exemplar course syllabus and compared it with BoK and external texts. We found:

- Some synonymous or semantically similar phrases (e.g.,

"strategies for choosing..." and "apply...") may not be detected by our method.
- There exist internal relationships between KUs (e.g., KU "Processing" under KA "Computational Science" overlaps with KU "Algorithms and Design" under KA "Software Development Fundamentals"), which may mislead the prediction of KUs.
- An increase in performance in predicting KAs may not guarantee an improvement in predicting KUs. Because in some cases, the improvement in predicting KAs is achieved by assigning a probability to an incorrect KU under the KA.

5. CONCLUSION AND FUTURE WORK

Summarizing, we proposed a supporting system that recommends an effective and efficient path of learning objects for a given individual. To realize this system, a threefold architecture is needed—Domain model, Learner model and Adaptation Model. As an initiative step, we conducted a course content analysis, in which Labeled LDA was utilized to predict the knowledge coverage of a course. The result provided the positive indication that involving external explanatory texts on domain knowledge facilitates the prediction of the knowledge coverage of unknown course syllabi. However, the precision of the the current experiment needs further improvement in addressing texts semantically. Specifically, a bigram or trigram method is expected to perform better than the unigram method. In addition, separate nouns and noun-phrases may increase the precision. From a holistic perspective, we also need to consider the estimation of learner characteristics when constructing domain knowledge bases. For example, a framework of knowledge that connects knowledge itself with its learning outcomes may be instrumental in mapping learning objects to learners.

6. ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Number 15K00423 and the Kayamori Foundation of Informational Science Advancement.

7. REFERENCES

[1] T. Barnes. Q-matrix Method: Mining Student Response Data for Knowledge. Technical report, 2005.

[2] Y. Belanger and J. Thornton. Bioelectricity: A Quantitative Approach Duke University's First MOOC. Report, 2013.

[3] R. J. C. Bose, O. Deshmukh, and B. Ravindra. Discovering Concept Maps from Textual Sources. In *Proceedings of the 8th International Conference on Educational Data Mining*, 2015.

[4] P. Brusilovsky, J. Eklund, and E. Schwarz. Web-based education for all: a tool for development adaptive courseware. *Computer Networks and ISDN Systems*, 30(1-7):291–300, 1998.

[5] F. Gasparetti, C. Limongelli, and F. Sciarrone. Exploiting wikipedia for discovering prerequisite relationships among learning objects. In *Proceedings of International Conference on Information Technology Based Higher Education and Training*, pages 1–6, 2015.

[6] A. D. Ho, I. Chuang, J. Reich, C. A. Coleman, J. Whitehill, C. G. Northcutt, J. J. Williams, J. D. Hansen, G. Lopez, and R. Petersen. HarvardX and MITx: Two Years of Open Online Courses Fall 2012-Summer 2014. SSRN Scholarly Paper ID 2586847, Social Science Research Network, 2015.

[7] I. Kiyoshi et al. Investigation on the Educational Contents among Informational Science and Engineering Departments by Using Syllabus (Intermediate Report). Technical Report 6, Information Processing Society of Japan, 2010.

[8] A. Klašnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac. E-Learning personalization based on hybrid recommendation strategy and learning style identification. *Computers & Education*, 56(3):885–899, 2011.

[9] F. W. B. Li, R. W. H. Lau, and P. Dharmendran. An Adaptive Course Generation Framework. *Int. J. Distance Educ. Technol.*, 8(3):47–64, July 2010.

[10] N. Matsuda, T. Furukawa, N. Bier, and C. Faloutsos. Machine beats experts: Automatic discovery of skill models for data-driven online course refinement. In *Proceedings of the 8th International Conference on Educational Data Mining*, 2015.

[11] A. Paramythis and S. Loidl-Reisinger. Adaptive Learning Environments and eLearning Standards. *ELECTRONIC JOURNAL OF ELEARNING, EJEL: VOL 2. ISSUE*, 2:181–194, 2004.

[12] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A Supervised Topic Model for Credit Attribution in Multi-labeled Corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 248–256. Association for Computational Linguistics, 2009.

[13] T. Sekiya, Y. Matsuda, and K. Yamaguchi. Curriculum analysis of CS departments based on CS2013 by simplified, supervised LDA. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge*, pages 330–339, 2015.

[14] I. C. Society. Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. Technical report, ACM, 2013. 999133.

[15] S. Sosnovsky and P. Brusilovsky. Evaluation of topic-based adaptation and student modeling in QuizGuide. *User Modeling and User-Adapted Interaction*, 25(4):371–424, 2015.

[16] J. C. R. Tseng, H.-C. Chu, G.-J. Hwang, and C.-C. Tsai. Development of an adaptive learning system with two sources of personalization information. *Computers & Education*, 51(2):776–786, 2008.

[17] S.-S. Tseng, P.-C. Sue, J.-M. Su, J.-F. Weng, and W.-N. Tsai. A new approach for constructing the concept map. *Computers & Education*, 49(3):691–707, 2007.

[18] A. Vuong, T. Nixon, and B. Towle. A method for finding prerequisites within a curriculum. In *Proceedings of the 4th International Conference on Educational Data Mining*, 2011.

[19] J. Řihák, R. Pelánek, and J. Nižnan. Student Models for Prior Knowledge Estimation. In *Proceedings of the 8th International Conference on Educational Data Mining*, pages 109–116, 2015.