Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8813615, 13 pages
https://doi.org/10.1155/2020/8813615

Research Article

A Novel Search Ranking Method for MOOCs Using Unstructured Course Information

Weiqiang Yao,[1] **Haiquan Sun**,[1,2] **and Xiaoxuan Hu**[1,2]

[1]*School of Management, Hefei University of Technology, Hefei, Anhui 230009, China*
[2]*Key Laboratory of Process Optimization and Intelligent Decision Making, Ministry of Education, Hefei, Anhui 230009, China*

Correspondence should be addressed to Weiqiang Yao; wqyao@ustc.edu.cn

Received 25 June 2020; Revised 15 August 2020; Accepted 8 September 2020; Published 23 September 2020

Academic Editor: Yin Zhang

Massive open online courses (MOOCs) are a technical trend in the field of education. As the number of available MOOCs continues to grow dramatically, the difficulty for learners to find courses that satisfy their personalized learning goals has also increased. Unstructured texts, such as course descriptions and course skills, contain rich course information and are useful for MOOC platforms in constructing personalized services. This paper proposes a novel search ranking method for MOOCs that integrates unstructured course information. We propose a latent Dirichlet allocation-based model to cluster courses into groups based on course descriptions. Courses in the same cluster are considered to share similar educational contents. We then propose the CourseRank algorithm based on the information of course skills to recommend and rank courses when students search for or click on a specific course. Our experiments on the dataset from Coursera indicate that our method is able to cluster courses effectively and produce satisfactory ranking results for courses in MOOC platforms.

1. Introduction

Massive open online courses (MOOCs) have gained considerable global attention in the field of education. It offers a new way for organizations to share their knowledge and offer world-class education to the public [1]. A survey by Class Central shows more than 900 universities around the world launched more than 11.4 thousand MOOCs in various MOOC platforms in 2018 [2]. The number of students enrolled in MOOCs increased from 78 million in 2017 to more than 101 million in 2018.

With the increasing popularity of MOOCs, hosting as many courses as possible to satisfy various demands from students is a profitable business strategy for MOOC platforms. However, a common issue for MOOC platforms is that many courses in a platform have similar titles but different technical contents. Many courses with different titles may also have similar content because they cover the same knowledge points. Take http://Coursera.com/, for example; when we search the keyword "machine learning," more than 100 courses show up with titles containing the keyword "machine learning." These courses, such as "TensorFlow in Practice," in the list of search results also include related knowledge points although their titles do not have the keyword "machine learning." In such a case, if a student wants to learn certain knowledge or skills, the large number of similar courses makes it difficult for students to choose the right courses and achieve their personalized learning objectives. From the perspective of the platform, designing methods to assist students in finding MOOCs that can satisfy their learning objectives are necessary.

Many methods have been proposed to construct selection and ranking models for MOOCs. For example, Bousbahi and Chorfi [3] designed the case-based reasoning (CBR) approach and information retrieval technique to recommend MOOCs for learners. Elbadrawy and Karypis [4] investigated how student characteristics and course features affect course enrollment patterns. In these studies, the structured demographic characteristics, study records, and course features are the main information used to infer the learning

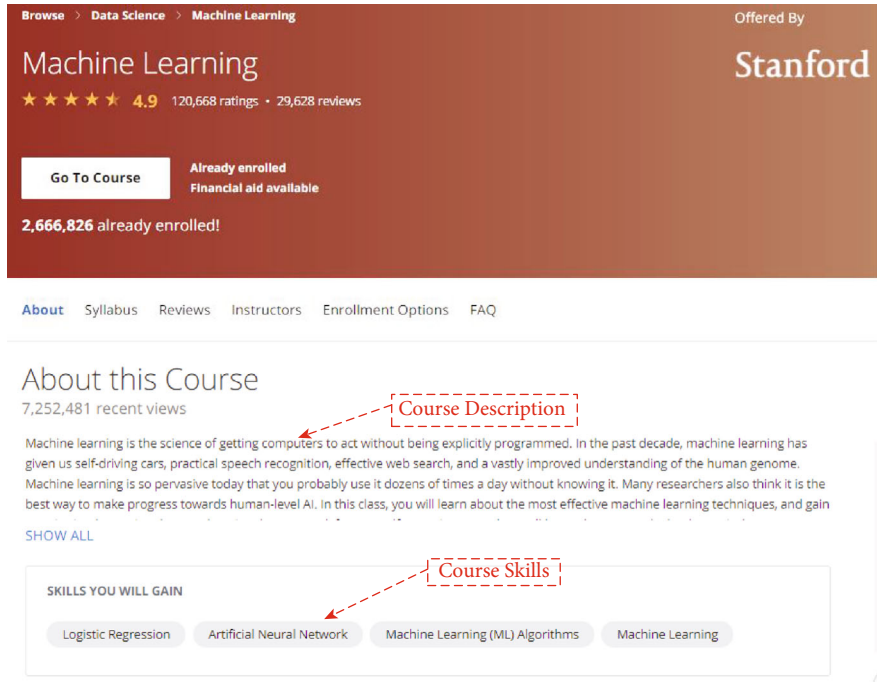


Figure 1: Unstructured information for a MOOC.

preferences of students. However, a large amount of unstructured data has not been explored fully to analyze student behaviors and provide personalized services.

In the MOOC platform, unstructured textual data, such as course descriptions, course skills, and student reviews usually imply useful course features. For example, http://Coursera.com/ (Figure 1) uses "About this Course" to introduce a specific course. Teachers can also present "skills you will gain" to indicate the contents and methods to be delivered in the course. The course description and skills help students know the teaching contents. Students can thus evaluate whether a course can meet their learning objectives [5]. For course search services, utilizing the textual information is useful for platforms to understand both the courses' teaching contents and students' learning objectives.

This paper designed a novel search ranking method for MOOCs with the unstructured course description released in MOOC platforms and the course skills given by teachers. When students click on a specific course or search some keywords in a MOOC platform search engine, we propose a model that can analyze the unstructured textual information and present students with a sorted list of courses. In the proposed method, the first stage is a latent Dirichlet allocation-(LDA-) based model to cluster courses into groups. The courses in the same clusters are considered to cover similar knowledge because they have comparable learning topics. All MOOC platforms offer many courses, and thus, each course cluster obtained in the first stage usually includes many courses. Hence, presenting all courses in the same cluster to students when they search for a keyword or click on a specific course is unreasonable and impractical. The second stage is a CourseRank algorithm to rank the courses in a cluster with the unstructured course skills. Courses with higher rankings are then selected and presented to the students. In general, the contributions of this study are three-fold:

(1) Technology-enhanced learning is a promoting trend in the field of education. Daniel suggested that working with big data and data science requires specialized skills lacking in many educational researchers [6]. This paper introduces machine learning technologies (i.e., LDA and PageRank algorithm) to the field of education research. The proposed models benefit research in the field of education by providing new technologies and tools to help researchers work with big data and data science. This study is valuable because it can help understand learners' cognition by analyzing the unstructured course information and can increase business efficiency for the MOOC platforms

(2) We employ the unstructured course description released in MOOC platforms and the course skills given by teachers for the course ranking algorithm. Although the unstructured course information contains rich knowledge on learning and teaching objectives, researches that have integrated the unstructured textual information are minimal, especially course skills information, into the course search ranking problem

(3) Instead of segmenting courses by clustering description words, the LDA-based model clusters the courses by extracting latent topics implied in the contents. This strategy can improve the results of

the course clustering and help platforms filter out unrelated courses to meet the individual preferences of students

The remainder of our research is organized as follows. Section 2 reviews the related work in literature. In Section 3, we propose the course clustering model and the course ranking model. In Section 4, we conduct experiments on the dataset from Coursera to test our proposed method. Section 5 concludes our research and provides the future directions.

2. Related Work

In this section, we review the previous works on MOOCs relevant to our study. We review the literature on student behaviors in the MOOC environment and the machine learning methods for MOOC ranking.

2.1. Student Behavior in the MOOC Environment. In the educational research, MOOC has drawn wide attention from scholars because it has been considered as one of the most effective online learning forms [7]. Bodily et al. regarded MOOC as one of the most important trends for instructional design and technology [8]. Zhu et al. reviewed MOOC research from 2014 to 2016 and classified current researches into several categories [9]. Costello et al. conducted a systematic review of research about the role of Twitter in the context of MOOCs from 2011 to 2017 [10]. Summarizing these literature reviews and current researches on MOOCs, student behavior is seen to be the most popular topic in literature, and current research generally used survey data to analyze student behaviors by descriptive statistics.

To study student behaviors in MOOCs environment, many scholars focused on student engagement in courses. For example, Aparicio et al. proposed a theoretical framework to identify the factors impacting MOOC use and satisfaction and empirically measure these factors in a real MOOC context [11]. Deng et al. developed and validated a MOOC engagement scale to measure learner engagement [12]. They found that behavioral engagement, emotional engagement, cognitive engagement, and social engagement are the four dimensions of student engagement in MOOCs. By taking into account factors such as expectancies, values, and social influence, Luik et al. studied factors that motivate the enrolment of learners in programming MOOCs [13]. Their study showed that interest in the course and personal suitability is the highest-rated motivational factors. Social influence and usefulness related to certification are the lowest-rated factors. Current literature investigated student engagement in MOOCs from the perspective of self-determination theory and the theory of relationship quality [14].

Aside from investigating student engagement, current literature also studied the learning behaviors of students after they enrolled in MOOC platforms. Cohen et al. characterized the active learners in forums and found that the completion status of learners significantly correlates to their activity in the forums [15]. Hood et al. examined how the current role and context of learners influence their ability to self-regulate their learning in the MOOC environment [16]. Significant differences were identified between learners with different characteristics. Guo and Reinecke studied the navigation behavior of students in the learning process [17]. Their results indicated that older students and those from countries with smaller student-teacher ratios are more comprehensive and nonlinear when navigating through the course.

The related works reviewed above indicate that most existing studies on MOOCs are empirical studies that use surveys or interview data [18]. New data sources and new methodologies are required to analyze learning behaviors in the MOOC environment. The literature review indicated that students with various characteristics often have different learning preferences and behaviors. Therefore, the MOOC platform needs a design operative strategy to predict student preference and provide suitable courses [19].

2.2. Machine Learning for MOOC Ranking. In the past several years, machine learning methods have been applied gradually to address issues in the field of MOOC research. Researchers employed methods such as random forest (RF), support vector machine (SVM), and LDA to understand student behaviors [20]. For example, Peng and Aggarwal transformed the MOOC dropout problem as a classification issue and designed several machine learning models based on SVM, gradient boosting decision trees, AdaBoost, and RF to solve the problem [21]. LDA is a popular text mining method for MOOCs. Ramesh et al. designed a seeded LDA model to understand MOOC discussion forums [22]. Atapattu and Falkner proposed an LDA-based framework to generate and label discussion topics automatically [23].

Course recommendation is an important research topic that emphasizes the employment of machine learning methods in the MOOC environment. Guo and Reinecke suggested that the function of course recommendation is necessary for MOOC platforms because it can help platforms provide proper courses to students and incentivize them to engage with the study process [17]. Hence, Bousbahi and Chorfi designed a MOOC recommendation method using CBR, which can effectively find the best learning resources for students [3]. Elbadrawy and Karypis proposed a domain-aware method to recommend courses based on the academic features of student and course groups [4]. Pang et al. proposed a multilayer bucketing recommendation method to recommend courses on MOOC platforms and designed a map-reduced technique to improve recommendation efficiency [24].

The above literature indicates that machine learning is one of the most popular methodologies in education research. However, although existing methods are useful, they usually rank courses by analyzing structured learning records or learner features. The unstructured data such as course descriptions and tags are yet to be explored. This paper employed LDA and PageRank to generate reasonable search results in the MOOC platforms. LDA and PageRank are machine learning methods widely used in various fields [25]. This study utilized the LDA algorithm to analyze course descriptions and cluster courses into groups, whereas the

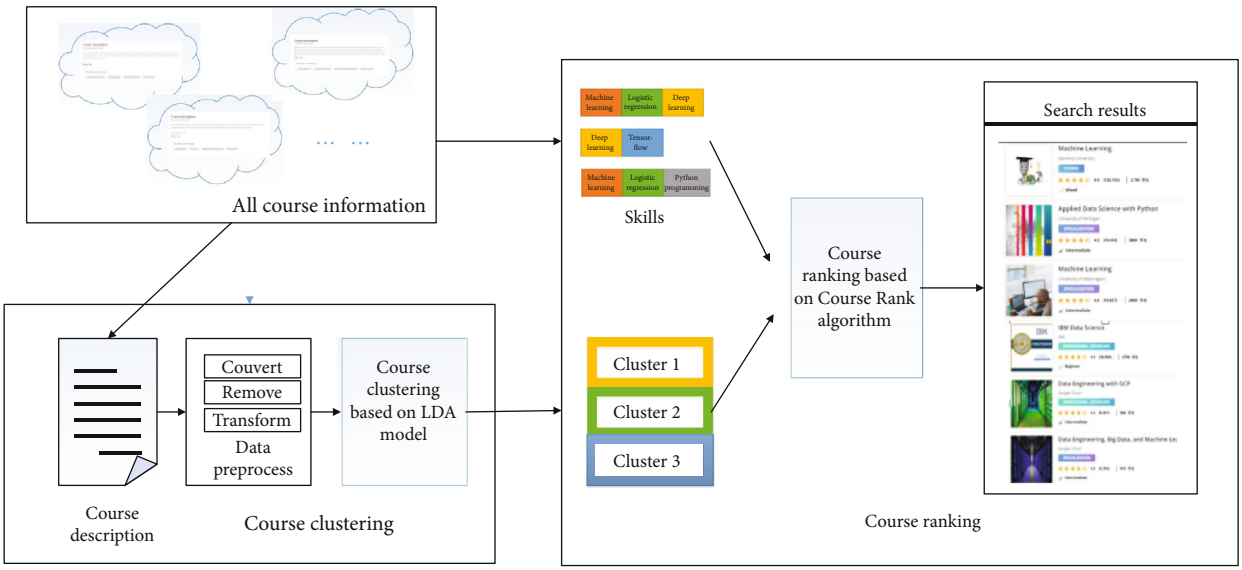


Figure 2: Framework of course ranking method.

PageRank algorithm was used to rank the courses in the same clusters. The proposed method is detailed next.

3. Search Ranking Method for MOOCs

In this section, we propose the search ranking method for courses based on the LDA and PageRank algorithm. Figure 2 provides the framework of our search ranking method. Figure 2 shows that based on the textual description information, we design a LDA-based model to cluster courses. For the courses in each cluster, a course ranking algorithm is proposed based on the skills which will gain through the courses. We provide the details of the proposed search ranking method for the course in the following sections.

3.1. Stage 1: LDA-Based Model for Course Description Clustering. We now provide the LDA-based model for course clustering. LDA uses an unsupervised Bayesian model to capture context-specific dimensions implied in the unstructured course description. Based on LDA, each observed word in the course description can be allocated to a certain topic and the course description is regarded as a mix of multiple topics. In this section, we first provide the related formulation, followed by an LDA-based model for course description clustering. Then, we propose the parameter inference process from the course description information.

3.1.1. Formulation. In our model, a collection of course description exists, $M = \{\mathbf{w}_m\}_{m=1}^{|M|}$ and $\mathbf{w}_m = \{w_{mi}\}_{i=1}^{N_m}$ is a vector of words in course description $\boldsymbol{m}$.

Definition 1. (Number definition). K, M, and V are the number of course topics, course descriptions, and unique words in all course descriptions, respectively. Words are indexed by $v \in \{1, 2, \cdots, V\}$, and N_m is the number of the word taken in course descriptions.

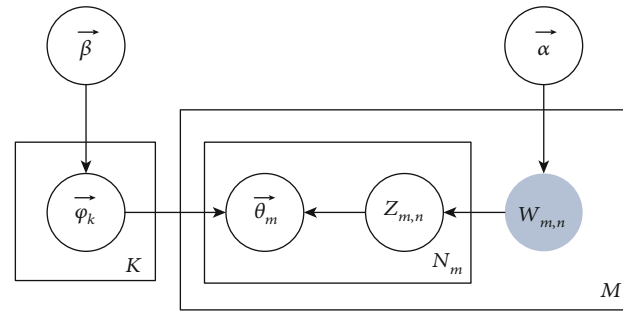


Figure 3: Graphical representation of LDA.

Definition 2. (Course topics and words). $Z_{m,n}$ is the topic associated with the n-th word in the course description $\boldsymbol{m}$, and $W_{m,n}$ is the n-th word in document $\boldsymbol{m}$.

Definition 3. (Variables for probability distribution). $\overrightarrow{\theta_m}$ is the multinomial distribution of topics specific to course description $\boldsymbol{m}$, which is a proportion for each course description, and each one is an $M * N$ matrix. $\overrightarrow{\varphi_k}$ is the multinomial distribution of words specific to the topics $\boldsymbol{k}$, which is a proportion for each topic and each one is a $K * V$ matrix.

Definition 4. (Variables for hyperparameter). $\vec{\alpha}$ is the hyperparameter to the multinomial distribution $\overrightarrow{\theta}$. $\vec{\beta}$ is the hyperparameter to the multinomial distribution $\vec{\varphi}$.

3.1.2. Model Description. This section presents the details of the LDA-based model for course description clustering. Figure 3 illustrates the relationships between the parameters used in the proposed model. The generative process is presented in Algorithm 1. For a better explanation, this model can be divided into two phases.

For each course topic $k \in [1, K]$:
(a) Draw a multinomial $\vec{\varphi}_k$ from a Dirichlet prior $\vec{\beta}$;
For each course description $m \in [1, M]$:
a. Draw a multinomial $\overrightarrow{\theta}_m$ from a Dirichlet prior $\vec{\alpha}$;
b. For each world $n \in [1, N_m]$ in course description m:
i. Draw a topic $Z_{m,n}$ from multinomial $\overrightarrow{\theta}_m$;
ii. Draw a word $W_{m,n}$ from multinomial $\vec{\varphi}_k(k = Z_{m,n})$;

Algorithm 1: Generative process of LDA.

(1) Phase 1: Modeling the Topic of the Course Description. In this model, we assume that each topic for the course description is represented by a word distribution. We model each topic $k \in \{1, 2, \cdots, K\}$ as a vector $\vec{\varphi}_k$ that follows a Dirichlet distribution over the V words.

$$\vec{\varphi}_k \sim \mathrm{Dir}(\beta), \tag{1}$$

where β is a symmetric Dirichlet prior.

(2) Phase 2: Modeling Words Distribution of Course Description. The key point of the LDA-based model for course description clustering is that each course description can be viewed as a mix of the latent topics, and each word in the course description has the corresponding topic. We model each course description $m \in \{1, 2, \cdots, M\}$ as a vector $\overrightarrow{\theta}_m$ that follows a Dirichlet distribution over the K topics.

$$\overrightarrow{\theta}_m \sim \mathrm{Dir}(\alpha), \tag{2}$$

where α is a symmetric Dirichlet prior.

We use the multinomial distribution $\overrightarrow{\theta_m}$ to sample a topic $Z_{m,n}$ for course contents. After determining the topic $Z_{m,n}$, we use the multinomial distribution $\overrightarrow{\varphi_k}$ to sample the word $W_{m,n}$.

3.1.3. Model Inference. The above process of the LDA-based model appears to be a relatively simple model but ensuring the accuracy of the derivation is difficult. We use Gibbs sampling to deal with this intractable question. Two steps (i.e., calculate the joint distribution and obtain the conditional distribution probability) are used to infer the parameters of the proposed model. The details of the reference process are as follows:

Calculate the Joint Distribution. The calculation of the joint distribution $P(\vec{W}, \vec{Z} \mid \vec{\alpha}, \vec{\beta})$ can be divided into two parts by

$$P\left(\vec{W}, \vec{Z} \mid \vec{\alpha}, \vec{\beta}\right) = P\left(\vec{W}, \vec{Z} \mid \vec{\beta}\right) P\left(\vec{Z} \mid \vec{\alpha}\right), \tag{3}$$

where $P(\vec{W}, \vec{Z} \mid \vec{\beta})$ is the probability of word generation in the entire course descriptions and $P(\vec{Z} \mid \vec{\alpha})$ is the probability of topic. Because the process of generating topics for the $\boldsymbol{M}$ courses in the course description sets is independent of each other, we can take the advantage of Dirichlet—the multinomial conjugated structure and conjugate priors to calculate the first probability in Equation (1) by

$$\begin{aligned} P\left(\vec{W}, \vec{Z} \mid \vec{\beta}\right) &= \int P\left(\vec{W} \mid \Phi, \vec{Z}\right) P\left(\Phi \mid \vec{\beta}\right) d\Phi \\ &= \int \prod_{k=1}^{K} \prod_{v=1}^{V} \varphi_{k,v}^{n_k^{(v)}} \prod_{k=1}^{K} \left(\frac{\Gamma\left(\sum_{v=1}^{V} \beta_v\right)}{\prod_{v=1}^{V} \Gamma(\beta_v)} \prod_{v=1}^{V} \varphi_{k,v}^{\beta_v - 1} \right) d\Phi, \end{aligned} \tag{4}$$

where $n_k^{(v)}$ is the number of words $\boldsymbol{v}$ assigned to topic $\boldsymbol{k}$ and $\Gamma(x)$ in Equation (4) is the gamma function. In a similar way, $P(\vec{Z} \mid \vec{\alpha})$ can be calculated by

$$\begin{aligned} P\left(\vec{Z} \mid \vec{\alpha}\right) &= \int P\left(\vec{Z} \mid \theta\right) P\left(\theta \mid \vec{\alpha}\right) d\theta \\ &= \int \prod_{m=1}^{M} \prod_{k=1}^{K} \theta_{\mathrm{m,k}}^{n_{\mathrm{m}}^{(\mathrm{k})}} \prod_{\mathrm{m}=1}^{M} \left(\frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_{\mathrm{m,k}}^{\alpha_{\mathrm{k}} - 1} \right) d\theta, \end{aligned} \tag{5}$$

where $n_m^{(k)}$ represents the number of words in course description $\boldsymbol{m}$ assigned to topic $\boldsymbol{k}$.

Through Equations (4) and (5), we can obtain the joint contribution:

$$\begin{aligned} P\left(\vec{W}, \vec{Z} \mid \vec{\alpha}, \vec{\beta}\right) = &\left(\frac{\Gamma\left(\sum_{v=1}^{V} \beta_v\right)}{\prod_{v=1}^{V} \Gamma(\beta_v)} \right)^{K} \left(\frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \right)^{M} \\ &\times \prod_{\mathrm{k}=1}^{K} \frac{\prod_{v=1}^{V} \Gamma\left(n_k^{(v)} + \beta_v\right)}{\Gamma\left(\sum_{v=1}^{V} \left(n_k^{(v)} + \beta_v\right)\right)} \\ &\cdot \prod_{m=1}^{M} \frac{\prod_{k=1}^{K} \Gamma\left(n_m^{(k)} + \alpha_k\right)}{\Gamma\left(\sum_{k=1}^{K} \left(n_m^{(k)} + \alpha_k\right)\right)}, \end{aligned} \tag{6}$$

Obtain the Conditional Distribution Probability. Using the chain rule, the conditional probability can be obtained as

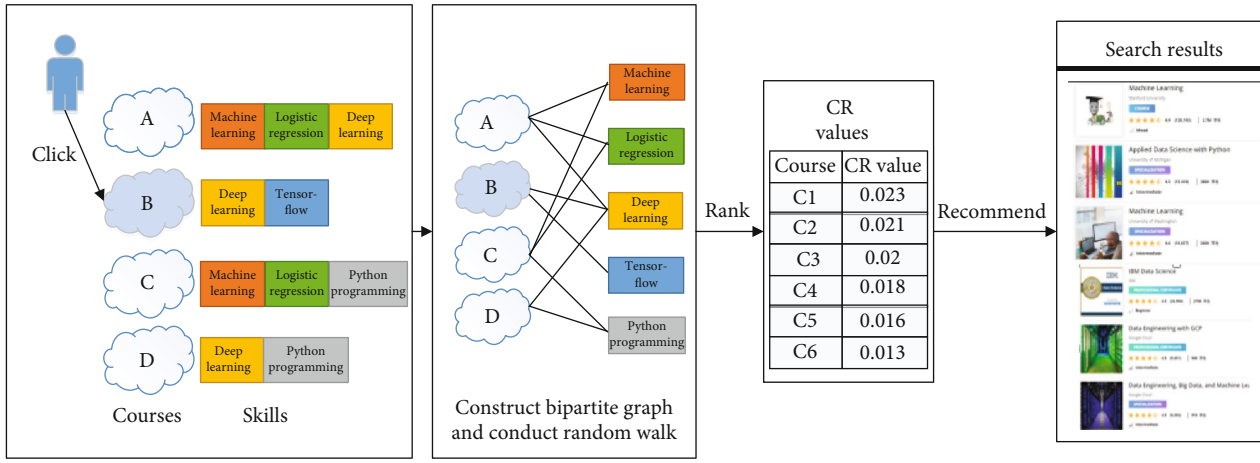


Figure 4: Framework of CourseRank algorithm.

$$P\left(Z_{m,n} \mid \vec{W}, \vec{Z}_{\neg(m,n)}, \vec{\alpha}, \vec{\beta}\right) = \frac{P\left(Z_{m,n}, W_{m,n} \mid \vec{W}_{\neg(m,n)}, \vec{Z}_{\neg(m,n)}, \vec{\alpha}, \vec{\beta}\right)}{P\left(W_{m,n} \mid \vec{W}_{\neg(m,n)}, \vec{Z}_{\neg(m,n)}, \vec{\alpha}, \vec{\beta}\right)},$$

$$\propto \frac{P\left(\vec{W}, \vec{Z} \mid \vec{\alpha}, \vec{\beta}\right)}{P\left(\vec{W}_{\neg(m,n)}, \vec{Z}_{\neg(m,n)} \mid \vec{\alpha}, \vec{\beta}\right)} \propto \frac{n_{Z_{m,n}}^{(W_{m,n})} + \beta_{W_{m,n}} - 1}{\sum_{v=1}^{V}\left(n_{Z_{m,n}}^{(v)} + \beta_v\right) - 1} \times \left(n_m^{(W_{m,n})} + \alpha_{Z_{m,n}} - 1\right), \tag{7}$$

where $\neg(m, n)$ is a two-dimensional subscript, $\vec{W}_{\neg(m,n)}$ corresponds to all the words in the course descriptions except for the n-th word in the course description $\boldsymbol{m}$, $\vec{Z}_{\neg(m,n)}$ is the topic assignments for all words except for the n-th word in the course description $\boldsymbol{m}$.

Finally, based on the definition of Dirichlet–multinomial conjugated structure and Bayes rule, we can to obtain the multinomial parameter sets θ and $\Phi$by

$$P\left(\vec{\theta}_m \mid \vec{Z}_{m,n}, \vec{\alpha}\right) = \frac{P\left(\vec{\theta}_m \mid \vec{Z}_{m,n}, \vec{\alpha}\right)}{P\left(\vec{Z}_{m,n}, \vec{\alpha}\right)} = \frac{1}{Z_{\vec{\theta}_m}} \prod_{k=1}^{K} \theta_{m,k}^{n_m^{(k)} + \alpha_k - 1}$$
$$= \text{Dirichlet}\left(\vec{\theta}_m \mid \vec{n}_m + \alpha\right), \tag{8}$$

$$P\left(\vec{\varphi}_k \mid \vec{Z}, \vec{W}, \vec{\alpha}\right) = \frac{P\left(\vec{\varphi}_k, \vec{W} \mid \vec{Z}, \vec{\beta}\right)}{P\left(\vec{W} \mid \vec{Z}, \vec{\beta}\right)} = \frac{1}{Z_{\vec{\varphi}_k}} \prod_{v=1}^{V} \varphi_{k,v}^{n_k^{(v)} + \beta_v - 1}$$
$$= \text{Dirichlet}\left(\vec{\varphi}_k \mid \vec{n}_k + \vec{\beta}\right), \tag{9}$$

where $\vec{Z}_{m,n}$ is the topic assignments for all words in course description $\boldsymbol{m}$, that is, $\vec{Z}_{m,n} = \{Z_{m,n}\}_{n=1}^{N_m}$, $\vec{n}_m = \{n_m^{(k)}\}_{k=1}^{K}$ is the vector of topic observation counts for course description $\boldsymbol{m}$ and $\vec{n}_k = \{n_k^{(v)}\}_{v=1}^{V}$ that of word observation counts for topic $\boldsymbol{k}$. Using the expectation of the Dirichlet distribution on Equations (8) and (9), we can obtain the following result:

$$\varphi_{k,v} = \frac{n_k^{(v)} + \beta_v}{\sum_{v=1}^{V}\left(n_k^{(v)} + \beta_v\right)}, \tag{10}$$

$$\theta_{m,k} = \frac{n_m^{(k)} + \alpha_k}{\sum_{k=1}^{K}\left(n_m^{(k)} + \alpha_k\right)}. \tag{11}$$

In Equations (10) and (11), $\varphi_{k,v}$and $\theta_{m,k}$ represent, respectively, the probability distribution of the words in content topic k and the probability of the content topics in course description m. From the perspective of MOOC recommendation, a topic may correspond to a knowledge point or a specific skill taught in various courses. We can consider each topic as a cluster and assign a course to the topic that corresponds to the largest course-topic probability in $\theta_{m,k}$. Based on $\theta_{m,k}$, we can also employ a classical algorithm to cluster the coursers.

3.2. Stage 2: Course Ranking Algorithm for MOOCs. With the clustering step in Stage 1, irrelevant courses can be filtered out for specific study purposes. However, many courses in each cluster remain, which would have a negative effect on the search ranking task for courses. Hence, to choose the right courses from a course cluster and present a precise ranking list for students, this paper designs an algorithm called CourseRank based on skills, which will gain through the courses to rank the courses in the same cluster.

The algorithm framework is illustrated in Figure 4. The figure shows that based on course skills, we construct a bipartite graph to rank the courses in the same clusters. The constructed bipartite graph consists of two kinds of disjoint and independent sets. The nodes on the left side represent courses and the nodes on the right side are skills. Based on the course-skill bipartite graph, we design the CourseRank algorithm to rank courses in each cluster when a student

Input: Bipartite graph G,ε,*root, maxstep*
Output: *CR* value
0. Initiative the root node $CR(root)$=1 and other nodes *CR* value is 0
1. while $k <$ maxstep:
2. Set all nodes temp value are 0
3. From G, get node j and $j^{*}s$ out-edges set out_j
4. From out_j, get the nodes i connected to node j
5. compute relevance score: $temp[i] += \varepsilon^{*} CR[j]/(len(\text{out}_j))$
6. temp[*roof*]+=(1-ε)
7. *CR*=*temp*
8. return *CR*

Algorithm 2: CourseRank algorithm.

searches for a keyword or clicks on a specific course. The proposed CourseRank algorithm, which is a variation of PageRank, is a strategy to rank nodes in a graph. In the PageRank algorithm, nodes are assumed to be connected with each other. However, this assumption cannot apply to the course-skill bipartite graph because we are required to estimate the relevance of all the courses to a specific course. Hence, we employ Equation (12) to compute the random access probability of a course node in CourseRank:

$$\text{PR}(i) = (1-\varepsilon) r_i + \varepsilon \sum_{j \in \text{in}(i)} \frac{\text{PR}(j)}{|\text{out}(j)|}. \tag{12}$$

In Equation (12), $\text{PR}(i)$ represents the probability that course i is accessed, $\text{in}(i)$ refers to all courses pointing to course i, and $\text{out}(j)$ represents other courses set up by course j. We replace $(1-\varepsilon)/N$ in classical PageRank algorithm with $(1-\varepsilon)r_i$ to compute the probability that course i will stay on the current course after being clicked on by the student as the starting point. Indicator r_i is 1 if the course is the target course and 0 otherwise. Equation (12) makes sure that, by walking randomly from the target course, the proposed CourseRank algorithm can compute the correlation from all other courses to the target course.

The algorithm details of CourseRank are presented in Algorithm 2.

The CourseRank algorithm will converge quickly to a stable state by calculating and updating the probabilities recursively. Based on CourseRank results, $\text{CR}(i)$ is used as the value to rank course i. We present the *top- k* courses in the same cluster of target courses or the *top- k* courses in the clusters associated with the search keywords.

4. Experiments

4.1. Dataset. The data used in our experiment were obtained from http://Coursera.com/, one of the most famous MOOC platforms in the world. Our data consisted of 2399 courses and 3981 course skills. The information related to each course included the course name, course description text in "About this Course," and the skill tags in "Skills you will gain." Because each course corresponds to several skills and each skill may be used to mark multiple MOOCs, the number of distinct skills in our data is 1590.

With the raw data obtained from the MOOC platform, we conduct the following preprocessing operations to obtain clean data:

(1) Convert all letters into lowercase and remove punctuation and meaningless words. After the preprocessing operation, the average length of the course descriptions is 90.79. The maximum length is 844 and the minimum length is 9. (2) Generate a word frequency matrix. In our experiment, we consider a course description as a document and the descriptions for all courses as the corpus. We construct a dictionary for the course corpus, assign a unique number to each word, and count the frequency of each word in the corpus. Because many course descriptions have words not related strongly to the course, we also conduct an operation to remove the noisy words from the corpus (e.g., a, able, about, and above). In our experiment, we have 19,746 distinct words in the course corpus

4.2. Course Clustering. We now evaluate the performance of the proposed LDA-based method to cluster courses.

4.2.1. Baseline Methods and Evaluation Metrics. In our paper, we designed an LDA-based method to cluster courses in MOOC platforms. In practice, many methods can group courses into clusters. For example, *K*-means [26] and DBSCAN [27] are the well-known clustering methods and are widely used for MOOC research. Chang et al. employed *k*-means to investigate the effects of learning style preferences on student intentions regarding MOOCs [28]. Chen et al. applied DBSCAN to cluster the learners into interested groups and analyzed their learning patterns of the groups [29]. This paper compares the proposed LDA-based method with *k*-means and DBSCAN. Before utilizing *k*-means and DBSCAN to cluster course descriptions, we use the TF-IDF method [30] to transform each course description as a numerical vector and conduct clustering with the TF-IDF matrices.

We use the coherence score to evaluate the performances of the proposed clustering model and *k*-means. Coherence

Table 1: Cluster results for courses.

Cluster	Typical courses
Cluster 1	(1) Advanced Instructional Strategies in the Virtual Classroom, (2) Blended Learning: Personalizing Education for Students, (3) Critical Issues in Urban Education, (4) Emerging Trends & Technologies in the Virtual K-12 Classroom, (5) Foundations of Teaching for Learning: Being a Teacher, (6) Foundations of Virtual Instruction, (7) Get Interactive: Practical Teaching with Technology, (8) Learning to Teach Online, (9) Powerful Tools for Teaching and Learning: Web 2.0 Tools, (10) University Teaching.
Cluster 5	(1) A Crash Course in Data Science, (2) Applied Plotting, Charting & Data Representation in Python, (3) Applying Machine Learning to your Data with GCP, (4) Basic Data Processing and Visualization, (5) Big Data Applications: Real-Time Streaming, (6) Building Data Visualization Tools, (7) Business Intelligence Concepts, Tools, and Applications, (8) Business intelligence and data warehousing, (9) Data Manipulation at Scale: Systems and Algorithms, (10) Foundations of marketing analytics.
Cluster 13	(1) Advanced Business Strategy, (2) Advanced Competitive Strategy, (3) Becoming a changemaker: Introduction to Social Innovation, (4) Business Growth Strategy, (5) Creating and Developing a Tech Startup, (6) Decision-Making and Scenarios, (7) Design Thinking for Innovation, (8) Design Thinking for the Greater Good: Innovation in the Social Sector, (9) Entrepreneurship 1: Developing the Opportunity, (10) FinTech Foundations and Overview.
Cluster 17	(1) Advanced Data Structures in Java, (2) Advanced R Programming, (3) Algorithmic Thinking, (4) An Introduction to Interactive Programming in Python, (5) Big Data Analysis with Scala and Spark, (6) Building Web Applications in PHP, (7) Cloud Computing Concepts, (8) Code Yourself! An Introduction to Programming, (9) Computational Thinking for Problem Solving, (10) Computer Science: Programming with a Purpose.
Cluster 26	(1) Adapt your leadership style, (2) Applications of Everyday Leadership, (3) Bridging the Gap between Strategy Design and Delivery, (4) Building High-Performing Teams, (5) Building Your Leadership Skills, (6) Designing and Implementing Your Coaching Strategy, (7) Giving Helpful Feedback, (8) Global sustainability and corporate social responsibility: Be sustainable, (9) Human Resources Management Capstone: HR for People Managers, (10) Influencing People.

score [31] is widely used to evaluate clustering quality. In our experiment, a course cluster is reasonable if the most probable words in the cluster cooccur more frequently in the course corpus. The coherence score is defined as follows:

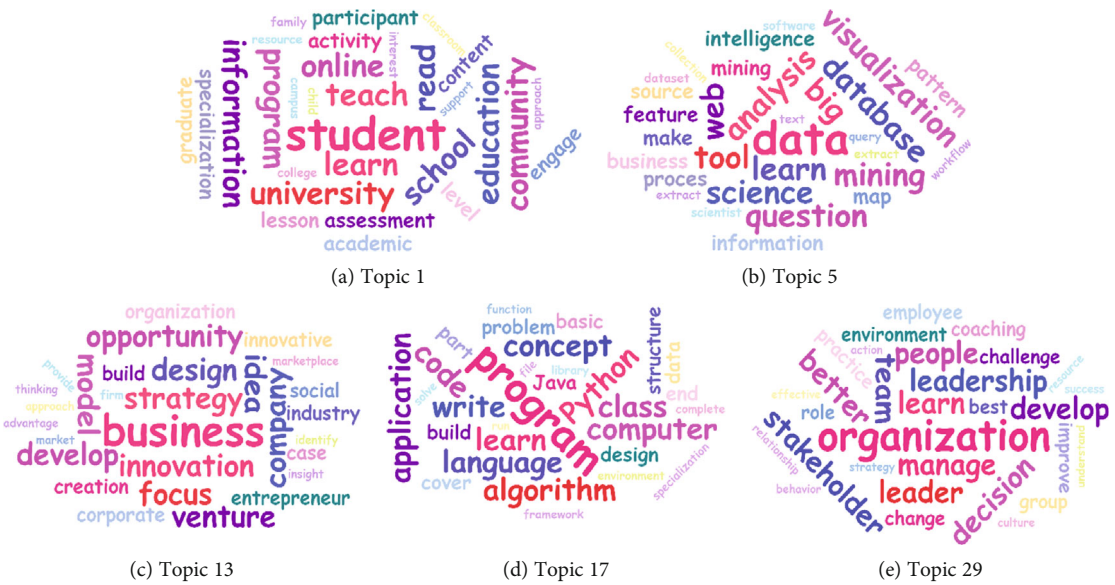
$$C\left(k\,;V^{(k)}\right)=\sum_{m=2}^{M}\sum_{l=1}^{m-1}\log\frac{D\left(v_m^{(k)},v_l^{(k)}\right)+1}{D\left(v_l^{(k)}\right)},\qquad(13)$$

where $V^{(k)}=(v_1^{(k)},\cdots,v_m^{(k)},\cdots,v_M^{(k)})$ is the list of the M most probable words in course cluster k, $D(v_l^{(k)})$ is the number of course descriptions containing word l, and $D(v_m^{(k)},v_l^{(k)})$ is the number of course descriptions containing word m and word l simultaneously.

4.2.2. Clustering Results. To obtain stable solutions, we run Gibbs samplers for 1000 iterations. In our experiment, $\alpha = 50/K$ and $\beta = 0.1$ where K is the number of clusters assumed by LDA. Based on the evaluation of optimal coherence value [32], both the number of clusters for the proposed method is set to be 36. To make a fair comparison, we predetermine the same cluster number for k-means.

We selected five clusters from the obtained clusters as examples and list them in Table 1. From Table 1, the proposed model can cluster courses with similar teaching objectives effectively. In Table 1, cluster 1 is a course group on teaching methodology. It gathers the courses for the new trend of teaching methods that can facilitate more effective learning environments. Students will gain skills, such as how to construct blended learning and how to organize interaction in the virtual classroom. Cluster 4 is a course group on data sciences, which includes content on data analysis, processing, visualization, and application in business intelligence and marketing. In the proposed model, course descriptions are analyzed by the LDA model. Therefore, we cluster courses according to their content topics rather than descriptive words. Many courses in the same clusters have distinct names but have similar teaching objectives for this reason. Cluster 12 is a course group on business strategy. In the cluster, we can see the courses that teach students how to formulate and innovate business strategies, especially in the new environment, such as the social and FinTech context. Cluster 16 contains courses about programming. Students can develop skills in data structure, programming language, and computational thinking ability. Based on the courses in cluster 25, students gain knowledge on how to build a team and form leadership in a team. From the courses, students can also learn how to communicate with others and optimize human resources management. In Figure 5, we illustrate the word clouds of the five clusters from which we can understand thoroughly the teaching objectives of the courses in each cluster.

Table 2 shows the comparison results on the coherence index between the proposed model and the baseline algorithms. We select the Top T words in each topic to evaluate the performance of these two methods. Table 2 shows that the proposed model always obtains the smaller coherence value regardless of the number of Top T. The proposed model performs better than k-means and DBSCAN. To test the robustness of the proposed method, we randomly split our data into two equal portions and cluster the courses in each portion by the three clustering methods. We illustrate the corresponding coherence scores on the top T representative words in Figure 6. From Figure 6, we can see that the



(a) Topic 1 (b) Topic 5

(c) Topic 13 (d) Topic 17 (e) Topic 29

Figure 5: Representative word clouds of clusters.

Table 2: Average coherence score on the top T representative words.

T	DBSCAN	k-means	LDA
5	-19.18	-17.14	-16.46
10	-115.46	-108.96	-92.18
15	-282.87	-262.51	-243.70
20	-541.50	-503.45	-472.54

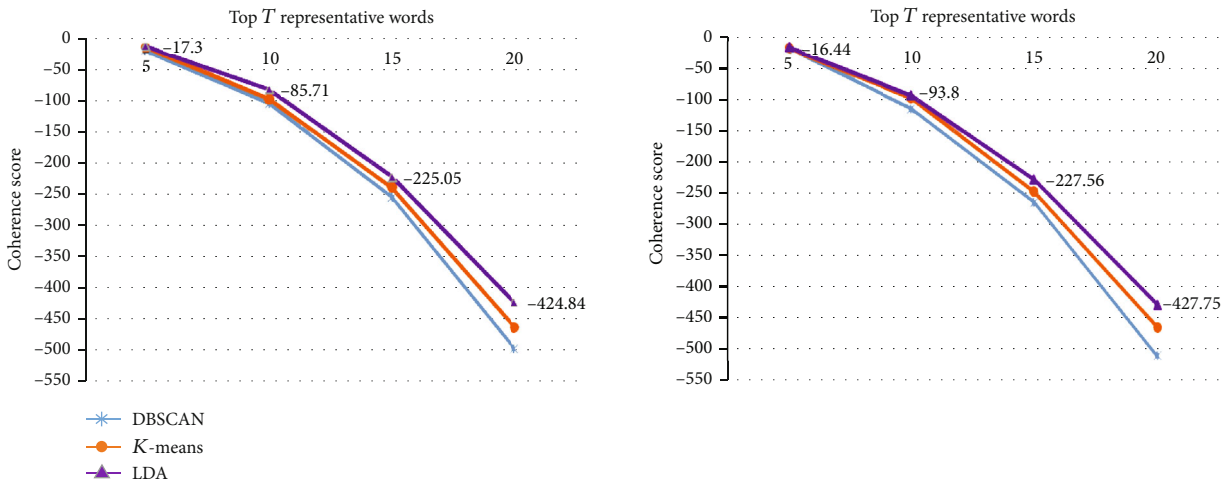
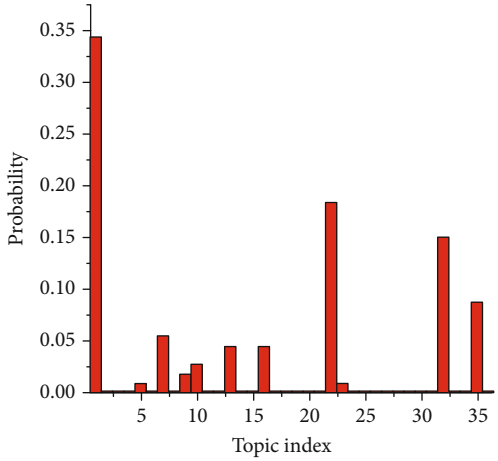


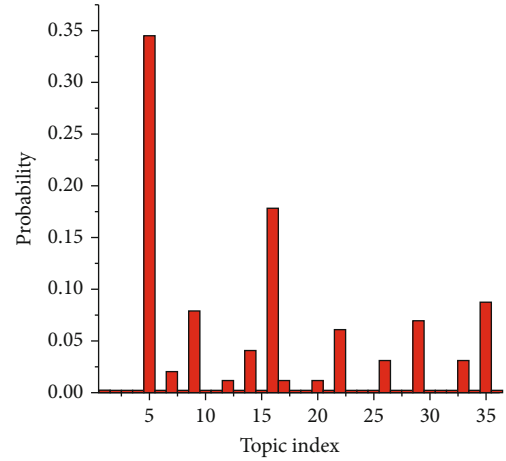
Figure 6: Robustness test experiment.

proposed LDA-based method is robust on the two datasets. And the LDA-based method always obtains smaller coherence values and performs better than k-means and DBSCAN.

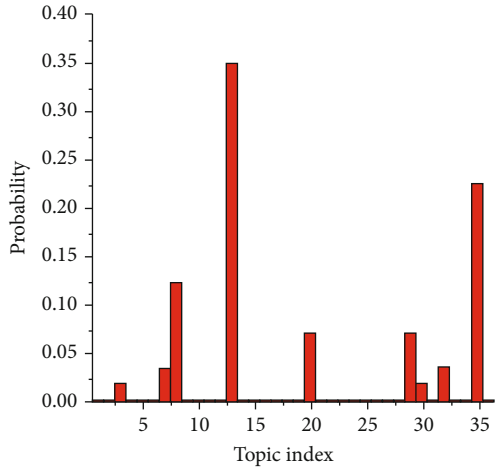
In the proposed model, we assign courses to clusters (topics) according to the course-topic distribution. A course is classified to the cluster corresponding to the maximum course-topic probability. We select one course from each cluster in Table 1 and illustrate their course-topic distribution in Figure 7. Figure 7 shows that the course–topic probabilities of the five courses are concentrated generally on one topic. For example, the probability of course "Building High-Performing Teams" belonging to topic (cluster) 29 is close to 50%, which is much bigger than the probabilities to other topics (clusters). Figure 7 indicates that the proposed LDA-based strategy can assign courses to the right clusters, which have a positive effect on the clustering results. In addition, clustering results in the k-means algorithm are affected significantly by the high-frequency words. The roles of the



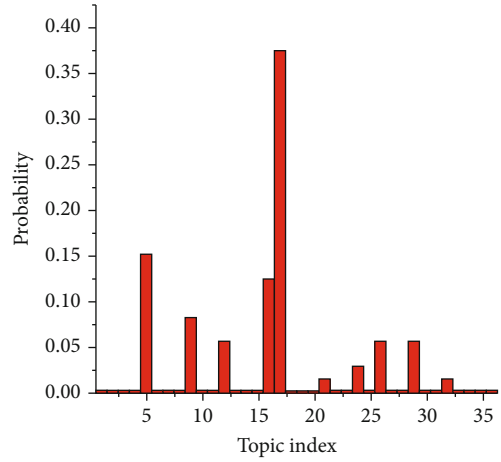
(a) Course Name: Blended Learning: Personalizing Education for Students



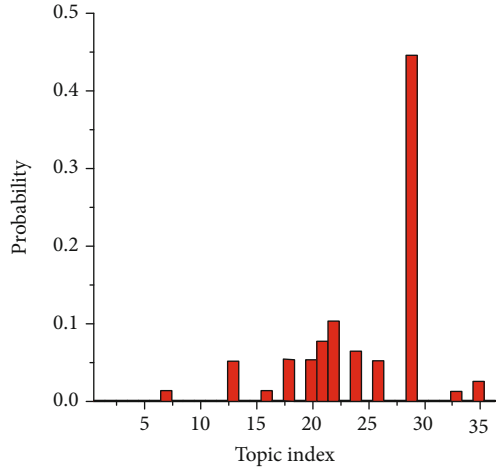
(b) Course Name: A Crash Course in Data Science



(c) Course Name: Advanced Business Strategy



(d) Course Name: Advanced R Programming



(e) Course Name: Building High-Performance Teams

Figure 7: Course-topic probability distribution.

nonhigh-frequency words, which indicate the teaching objectives, are likely to be weakened by the high-frequency words. In the proposed model, the courses are clustered according to topics rather than words. The latent topic strategy can smoothen the effects of the high-frequency words into multiple topics, thereby enabling us to obtain better clustering results than k-means.

4.3. Results on Course Ranking. Based on the course-topic (cluster) and the topic (cluster)-keyword distributions, we

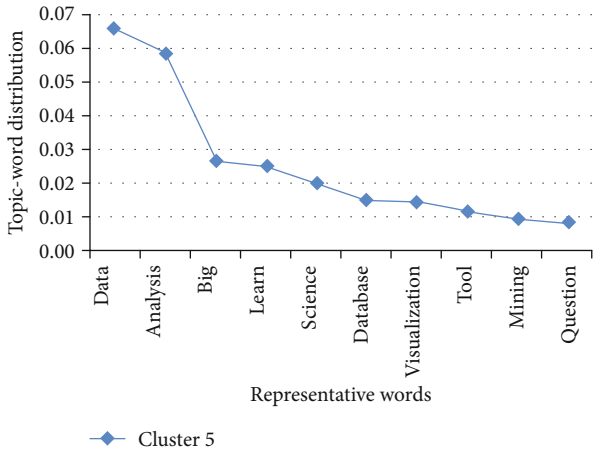


Figure 8: Representative words in Cluster 5 and their distribution probability.

Table 3: CourseRank values related to "Advanced Business Strategy.".

ID	Course name	CR value
1	Business growth strategy	0.029
2	Strategy formulation	0.027
3	Strategic management	0.026
4	Strategic organization design	0.021
5	Foundations of business strategy	0.019
6	Strategic planning and execution	0.017
7	Innovation and emerging technology: be disruptive	0.0166
8	(Re)-invent your business model with the odyssey 3.14 approach	0.015
9	Grow your business with Goldman Sachs 10,000 women	0.0145
10	Strategy implementation	0.0146

can optimize the course ranking task when students click on a specific course or search for a keyword through the search engine of a MOOC platform. If a student searches for a keyword through a search engine, we can filter out the topics unrelated to the keyword and list the courses in the related clusters according to the topic-keyword distribution. For example, if a student searches "Big data analysis," we can easily lock Cluster 5 as the target course cluster because its representative words are obviously related to "Big data analysis" (Figure 8). After locking the cluster, we can then show the representative courses in the cluster in the search list. In our experiment, courses such as "A Crash Course in Data Science" and "Applied Plotting, Charting & Data Representation in Python" belonging to Cluster 5 in Table 1 would be presented in the search list.

If a student clicks on a specific course in a MOOC platform, our experiment employs the CourseRank algorithm to rank the courses in the cluster where the clicked course belongs. For example, if a student clicks the course "Advanced Business Strategy," we employ the CourseRank algorithm to calculate the CourseRank value for each course in Cluster 13. The results are provided in Table 3. From Table 3, 10 courses are related to "Advanced Business Strategy," which is ranked in descending order by CourseRank values. These 10 courses together with the other courses would be shown in the recommendation lists of students who click on "Advanced Business Strategy." Similarly, if a student clicks on the course "Advanced Data Structures in Java" in Cluster 17, the proposed model would recommend the courses listed in Table 4 to the student.

Table 4: CourseRank values related to "Advanced Data Structures in Java.".

ID	Course name	CR value
1	Algorithms, part II	0.045
2	C++ for C programmers, part A	0.036
3	Algorithmic thinking (part 1)	0.031
4	C++ for C programmers, part B	0.015
5	Cloud computing concepts, part 1	0.0052
6	Algorithms, part I	0.0045
7	Data structures and performance	0.0042
8	Java programming: arrays, lists, and structured data	0.0035
9	Algorithmic thinking (part 2)	0.0034
10	Greedy algorithms, minimum spanning trees, and dynamic programming	0.0033

5. Conclusions

This paper proposed a novel search ranking method for MOOCs with the unstructured course descriptions and skills. The proposed model segments courses in the MOOC platforms into clusters based on course descriptions and ranks the courses in each cluster using course tags. This paper contributes theoretically to the educational research because we have introduced machine learning methods and employed new unstructured course information to deal with an important topic in the field.

Our experiments on the Coursera dataset showed that the proposed model can utilize the unstructured course description and skills efficiently to cluster courses and generate satisfactory search results. The experimental results indicated that the unstructured course descriptions and tags have rich information for MOOC services. Exploring the textual data using machine learning methods can help MOOC platforms improve recommendation accuracy. Figure 7 shows that a course usually provides knowledge across several education areas. Therefore, limiting a course to one education area would weaken service flexibility for MOOC platforms. The proposed models can help MOOC platforms position their courses accurately and improve their service qualities.

For future research, we will introduce more information to improve course ranking results. In this study, two kinds of unstructured data (i.e., course description and skills) were used to rank courses. In MOOC platforms, other kinds of data, such as word-of-mouth, can contain valuable information for the quality of courses. We will develop new search ranking models by considering these data. Another future

direction is to design methods to evaluate the search ranking results. Because we did not have the browsing logs of the search results, our experiment could not evaluate the accuracy of the obtained search ranking results. In the future, we will design subjective and objective strategies to test the effectiveness of the proposed method. The third direction is that many courses are missing learning skills in our study. Although we can infer the skills objectively from course contents and student reviews, new methods will be developed to infer course skills automatically.

Data Availability

Data are available for Requirement. Please send EMAIL to wqyao@ustc.edu.cn to obtain the data.

Ethical Approval

We have received approval from the ethics committee of Hefei University of Technology. We declare that no human participants were involved in this study.

Conflicts of Interest

We declare no conflict of interest concerning this study.

References

[1] R. F. Kizilcec, A. J. Saltarelli, J. Reich, and G. L. Cohen, "Closing global achievement gaps in moocs," *Science*, vol. 355, no. 6322, pp. 251-252, 2017.

[2] D. Shah, *By the Numbers: Moocs in 2018*, Class Central Moocreport, 2018, https://www.classcentral.com/report/mooc-stats-2018/.

[3] F. Bousbahi and H. Chorfi, "Mooc-rec: a case based recommender system for moocs," *Procedia-Social and Behavioral Sciences*, vol. 195, pp. 1813–1822, 2015.

[4] A. Elbadrawy and G. Karypis, "Domain-aware grade prediction and top-n course recommendation," in *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 183–190, Boston, MA, USA, 2016.

[5] M. Lin and D. W. Cheung, "An automatic approach for tagging web services using machine learning techniques1," in *Presented at Web Intelligence*, vol. 14, pp. 99–118, IOS Press, 2016.

[6] B. K. Daniel, "Big data and data science: a critical review of issues for educational research," *British Journal of Educational Technology*, vol. 50, no. 1, pp. 101–113, 2019.

[7] J. A. Ruipérez-Valiente, S. Halawa, R. Slama, and J. Reich, "Using multi-platform learning analytics to compare regional and global mooc learning in the Arab world," *Computers & Education*, vol. 146, p. 103776, 2020.

[8] R. Bodily, H. Leary, and R. E. West, "Research trends in instructional design and technology journals," *British Journal of Educational Technology*, vol. 50, no. 1, pp. 64–79, 2019.

[9] M. Zhu, A. Sari, and M. M. Lee, "A systematic review of research methods and topics of the empirical mooc literature (2014–2016)," *The Internet and Higher Education*, vol. 37, pp. 31–39, 2018.

[10] E. Costello, M. Brown, M. N. G. Mhichíl, and J. Zhang, "Big course small talk: twitter and moocs—a systematic review of research designs 2011–2017," *International Journal of Educational Technology in Higher Education*, vol. 15, no. 1, p. 44, 2018.

[11] M. Aparicio, T. Oliveira, F. Bacao, and M. Painho, "Gamification: a key determinant of massive open online course (mooc) success," *Information & Management*, vol. 56, no. 1, pp. 39–54, 2019.

[12] R. Deng, P. Benckendorff, and D. Gannaway, "Learner engagement in moocs: scale development and validation," *British Journal of Educational Technology*, vol. 51, no. 1, pp. 245–262, 2019.

[13] P. Luik, R. Suviste, M. Lepp et al., "What motivates enrolment in programming moocs?," *British Journal of Educational Technology*, vol. 50, no. 1, pp. 153–165, 2019.

[14] Y. Sun, L. Ni, Y. Zhao, X. L. Shen, and N. Wang, "Understanding students' engagement in moocs: an integration of self-determination theory and theory of relationship quality," *British Journal of Educational Technology*, vol. 50, no. 6, pp. 3156–3174, 2018.

[15] A. Cohen, U. Shimony, R. Nachmias, and T. Soffer, "Active learners' characterization in mooc forums and their generated knowledge," *British Journal of Educational Technology*, vol. 50, no. 1, pp. 177–198, 2019.

[16] N. Hood, A. Littlejohn, and C. Milligan, "Context counts: how learners' contexts influence learning in a mooc," *Computers & Education*, vol. 91, pp. 83–91, 2015.

[17] P. J. Guo and K. Reinecke, "Demographic differences in how students navigate through moocs," in *Presented at Proceedings of the first ACM conference on Learning@ scale conference*, pp. 21–30, Atlanta, GA, USA, 2014.

[18] K. Li, "Mooc learners' demographics, self-regulated learning strategy, perceived learning and satisfaction: a structural equation modeling approach," *Computers & Education*, vol. 132, pp. 16–30, 2019.

[19] J. Reich and J. A. Ruipérez-Valiente, "The mooc pivot," *Science*, vol. 363, no. 6423, pp. 130-131, 2019.

[20] B. Hong, Z. Wei, and Y. Yang, "Discovering learning behavior patterns to predict dropout in mooc," in *Presented at 2017 12th International Conference on Computer Science and Education (ICCSE)*, pp. 700–704, Houston, TX, USA, 2017.

[21] D. Peng and G. Aggarwal, "Modeling mooc dropouts," *Entropy*, vol. 10, pp. 1–5, 2015.

[22] A. Ramesh, D. Goldwasser, B. Huang, H. Daume, and L. Getoor, "Understanding mooc discussion forums using seeded lda," in *Presented at Proceedings of the ninth workshop on innovative use of NLP for building educational applications*, pp. 28–33, Baltimore, Maryland USA, 2014.

[23] T. Atapattu and K. Falkner, "A framework for topic generation and labeling from mooc discussions," in *Presented at Proceedings of the Third (2016) ACM conference on learning@ scale*, pp. 201–204, Edinburgh, Scotland, UK, 2016.

[24] Y. Pang, Y. Jin, Y. Zhang, and T. Zhu, "Collaborative filtering recommendation for mooc application," *Computer Applications in Engineering Education*, vol. 25, no. 1, pp. 120–128, 2017.

[25] C. Lang, R. Levy-Cohen, C. Woo et al., "Automated extraction of learning goals and objectives from syllabi using lda and neural nets," *Presented at Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, 2018.

[26] J. A. Hartigan and M. A. Wong, "Algorithm as 136: a k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, pp. 100–108, 1979.

[27] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Presented at Kdd*, vol. 96, pp. 226–231, 1996.

[28] R. I. Chang, Y. H. Hung, and C. F. Lin, "Survey of learning experiences and influence of learning style preferences on user intentions regarding MOOCs," *British Journal of Educational Technology*, vol. 46, no. 3, pp. 528–541, 2015.

[29] Y. Chen, Q. Chen, M. Zhao, S. Boyer, K. Veeramachaneni, and H. Qu, "Dropoutseer: visualizing learning patterns in massive open online courses for dropout reasoning and prediction," in *Presented at 2016 IEEE conference on visual analytics science and technology (VAST)*, pp. 111–120, Baltimore, MD, USA, 2016.

[30] R. R. Larson, "Introduction to information retrieval," *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 852-853, 2010.

[31] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Presented at Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, United Kingdom*, pp. 262–272, Edinburgh, Scotland, UK, 2011.

[32] S. Mankad, H. S. Han, J. Goh, and S. Gavirneni, "Understanding online hotel reviews through automated text analysis," *Service Science*, vol. 8, no. 2, pp. 124–138, 2016.