PS-LDA: A Course Item Model for Tutorial Personalized Recommendation

Yuefeng Du, Angzhi Liu, Xiaoguang Li(✉), and Baoyan Song

College of Information, Liaoning University, Shenyang 110036, China
{duyuefeng, liuangzhi, xgli, bysong}@lnu.edu.cn

Abstract. With the development of educational big data, personalized tutoring has become an important research direction to help people find interesting learning resources. However, due to limitation of learning resources, especially for the resource in unfamiliar subject areas, it may bring data sparseness of users' learning matrix. In this paper, we propose PS-LDA, a potential probability generation model for course item on learning preferences and subject area aware. By considering the mix of these two factors, our model provides personalized guidance for designated users. Moreover, we present a top-k method for online recommendation by matching the results from P-LDA and S-LDA. Finally, the experiments on two real-life datasets can verify the effectiveness and efficiency of our model.

Keywords: Personalized recommendation · Course item model · Online tutorial · Top-k recommendation

1 Introduction

With the widespread application in educational big data, varieties of online tutorial approaches have been proposed to acquire knowledge and skills, such as MOOC. Most of the existing learning systems have realized resource sharing, which helps users to study by resource categories. However, users may confuse their learning goals sometimes. In that case, it will lead to an inefficient guidance. Thus, personalized recommendations [1, 2] are required for capturing users' expectations.

Personal preferences and subject area are two factors of tutorial recommendations. Many researches [2–4] are existing for tutorial personalized recommendation. For example, the discussion on strategy behavior of teaching recommendation is based on the user's cognitive characteristics, cognitive style, learning motivation, personality structure characteristics, and personality type factor theories. Sarwar *et al*. [1] presented a method combining users' learning preferences and subject area aware. By establishing prediction model, LCARS [5] can analyze the relationship between personal preferences and hot topics. Unfortunately, it is difficult to make recommendations when users face to unfamiliar subject areas, even confusion. Hence, we focus on the perception of recommendation issues in different subject areas.

Note that, it brings a challenge to infer items from unfamiliar subject areas through using a user's historical learning data. CF (collaborative filtering) can make recommendation by tracing users' common interests. Usually, users only access a limited

G. Wang et al. (Eds.): WISA 2020, LNCS 12432, pp. 71–83, 2020.
https://doi.org/10.1007/978-3-030-60029-7_7

number of subject areas, this leads to data sparseness of users' learning preference matrix, even cold start to CF. In this case, it is not feasible to use only CF-based methods [6], especially when dealing with problems in unfamiliar subject areas, because query users often do not have sufficient activity history in their unfamiliar subject areas. To solve this problem, we propose a potential probability generation model PS-LDA, it consists of offline modeling and online recommendation. The offline model is designed to take into account the following two factors in a unified way at the same time. In fact, one has his own learning preference which can be obtained by trace his historical learning data. Besides, popular learning courses in various subject areas also attract one's interests. When users access a new subject area, especially unfamiliar subject areas, they are more likely to be interested in popular learning courses. Specifically, our model employs P-LDA to understand the user's learning preferences from the user's historical learning data. To pick the courses of subject areas aware, S-LDA utilizes subject-area-aware information from the subject areas. Next, given the query user u who visits the subject area s_u, the online recommendation of our model calculates the ranking score of each course item v within s_u by automatically combining u's learning preferences and s_u's popular courses. Thus, our model contributes to tutorial personalized recommendations both in one's own subject areas and unfamiliar subject areas.

The main contributions of our research are summarized as follows:

1) We propose a potential probability generation model PS-LDA. Specifically, P-LDA performs user topic modeling to obtain user learning preferences. S-LDA performs subject area topic modeling to obtain popular courses in the subject area. We also investigate the inference problem of our model.
2) We present a top-k method for personalized recommendations by matching the learning preferences and subject areas from the results of P-LDA and S-LDA.
3) We conducted experiments to evaluate the performance of our recommendation model on two real-life datasets. The results verified the effectiveness and efficiency of our model both in one's own subject areas and unfamiliar subject areas.

The rest of the paper is organized as follows: Sect. 2 reviews the related work. Section 3 details the model PS-LDA on learning preferences and subject area perception. Section 4 introduces the top-k method for online recommendation. The experimental results are reported in Sect. 5. The paper is summarized in Sect. 6.

2 Related Work

Recommender System. Collaborative filtering and content-based recommendation techniques are two widely applied methods for recommender systems. They can find relevant items according to the user's personal interests. Collaborative filtering [1, 6] automatically recommends related items to users by referencing item rating information from other similar users. The content-based recommendation [7] assumes that the descriptive characteristics of an item well reflect the user's preference for the item. Nevertheless, the data sparseness will affect CF, even cold start. It also brings limitation

to content-based recommendations. Therefore, a great deal of researches [8] were proposed on the advantages of combining both these approaches. Our recommendations focus on incorporating popular courses in subject areas.

Personalized Generation Model. Many models [9] were presented for obtaining and analyzing users' preferences. Yu *et al*. [11] used the content sentiment analysis to improve the performance of recommendation algorithm based on CF. Based on the LOM (Learning Object Meta-data), Mei *et al*. [10] modeled user interests and educational resources for online course recommendation. Apaza *et al*. [9] used the LDA (Latent Dirichlet Allocation) model to extract the features of online courses. Chen *et al*. [6] used cluster analysis and multiple linear regression models to recommend students' interest courses from their behavioral information such as attendance. However, it is lack of studies on the interaction between personal preferences and unfamiliar subject areas.

Our recommendation model differs from the above in the following three aspects. 1) We abstract a preference from user's historical learning records to match unfamiliar subject areas. 2) We analyze the popular courses to obtain the hot topic. 3) We propose a course item model mixed with personal preferences and subject area aware.

3 Personalized Generation Model

In this section, we first introduce the key data structures and symbols used in this paper. Then we propose PS-LDA on learning preferences and subject area awareness for personalized recommendations.

3.1 Problem Definition

To facilitate the following demonstration, we have defined the key data structures and symbols used in this article. Table 1 lists the relevant symbols used in this article.

Definition 1 Course Item. Course item v refers to a specific course in an access subject area.

Definition 2 User Learning. The user learning is a triple (u, v, s_v), which indicates that the user u selects the course item v in the subject area s_v.

Definition 3 User Learning Record. For each user u in dataset D, we create a user learning record D_u, which is a set of quaternions associated with u. We denote users, course items, subject areas and labels as $(u, v, s_v, c_v) \in D$, where $u \in \mathrm{U}$, $v \in \mathrm{V}$, $s_v \in \mathrm{S}$, $c_v \in C_v$. C_v represents the set of labels associated with the course item v. Note that, course items may contain multiple labels. For the learning record of the user activity, user u selects the course item v in s_v. Then we have a set of quaternions, which is $D_{uv} = \{(u, v, s_v, c_v): c_v \in C_v\}$. Obviously, $D_{uv} \subseteq D_u$.

Definition 4 Topic. A topic z in the course item set V is represented by the topic model ϕ_z, $\{P(v|\phi_z) : v \in \mathrm{V}\}$ or $\{\phi_{zv} : v \in \mathrm{V}\}$, which is the probability distribution of the geographic items. By analogy, the learning preference topic in the user set U is

represented by the label c_v in the user's historical learning record, and is represented by the topic model ϕ'_z, $\{P(c|\phi'_z) : c \in C\}$ or $\{\phi'_{zc} : c \in \mathrm{C}\}$, which is the probability distribution of the user's learning preferences. In summary, each topic z corresponds to two topic models in our work, namely ϕ_z and ϕ'_z.

Table 1. Definition of symbols.

Symbol	Description
N, V, M, C	The number of users, course items, subject areas, labels
U, V, S, C	The set of users, course items, subject areas, labels
V_s	The set of course items belong to subject areas s
s_v	The course item v of subject area s
c_v	The label describing course item v
K	The number of topics
D_u	The historical learning record of u
θ_u	The learning preferences of user u, expressed by a multinomial distribution over topics
θ'_s	The popular courses of subject Area s, expressed by a multinomial distribution over topics
ϕ_z	A multinomial distribution over course items specific to topic z
ϕ'_z	A multinomial distribution over labels specific to topic z
β, β'	Dirichlet priors to multinomial distributions ϕ_z, ϕ'_z
α, α'	Dirichlet priors to multinomial distributions θ_u, θ_s
λ_u	The mixing weight specific to user u
γ, γ'	Beta priors to generate λ_u

Definition 5 User Learning Preferences. The learning preference of user u is represented by θ_u, where θ_u is the probability distribution of the topic.

Definition 6 Popular Courses. Popular courses in subject area s are represented by θ'_s, the probability distribution of topics, which can mine popular courses in subject areas.

3.2 PS-LDA

The hybrid model considers the user's learning preferences and the influence of popular courses in a unified way. Given the querying user u and the visiting subject area s, the probability that user u chooses course item v when visiting the intersection of the subject area is sampled from the following model.

$$P(v|\theta_u, \theta'_{su}, \phi, \phi') = \lambda_u P(v|\theta_u, \phi, \phi') + (1 - \lambda_u) P(v|\theta'_{su}, \phi, \phi') \tag{1}$$

$P(v|\theta_u, \phi, \phi')$ is the probability of generating the curriculum item v based on learning preferences θ_u of u. And the process of generating $P(v|\theta_u, \phi, \phi')$ is denoted as P-LDA. $P(v|\theta'_{su}, \phi, \phi')$ is the probability of generating the curriculum item v according

to popular courses θ'_s in the subject area s. And the process of generating $P(v|\theta'_{su}, \phi, \phi')$ is denoted as S-LDA. λ_u is the parameter mixed weight for controlling the selection.

In order to further alleviate the problem of data sparseness, PS-LDA combines the label information of user history learning records. We redefine Eq. 1 as follows:

$$P(v|\theta_u, \theta'_{su}, \phi, \phi') = \sum\nolimits_{c \in C_v} P(v, c|\theta_u, \theta'_{su}, \phi, \phi') \tag{2}$$

$$P(v|\theta_u, \phi, \phi') = \sum\nolimits_{c \in C_v} P(v, c|\theta_u, \phi, \phi') \tag{3}$$

$$P(v|\theta'_{su}, \phi, \phi') = \sum\nolimits_{c \in C_v} P(v, c|\theta'_{su}, \phi, \phi') \tag{4}$$

Where C_v represents the set of labels associated with the course item v. In PS-LDA, users' learning interest θ_u and popular courses θ'_s are both modeled by polynomial distributions on potential topics. Each course item v is generated from a sample topic z. PS-LDA also parameterizes the distribution of labels associated with each topic z. So, z is responsible for generating course items and their labels at the same time.

$$P(v, c|\theta_u, \phi, \phi') = \sum_z P(v, c|z, \phi_z, \phi'_z) P(z|\theta_u) = \sum_z P(v|z, \phi_z) P(c|z, \phi'_z) P(z|\theta_u) \tag{5}$$

$$P(v, c|\theta'_{su}, \phi, \phi') = \sum_z P(v, c|z, \phi_z, \phi'_z) P(z|\theta'_{su}) = \sum_z P(v|z, \phi_z) P(c|z, \phi'_z) P(z|\theta'_{su}) \tag{6}$$

We assume that the course items and their labels are independent of the topic. $P(v, c|\theta_u, \phi, \phi')$ and $P(v, c|\theta'_{su}, \phi, \phi')$ are calculated according to formulas (5) and (6).

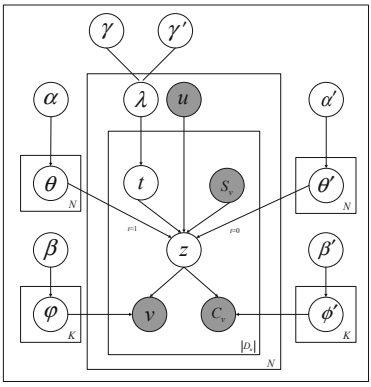


Fig. 1. Graphical representation of PS-LDA

By estimating the parameters of the PS-LDA model to obtain the topics of the course items and labels, this validates our prior knowledge that course items with many users. Otherwise, we cluster similar content into the same topic with high probability. Figure 1 illustrates the generation process with a graphical model. Algorithm 1 outlines the generation process, where Beta (.) is the Beta distribution. And γ, γ' are two of the parameters.

3.3 Model Inference

We use folded Gibbs sampling to obtain samples of hidden variable assignments, which helps to estimate unknown parameters $\{\theta, \theta', \phi, \phi', \lambda\}$ in PS-LDA. To simplify, we specify the hyperparameters α, α', β, β', γ, γ' with fixed values, *e.g.*, $\alpha = \alpha' = 50/K$, $\beta = \beta' = 0.01$, $\gamma = \gamma' = 0.5$. During the sampling process, we start with the joint probability of all user profiles in the dataset. Next, using the chain rule, we obtain the posterior probability of the sampled subject of each quadruplet (u, v, s_v, c_v). Specifically, we use a two-step Gibbs sampling procedure.

Algorithm 1. Probabilistic generative process in PS-LDA

Input: Topic z; User learning record D;
Output: Model parameters θ, θ', ϕ, ϕ' and λ;

For *each topic z* **do**
  Draw $\phi_z \sim Dirichlet(\cdot|\beta)$;
  Draw $\phi'_z \sim Dirichlet(\cdot|\beta')$;
End
For *each* D_u *in* D **do**
  For *each record* $(u, v_{ui}, s_{ui}, c_{ui}) \in D_u$
    Toss a coin t_{ui} according to $bernoulli(t_{ui}) \sim beta(\gamma, \gamma')$;
    If $t_{ui} = 1$ **then**
      Draw $\theta_u \sim Dirichlet(\cdot|\alpha)$;
      Draw a topic $z_{ui} \sim multi(\theta_u)$ according to the learning preference of user u;
    End
    If $t_{ui} = 0$ **then**
      Draw $\theta'_{s_{ui}} \sim Dirichlet(\cdot|\alpha')$;
      Draw a topic $z_{ui} \sim multi(\theta'_{s_{ui}})$ according to the popular courses of s;
    End
    Draw a course item $v_{ui} \sim multi(\phi_{z_{ui}})$ from z_{ui} - specific course item distribution;
    Draw a label $c_{ui} \sim multi(\phi'_{z_{ui}})$ from z_{ui} - specific label distribution;
  End
End

Due to space constraints, we only show the derived Gibbs sampling formula, omitting the detailed derivation process. We sample t based on the posterior probability as show in Eq. 7 and 8:

$$P(t_{ui}=1|t_{\neg ui},z,u,.)\propto\frac{n_{uz_{ui}}^{\neg ui}+\alpha_{z_{ui}}}{\sum_z(n_{uz}^{\neg ui}+\alpha_z)}\times\frac{n_{ut_1}^{\neg ui}+\gamma}{n_{ut_0}^{\neg ui}+n_{ut_1}^{\neg ui}+\gamma+\gamma'} \tag{7}$$

$$P(t_{ui}=0|t_{\neg ui},z,u,.)\propto\frac{n_{s_{ui}z_{ui}}^{\neg ui}+\alpha'_{z_{ui}}}{\sum_z(n_{s_{ui}z}^{\neg ui}+\alpha'_z)}\times\frac{n_{ut_0}^{\neg ui}+\gamma'}{n_{ut_0}^{\neg ui}+n_{ut_1}^{\neg ui}+\gamma+\gamma'} \tag{8}$$

Where n_{ut_1} is the number of times when $t = 1$ in the user profile D_u. So is n_{ut_0} when $t = 0$. n_{uz} is the number of times when the topic z is sampled from a polynomial distribution specific to user. n_{sz} is the number of times when the topic z is sampled in the polynomial distribution of subject area s. The number $n^{\neg ui}$ with a superscript $\neg ui$ indicates that it does not include the number of current instances.

For $t_{ui} = 1$ and $t_{ui} = 0$, we sample the topic z according to the following posterior probability as show in Eq. 9 and 10:

$$P(z_{ui}|t_{ui}=1,z_{\neg ui},v,c,u,.)\propto\frac{n_{uz_{ui}}^{\neg ui}+\alpha_{z_{ui}}}{\sum_z(n_{uz}^{\neg ui}+\alpha_z)}\frac{n_{z_{ui}v_{ui}}^{\neg ui}+\beta_{v_{ui}}}{\sum_v(n_{z_{ui}v}^{\neg ui}+\beta_v)}\frac{n_{z_{ui}c_{ui}}^{\neg ui}+\beta'_{c_{ui}}}{\sum_c(n_{z_{ui}c}^{\neg ui}+\beta'_c)} \tag{9}$$

$$P(z_{ui}|t_{ui}=0,z_{\neg ui},v,c,u,.)\propto\frac{n_{s_{ui}z_{ui}}^{\neg ui}+\alpha'_{z_{ui}}}{\sum_z(n_{s_{ui}z}^{\neg ui}+\alpha'_z)}\frac{n_{z_{ui}v_{ui}}^{\neg ui}+\beta_{v_{ui}}}{\sum_v(n_{z_{ui}v}^{\neg ui}+\beta_v)}\frac{n_{z_{ui}c_{ui}}^{\neg ui}+\beta'_{c_{ui}}}{\sum_c(n_{z_{ui}c}^{\neg ui}+\beta'_c)} \tag{10}$$

Where n_{zv} is the number of times the topic z generates a course term v. n_{zc} is the number of times the label c is sampled from the topic z.

After a sufficient number of sampling iterations, we can estimate the parameters $\theta, \theta', \phi, \phi'$ and λ as shown in Eq. 11 to 15:

$$\hat{\theta}_{uz}=\frac{n_{uz}+\alpha_z}{\sum_{z'}(n_{uz'}+\alpha_{z'})} \tag{11}$$

$$\hat{\theta}'_{sz}=\frac{n_{sz}+\alpha'_z}{\sum_{z'}(n_{sz'}+\alpha'_{z'})} \tag{12}$$

$$\hat{\phi}_{zv}=\frac{n_{zv}+\beta_v}{\sum_{v'}(n_{zv'}+\beta_{v'})} \tag{13}$$

$$\hat{\phi}'_{zc}=\frac{n_{zc}+\beta'_c}{\sum_{c'}(n_{zc'}+\beta'_{c'})} \tag{14}$$

$$\hat{\lambda}_u=\frac{n_{ut_1}+\gamma}{n_{ut_1}+n_{ut_0}+\gamma+\gamma'} \tag{15}$$

4 Top-K Online Recommendation

In our recommendation, we denote a two-parameter pair (u, s_u) as query task with query user u and subject area s_u. The result of the query is a sequential list of course items, which matches the user's learning preferences. After we infer PS-LDA model parameters θ_u, θ'_s, ϕ_z, ϕ'_z, λ_u during the offline modeling phase, the online recommendation section calculates the ranking of each course item v in the query subject area s_u Scores.

$$S(u, s_u, v) = \sum_z F(s_u, v, z) W(u, s_u, z) \tag{16}$$

$S(u, s_u, v)$ is the ranking framework in Eq. 16, which separates offline process from online process for scoring calculation. Specifically, $F(s_u, v, z)$ represents the offline score part for the course item v with respect to the subject area s_u in the dimension z. $F(s_u, v, z)$ is independent to query users. The weight score $W(u, s_u, z)$ is calculated in the online part to find expected weight of the query task (u, s_u).

$$W(u, s_u, z) = \hat{\lambda}_u \hat{\theta}_{uz} + (1 - \hat{\lambda}_u) \hat{\theta}'_{s_u z} \tag{17}$$

$$F(s_u, v, z) = \begin{cases} \hat{\phi}_{zv} \sum_{c_v \in C_v} \hat{\phi}'_{zc_v} & v \in V_{s_u} \\ 0 & v \notin V_{s_u} \end{cases} \tag{18}$$

The main time-consuming components of $W(u, s_u, z)$ are implemented offline. The online calculation can combine the processes shown in Eq. 17. In the process of querying, the offline score $F(s_u, v, z)$ needs to be aggregated in the K dimension by a simple weighted sum function from Eqs. 17 and 18. $W(u, s_u, z)$ is composed of two components, which are used to simulate user learning preferences and popular courses. Each component is associated with a user motivation. $F(s_u, v, z)$ concerns about similarities between the project co-occurrence information and the project content to generate recommendations.

5 Experiments

In this section, we conduct several experiments to compare the recommendation quality of our model.

5.1 Data Setting

Data Sets. We employ the two real-life datasets to evaluate the performance of our model on the course recommendation task.

EdX[1]. *EdX* is an online MOOC platform launched by Harvard and MIT. Users can learn the super-quality courses offered by these two famous schools on edX, covering different fields such as computer science, mathematics. EdX provides data on 290 Harvard and MIT online courses, 250 thousand certifications, 4.5 million participants, and 28 million participant hours since 2012.

GCSE[2]. Google Custom Search Engine (GCSE) is designed to retrieve LinkedIn profiles with the keyword "coursera". Overall, the dataset consists of 15,744 coursera MOOC entries for 5,668 professionals from LinkedIn.

Comparison Methods. We compare our proposed PS-LDA with the following five recommendation methods.

User-Topic Model (UT) [12]: This model is similar to the classic author-topic model (AT model) which assumes that topics are generated according to user interests. The probabilistic formula of the user topic model is presented as follows, where θ_B is a background for smoothing.$P(v|u;\Psi) = \lambda_B P(v|\theta_B) + (1-\lambda_B)\sum_z P(z|\theta_u)P(v|\phi_z)$.

Category-based k-Nearest Neighbors Algorithm (CKNN) [3]: CKNN projects a user's learning history into the category space and models user's learning preference using a weighted category hierarchy. When receiving a query, CKNN retrieves all the users and course items belong to the querying subject area. Then it applies a user-based CF method to predict the querying user rating of an unvisited course item. Note that the similarity between two users in CKNN is computed according to their weights in the category hierarchy, making CKNN a hybrid recommendation method.

Item-based k-Nearest Neighbors Algorithm (IKNN) [13]: This method utilizes the user's learning history to create a user-course item matrix. When receiving a query, IKNN retrieves all users to find k nearest neighbors by computing the Cosine similarity between two users' course item vectors. Finally, the course items in the user-specific querying subject area that have a relatively high ranking score will be recommended.

Learning Preference LDA (P-LDA): As a component of the proposed PS-LDA model, P-LDA means our method without exploiting the subject area information of course items. For online recommendation, the ranking score is computed by Eq. 16 with $F(s_u, v, z) = \hat{\phi}_{zv}\sum_{c_v \in C_v}\hat{\phi}'_{zc_v}$ and $W(u, s_u, z) = \hat{\theta}_{uz}$.

Subject Area Aware LDA (S-LDA): As another component of the PS-LDA model, S-LDA means our method without considering the content information of course items. For online recommendation, the ranking score is computed by Eq. 16 with $F(s_u, v, z) = \hat{\phi}_{zv}$ and $W(u, s_u, z) = \hat{\lambda}_u\hat{\theta}_{uz} + (1-\hat{\lambda}_u)\hat{\theta}'_{s_u z}$.

[1] https://www.edx.org/.

[2] https://www.gcse.com/.

5.2 Evaluation Methods and Indicators

To make an overall evaluation of the recommendation effectiveness of our proposed PS-LDA, we first design the following two real settings: 1) querying subject areas are new areas to querying users; 2) querying subject areas are familiar to querying users. We divide a user's learning history into a test set and a training set. And we adopt two different dividing strategies with respect to the two settings. For the first setting, we select all course items visited by the user in an unfamiliar subject area as the test set. The rest of the user's learning history is used as the training set. For the second setting, we randomly select 20% of course items visited by the user in familiar subject area as the test set. The rest of personal learning history is used as the training set. We split the user learning history D_u into the training data set $D_{training}$ and the test set D_{test}. To evaluate the recommender models, we adopt the testing methodology and the measurement Recall @k for each test case (u, v, s_v) in D_{test}.

1. We randomly select 1000 additional course in s_v and unrated by user u. We assume that most of them will not be of interest to user u.
2. We compute the ranking score for the test item v as well as the additional 1000 course items.
3. We form a ranked list by ordering all the 1001 course items according to their ranking scores. Let p denote the rank of the test item v within this list. The best result corresponds to the case where v precedes all the random items (*i.e.*, $p = 0$).
4. We form a top-k recommendation list by picking the top-k ranked items from the list. If $p < k$, we have a hit (*i.e.*, the test item v is recommended to the user). Otherwise, we have a miss. The probability of a hit increases with the increasing value of k. When $k = 1001$, we always have a hit.

The computation of Recall @k proceeds as follows. We set hit @$k = 1$ for a single test case if the test course item v appears in the top-k results. If not, hit @k will be set with 0. The overall Recall @k are defined by averaging all test cases.

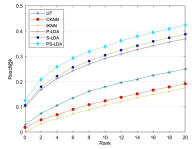
$$Recall@k = \frac{\#hit@k}{|D_{test}|} \tag{19}$$

Where #hit @k denotes the number of hits in the test set, and $|D_{test}|$ are all test cases.

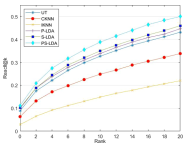
5.3 Experimental Results

Overall Performance. We first present the optimal performance with well-tuned parameters. And we also study the impact of model parameters. Figure 2 reports the performance of the recommendation algorithms on *EdX*. We show the performance where k is in the range from 1 to 20 since a greater value of k is usually ignored for a typical top-k recommendation task. It is apparent that the algorithms have significant performance disparity in terms of top-k recall. As shown in Fig. 2(a) where querying subject areas are new areas, the recall of PS-LDA is about 0.34 when $k = 10$ and 0.42

when $k = 20$ (*i.e.*, the model has a probability of 34% of placing an appealing event within the querying subject area in the top-10 and 42% of placing it in the top-20). Clearly, our proposed PS-LDA model outperforms other competitor recommendation methods. First, IKNN, CKNN and UT drop behind three other model-based methods, showing the advantage of using latent topic models to model users' preferences. Second, PS-LDA outperforms both P-LDA and S-LDA, showing the advantages of combining learning preferences and subject area in a unified manner.

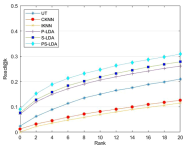


(a) Users learning in new subject areas

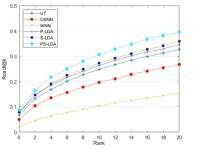


(b) User learning in familiar subject areas

Fig. 2. Top-k performance on EdX



(a) Users learning in new subject areas

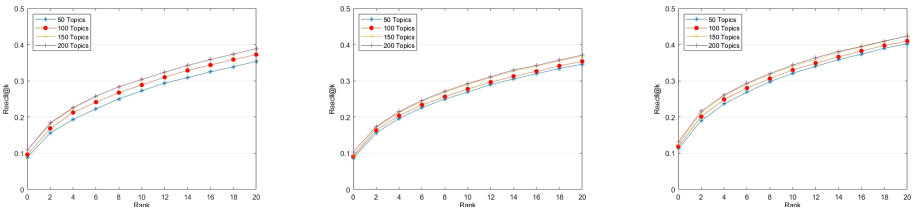


(b) User learning in familiar subject areas

Fig. 3. Top-k performance on GCSE

In Fig. 2(b), we report the performance of all recommendation algorithms for the second setting where querying subject areas are familiar to querying users. We can see that the trend of comparison result is similar to that presented in Fig. 2(a). The main difference is that CKNN outperforms IKNN in Fig. 2(a) while IKNN exceeds CKNN significantly in Fig. 2(b). It shows that the CF-based method (*i.e.*, IKNN) better suits the setting if the user-item matrix is not very sparse. The hybrid method (*i.e.*, CKNN) is more capable of overcoming the difficulty of data sparseness, *e.g.*, the new subject area problem. Another observation is that UT almost performs as well as PS-LDA, and outperforms CKNN and IKNN in the familiar subject area setting, verifying the benefit brought with the subject area influence. However, UT is still less effective than PS-LDA under this setting. Furthermore, the performance of UT is poor in the new subject area setting, as shown in Fig. 2(a), which shows that exploiting subject area influence cannot alleviate the new subject area problem since there is no learning history of the querying user in the new subject area.

Figure 3 reports the performance of the recommendation algorithms on the GCSE dataset. We compare PS-LDA with UT, CKNN, IKNN, P-LDA and S-LDA. From the figure, we can see that the trend of comparison result is similar to that presented in Fig. 2, and PS-LDA performs best.



(a) P-LDA (b) S-LDA (c) PS-LDA

Fig. 4. Impact of the number of latent topics

Impact of Model Parameters. Tuning model parameters, such as the number of topics for all topic models, is critical to the performance of models. We therefore also study the impact of model parameters on *EdX* dataset. Because of space limitations, we only show the experimental results for the new subject area setting. As for the hyperparameters α, α', β, β', γ and γ', following the existing works [2], we empirically set fixed values (*i.e.*, $\alpha = \alpha' = 50/K$, $\beta = \beta' = 0.01$, $\gamma = \gamma' = 0.5$). We tried different setups and found that the estimated topic models are not sensitive to the hyperparameters. But the performances of the topic models are slightly sensitive to the number of topics. Thus, we tested the performance of P-LDA, S-LDA and PS-LDA models by varying the number of topics shown in Figs. 4(a) to 4(c). From the results, we observe 1) the Recall @k values of all latent topic-based recommender models slightly increase with the increasing number of topics. 2) The performance of latent topic-based recommender models does not change significantly when the number of topics is larger than 150. 3) P-LDA, S-LDA and PS-LDA perform better under any number of topics, and PS-LDA consistently performs best.

6 Conclusion

This paper proposed a personalized recommendation, PS-LDA, which can facilitate people's study not only in their familiar subject area but also in a new area where they have no learning history. By taking advantage of both the content and subject area information of course items, our system overcomes the data sparsity problem in the original user-item matrix. We evaluated our system using extensive experiments based on two real-life datasets. According to the experimental results, our approach significantly outperforms existing recommendation methods in effectiveness. The results also justify each component proposed in our system, such as taking learning preferences and subject area information into account.

Acknowledgement. This research was supported by the Joint Funds of the National Natural Science Foundation of China under Grant No. U1811261, the Project of Liaoning Provincial Public Opinion and Network Security Big Data System Engineering Laboratory.

References

1. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM (2001). https://doi.org/10.1145/371920.372071
2. Du, M., Christensen, R., Zhang, W., Li, F.: Pcard: personalized restaurants recommendation from card payment transaction records. In: Proceedings of the 28th International Conference on World Wide Web, pp. 951–958. ACM (2019)
3. Bao, J., Zheng, Y., Mokbel, M.F.: Location-based and preference-aware recommendation using sparse geo-social networking data. In: Proceedings of the 20th International Conference on Advances in Geographic Information Systems, pp. 199–208. ACM (2012)
4. Fan, J., et al.: Octopus: an online topic-aware influence analysis system for social networks. In: Proceedings of the 34th International Conference on Data Engineering, pp. 1569–1572. IEEE (2018)
5. Yin, H., Sun, Y., Cui, B., Hu, Z., Chen, L.: LCARS: a location-content-aware recommender system. In: Proceedings of the 19th International Conference on Knowledge Discovery and Data Mining, pp. 221–229. ACM (2013)
6. Chen, W., Chu, J., Luan, J., Bai, H., Wang, Y., Chang, E.Y.: Collaborative filtering for orkut communities: discovery of user latent behavior. In: Proceedings of the 18th International Conference on World Wide Web, pp. 681–690. ACM (2009)
7. Hou, Y., Zhou, P., Wang, T., Yu, L., Hu, Y., Wu, D.: Context-aware online learning for course recommendation of MOOC big data. arXiv preprint arXiv:1610.03147 (2016)
8. Popescul, A., Ungar, L.H., Pennock, D.M., Lawrence, S.: Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. arXiv preprint arXiv:1301.2303 (2013)
9. Apaza, R.G., Cervantes, E.V., Quispe, L.C., Luna, J.O.: Online courses recommendation based on LDA. In: Proceedings of the 1st Symposium on Information Management and Big Data, pp. 42–48. CEUR (2014)
10. Mei, L., He, J., Liu, H., Du, X.: Latent path connected space model for recommendation. In: Shao, J., Yiu, M.L., Toyoda, M., Zhang, D., Wang, W., Cui, B. (eds.) APWeb-WAIM 2019. LNCS, vol. 11642, pp. 163–172. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26075-0_13
11. Yu, J., An, Y., Xu, T., Gao, J., Zhao, M., Yu, M.: Product recommendation method based on sentiment analysis. In: Meng, X., Li, R., Wang, K., Niu, B., Wang, X., Zhao, G. (eds.) WISA 2018. LNCS, vol. 11242, pp. 488–495. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-02934-0_45
12. Takeuchi, Y., Sugimoto, M.: CityVoyager: an outdoor recommendation system based on user location history. In: Ma, J., Jin, H., Yang, L.T., Tsai, J.J.-P. (eds.) UIC 2006. LNCS, vol. 4159, pp. 625–636. Springer, Heidelberg (2006). https://doi.org/10.1007/11833529_64
13. Song, Y., Huang, J., Zhou, D., Zha, H., Giles, C.L.: IKNN: informative k-nearest neighbor pattern classification. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 248–264. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74976-9_25