Forum Thread Recommendation for Massive Open Online Courses

Diyi Yang, Mario Piergallini, Iris Howley, Carolyn Rose
Carnegie Mellon University
Language Technologies Institute
5000 Forbes, Pittsburgh, PA
{diyiy, mpiergal,ihowley, cprose}@cs.cmu.edu

ABSTRACT

Recently, Massive Open Online Courses (MOOCs) have garnered a high level of interest in the media. With larger and larger numbers of students participating in each course, finding useful and informative threads in increasingly crowded course discussion forums becomes a challenging issue for students. In this work, we address this thread overload problem by taking advantage of an adaptive feature-based matrix factorization framework to make thread recommendations. A key component of our approach is a feature space design that effectively characterizes student behaviors in the forum in order to match threads and users. This effort includes content level modeling, social peer connections, and other forum activities. The results from our experiment conducted on one MOOC course show promise that our thread recommendation method has potential to direct students to threads they might be interested in.

Keywords

Thread Recommendation, Massive Open Online Courses, Matrix Factorization

1. INTRODUCTION

Massive Open Online Courses (MOOCs) have rapidly moved into a place of prominence in the media in recent years. MOOC platforms, such as Coursera[1] and EdX[2], are faced with course registration and participation in the hundreds of thousands, and potentially have even larger student populations. A novel component of online learning courses is the use of interactive discussion forums where instructors and students can ask questions, discuss ideas, provide help to or even socialize with other students. As class sizes grow, the number of threads per course forum increases rapidly. Consequently, it becomes more difficult for students to find what they are looking for or truly interested in.

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

One solution to this problem would be to give each student a short list of threads that we believe would interest them. If a forum system could automatically detect user interests to generate personalized thread recommendations, it would make it much faster and more convenient for students to find the threads they want to participate in. Additionally, this would decrease the amount of time that new questions go unanswered by directing appropriate users there. A student's potential interest in a thread is largely determined by the match between the student's preferences and the content focus of the thread.

In this work, we propose a model for thread recommendation in MOOC discussion forums that addresses issues caused by massive thread volumes and help students to answer their fellow students' questions more quickly. To do this, first we use content level indicators of threads students have participated in to capture their preferences over threads, which helps to do latent matching between threads and students' interests; then we design an adaptive time window matrix factorization model to take students' behavior in the current time window and predict their behavior in the following time window; finally, we conduct experiments on one Coursera[3] course, and demonstrate that our model gives significantly improved performance over several baselines. Quantitative analysis, including exploration of differing window sizes, is provided to validate our approach.

2. RELATED WORK

Considerable prior research on MOOCs has focused on concerns related to activities in MOOCs apart from discussion, such as watching videos, peer grading [11], and dropout [6]. Previous work in a variety of other contexts [2] explores student activities in discussion forums. In that work, the underlying hypothesis is that participation in learning-related activities such as discussing and sharing, could have a positive influence on knowledge gain [3]. Neo-Piagetian theory on collaborative learning suggests that discussion provides opportunities to experience cognitive conflict, which potentially produces learning [10]. In a classroom setting, Wood et al. [14] explored how learner and tutor interaction influence learning outcomes, which further argues for a relationship between participation in discussion and students' learning. Thus, discussion participation is an important activity to support as it is another source of knowledge and learning within a MOOC.

[3]https://www.coursera.org/

In terms of the context of MOOCs, where the interaction with or guidance from instructors are limited, and dropout in these massively enrolled environments is very high, it becomes more necessary to improve the participation and engagement of students in the course [9]. One direct indication of students' commitment in MOOCs is their activities in the discussion forum. Those discussion threads focus on questions and confusion about lectures, including clarification requests about assignments or exams. Other times they are off-topic or just socializing [7]. Finding an earlier thread that answers one's questions or applies to one's interests among such a large set of threads is challenging. This becomes even worse for students who created threads seeking help since their potential helpers may simply not find them [13]. Thus, thread recommendation (i.e., the production of a short list of potentially interesting threads) has great potential for increasing the value and approachability of MOOC discussion forums.

Existing work in question recommendation mainly focuses on online discussion forums such as Yahoo! Answers [12]. For instance, the work by Hu et al. [8] introduced a user modeling method that estimates the interests and professional areas of each user in order to generate a suitable user set to answer a given question. However, MOOC discussion forums frequently lack the rich information that generic online forums have, such as user reputation (which would be important here because threads of a highly reputed person is more likely to attract others' attentions) [1]. Besides, students in a MOOC forum differ from common users of more typical types of web forums, since their length of participation is typically only around eight weeks in the forum and most students choose to drop out as time proceeds [15]. Thus, research is needed to determine how best to take advantage of expertise in MOOC forums so that the thread recommendation problem is solvable. To the best knowledge of the authors, this is the first work on thread recommendation in MOOC forums.

3. THREAD RECOMMENDATION

We introduce the adaptive feature-based matrix factorization (MF) framework that we use to recommend threads to students in this section. To begin with, we describe the adaptive MF framework, then we explain how we incorporate content level modeling, social peer connection and other contextual information into our framework.

3.1 Adaptive Matrix Factorization

Classical matrix factorization (MF) [5] could address the thread recommendation problem efficiently. MF constructs a reduced representation that mediates some feature based representation of users and threads. That representation can then be used to match users with appropriate threads. However, different from traditional product recommendation, for MOOC thread recommendation one important property is that each time a student logs into the forum, they are more likely to participate in threads that were posted more recently. New threads are more likely to be relevant to the current subject in the course while old threads may be irrelevant to them. Taking advantage of both the MOOC property and state-of-art matrix factorization framework, we propose an adaptive matrix factorization model. We illustrate how we design this adaptive model in two steps as follows.

In the first step, we give a detailed formulation of the feature based matrix factorization. Formally, suppose we have three feature sets G, M, and N called global features, student features and thread features, respectively. α, β, and γ are the extracted feature values. α is for global features, β is for student features, and γ stands for thread features. Then, for each record user, thread, and participation or not indicator, $< u, t, r_{u,t} >$, the predicted score $\hat{r}_{u,t}$ is defined as follows (p_u and q_t are latent vectors associated with users and threads):

$$\begin{aligned}\hat{r}_{u,t} = \quad & \mu + (\textstyle\sum_{g\in G}\gamma_g b_g + \sum_{m\in M}\alpha_m b_m^u + \sum_{n\in N}\beta_n b_n^t) \\ & + \textstyle\sum_{m\in M}\alpha_m {p_m}^T \sum_{n\in N}\beta_n q_n\end{aligned} \tag{1}$$

The global features are used to incorporate information which is related to all students and threads, i.e. tendencies that hold for the entire forum. Meanwhile, student features and thread features can capture the information related only to specific students or threads. When the indicators of student and thread are the only student and thread features without any other global features, this feature-based matrix factorization model naturally degenerates to classical matrix factorization. This matrix factorization framework gives us the ability to incorporate as many features as desired.

In the second step, we elaborate how we adapt the basic framework into the adaptive model. Firstly, we define a time window of size Δ that moves along the course weeks. In order to recommend threads to students in week w, our feature-based matrix factorization will be trained only on the data between time $w - \Delta$ and $w - 1$. If $w \leq \Delta$, only the data between week 1 and week $w-1$ is utilized. Additionally, the candidates for recommendation are only active threads, which are threads that were posted or received at least one reply during the time window. Since the time window slides across the course period, the performance of our model can be evaluated by averaging the performances of each week.

3.2 Contextual Modeling

In this section, we present several contextual aspects that we incorporate into the adaptive feature-based matrix factorization framework.

Content Level Modeling: We assume that students' preferences over threads are approximately equivalent to their preferences for the contents of those threads. To exploit the content of the thread to do the latent matching between threads and the interests of students, we represent the content of the thread as a bag-of-words [16]. Thus, we can transform the problem into whether the student is interested in the words or topics in the thread, rather than the thread itself. Intuitively, a thread question t consists of a set of words $W(t)$ out of the entire word set Z. This content level modeling is formulated as follows:

$$\hat{r}_{u,t} = bias + {p_u}^T(q_t + \frac{\sum_{w\in W(t)}\phi_w}{|W(t)|}) \tag{2}$$

$bias$ is some constant representing generalized possible biases and $|W(t)|$ is the number of word contained in thread t. ϕ_w captures the influence of word w on students.

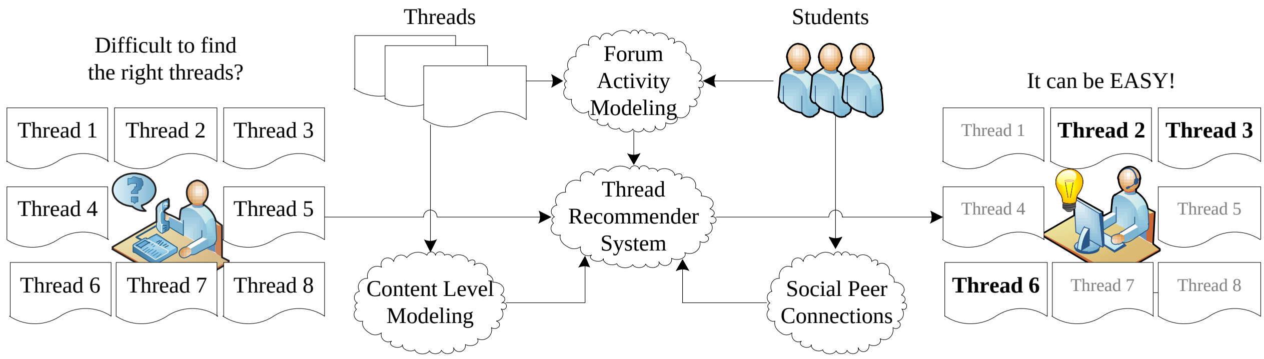

Figure 1: Why do we need a thread recommendation system in MOOCs? For students, it is hard to find which threads among thousands they want to participate. Our designed Thread Recommender System, can fully leverage these features and provide accurate recommendations.

Social Peer Connections: Students who interact frequently in the forum share similar engagement conditions and similar learning interests toward the course. Here, we use the connections between a student and their close peers $S(u)$ to capture peers' influence on students. We define the close peers as the top students who have the most interaction occurrences with student u based on replies. This peer influence could be characterized as follows. (φ_v models the influence ability of the student v as a peer on other students.)

$$\tilde{r}_{u,t} = bias + (p_u + \frac{\sum_{v \in S(u)} \varphi_v}{\sqrt{|S(u)|}})^T q_t \qquad (3)$$

Forum Activity Modeling: For student related features, the number of different threads students participated in before the current week (**Thread Count**) shows their historical participation level, while the number of posts made in the previous week (**Previous Count**) is recorded to reflect their recent activity level. **Cohort** representing when this student registered for the course can be treated as a proxy for the level of motivation(i.e. early course registration indicates initiative motivation). The number of replies and comments **Reply Number** is counted as a thread feature. **Thread Length** representing the total number of words appearing in the thread is also computed.

4. EXPERIMENTS

In this section, we introduce the dataset, experiment setup and baselines. Then we discuss our experimental results along with the adaptive window size exploration.

4.1 Experiment Setup

We conducted our experiments on one Coursera course, 'Learn to Program: The Fundamentals', shortened to 'Python Course'. It has 3590 active students who have at least one post in the forum and 3079 threads across around eight weeks, based on which we performed the time window evaluation. For each thread, we have its replies and comments; threads' contents and students' registration time are also available. Mean Average Precision (MAP) [4] is our evaluation metric. Specifically, we use MAP@1, MAP@3, and MAP@5 to evaluate the performance. Our analysis is limited to only behaviors within the discussion forums.

To make our analysis clear and concise, we define some notation here. Content level modeling is denoted as C; we denote social peer connections as P; the student related features are S, and thread related features as T. Specially, we use *All* to denote the integration of all aspects of the features. We empirically set the size of the time window as 2 weeks. That is, when we predict the preferences of students over threads in week 2, only the data in week 1 is used to train the model; likewise, to make the prediction in week 5, the forum history in week 4 is used.

Baselines used in this work include **Popularity (PPL)** which conducts thread recommendations based on thread popularity, **Directly Content Match (DCM)** which recommends threads based on how similar the content of the thread is to the post history of the student, and **Classical Matrix Factorization (MF)** that maps students and threads into the same latent space without contextual information. Our proposed **Adaptive Matrix Factorization (AMF)** utilizes the rich contextual information via encoding different information into its feature space. We use the notation AMF-{All} to describe a model using all types of features. AMF-{C} means that only content features are used in the AMF model.

4.2 Recommendation Performance

In this section, we present the recommendation results from one MOOC. Based on the results of different models shown in Table 1, we could observe that the DCM is the worst among all models. The performance of the MF model is at least 0.02 higher than the PPL model regarding to MAP@1, MAP@3 and MAP@5. The series of AMF models, which contain each aspect of our designed four aspect features, has better performance over PPL and MF. This demonstrates that each type of feature makes an important contribution in capturing the latent matching between interest of students and the topics involved in threads. One notable point is that the Student Forum Activity Modeling feature set is better than any of the other single feature dimensions. Fully combining all types of features makes the best model, which indicates that the four types of features capture different aspects of modeling of the latent matching between students and threads.

Method	MAP@1	MAP@3	MAP@5
PPL	0.154	0.254	0.307
DCM	0.092	0.198	0.172
MF	0.171	0.280	0.332
AMF-{C}	0.177	0.282	0.340
AMF-{P}	0.178	0.286	0.340
AMF-{S}	0.183	0.290	0.341
AMF-{T}	0.174	0.280	0.334
AMF-{All}	**0.198**	**0.323**	**0.376**

Table 1: Average Results on Python Course

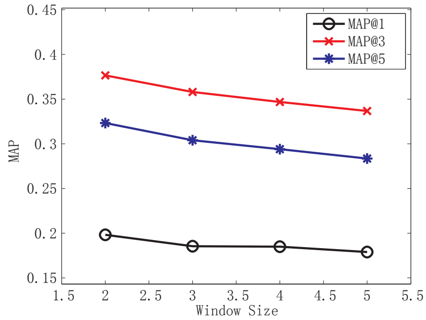

Figure 2: Window Size Exploration

4.3 Window Size Exploration

One important parameter in our adaptive MF model is the window size. We explained that we chose two weeks as the proper time window size. In this section, we describe how we tune this parameter and how the recommendation performance changes as window size increases. We only test how the performance of the best model AMF-{All} changes as window size changes. The results of the MAP changing curve is presented in Figure 2. We can observe that AMF-{All}'s performance is always decreasing as the window size increases. This makes sense because when students log into the forum system, they are more likely to pay attention to recent threads. In conclusion, students' activities in very recent weeks are more predictive of their participation in the later week. The smaller the window size, the better the thread recommendation performance.

5. CONCLUSION AND FUTURE WORK

In this work, we created a thread recommendation system for MOOC discussion forums in order to improve the learning experience of students. For this purpose, we proposed an adaptive matrix factorization framework to capture the affinity of students for threads; then we integrated content-level modeling, social peer connections, as well as measures of students' overall forum activities into that framework. Experiments conducted on the MOOC dataset show that our proposed model significantly outperforms several baselines. In the future, we plan to conduct some deployed studies in active MOOCs to validate our framework.

Acknowledgement

This research was supported in part by NSF grants OMA-0836012 and IIS-1320064.

6. REFERENCES

[1] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: A case study of stack overflow. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 850–858, New York, NY, USA, 2012. ACM.

[2] G. Balakrishnan. Predicting student retention in massive open online courses using hidden markov models. Master's thesis, EECS Department, University of California, Berkeley, May 2013.

[3] R. M. Carini, G. D. Kuh, and S. P. Klein. Student engagement and student learning: Testing the linkages*. *Research in Higher Education*, 47(1):1–32, 2006.

[4] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu. Collaborative personalized tweet recommendation. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 661–670, New York, NY, USA, 2012. ACM.

[5] T. Chen, W. Zhang, Q. Lu, K. Chen, Z. Zheng, and Y. Yu. Svdfeature: a toolkit for feature-based collaborative filtering. *The Journal of Machine Learning Research*, 13(1):3619–3622, 2012.

[6] D. Clow. Moocs and the funnel of participation. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge*, pages 185–189. ACM, 2013.

[7] S. S. Glenda, D. Jennifer, W. Jonathan, and B. Lori. Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses. In *Workshop on Data Driven Education, Advances in Neural Information Processing Systems 2013*, 2013.

[8] D. Hu, S. GU, S. Wang, L. Wenyin, and E. Chen. Question recommendation for user-interactive question answering systems. In *Proceedings of the 2Nd International Conference on Ubiquitous Information Management and Communication*, ICUIMC '08, pages 39–44, New York, NY, USA, 2008. ACM.

[9] R. Junco. The relationship between frequency of facebook use, participation in facebook activities, and student engagement. *Computers and Education*, 58(1):162–171, 2012.

[10] J. Piaget. The equilibrium of cognitive structures: the central problem of intellectual development. *Human development*, 1985.

[11] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng, and D. Koller. Tuned models of peer assessment in MOOCs. In *Proceedings of The 6th International Conference on Educational Data Mining (EDM 2013)*, 2013.

[12] M. Qu, G. Qiu, X. He, C. Zhang, H. Wu, J. Bu, and C. Chen. Probabilistic question recommendation for question answering communities. In *Proceedings of the 18th international conference on World wide web*, pages 1229–1230. ACM, 2009.

[13] I. Roll, V. Aleven, B. M. McLaren, and K. R. Koedinger. Improving students help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, 21(2):267–280, 2011.

[14] H. Wood and D. Wood. Help seeking, learning and contingent tutoring. *Computers and Education*, 33(2):153–169, 1999.

[15] D. Yang, T. Sinha, D. Adamson, and C. P. Rose. Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses. In *Workshop on Data Driven Education, Advances in Neural Information Processing Systems 2013*, 2013.

[16] Y. Zhang, R. Jin, and Z.-H. Zhou. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52, 2010.