Improving Latent User Models in Online Social Media

Adit Krishnan
Dept of Computer Science, University of Illinois at Urbana-Champaign
aditk2@illinois.edu

Ashish Sharma
Dept of Computer Science, Indian Institute of Technology Kharagpur
ashish4@illinois.edu

Hari Sundaram
Dept of Computer Science, University of Illinois at Urbana-Champaign
hs1@illinois.edu

ABSTRACT

Modern social platforms are characterized by the presence of rich user-behavior data associated with the publication, sharing and consumption of textual content. Users interact with content and with each other in a complex and dynamic social environment while simultaneously evolving over time. In order to effectively characterize users and predict their future behavior in such a setting, it is necessary to overcome several challenges. Content heterogeneity and temporal inconsistency of behavior data result in severe sparsity at the user level. In this paper, we propose a novel mutual-enhancement framework to simultaneously partition and learn latent activity profiles of users. We propose a flexible user partitioning approach to effectively discover rare behaviors and tackle user-level sparsity.

We extensively evaluate the proposed framework on massive datasets from real-world platforms including Q&A networks and interactive online courses (MOOCs). Our results indicate significant gains over state-of-the-art behavior models (15% avg) in a varied range of tasks and our gains are further magnified for users with limited interaction data. The proposed algorithms are amenable to parallelization, scale linearly in the size of datasets, and provide flexibility to model diverse facets of user behavior.

KEYWORDS

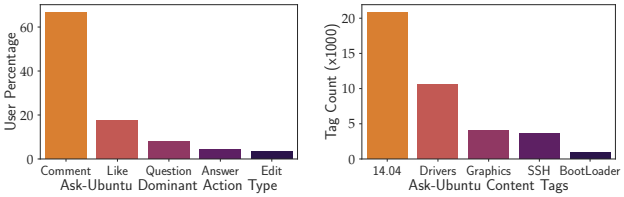
Behavior Models; Latent Factor Models; Knowledge Exchange Networks; MOOCs; Data Skew

1 INTRODUCTION

This paper addresses the problem of developing robust statistical representations of participant behavior and engagement in online knowledge-exchange networks. Examples of knowledge-exchange networks include interactive MOOCs (Massive Online Open Courses), where participants interact with lecture content and peers via course forums, and community Q & A platforms such as Stack-Exchanges[1]. We would like statistical representations to provide insight into dominant behavior distributions, and understand how an individual's behavior evolves over time. Understanding and profiling time-evolving participant behavior is important in several applications; for instance, pro-actively identifying unsatisfactory student progress in MOOCs may lead to redesign of the online learning experience. The dynamic nature and diversity of user activity in such learning environments pose several challenges to profiling and predicting behavior.

Behavior skew poses a significant challenge in identifying informative patterns of user engagement with interactive social-media

[1]https://stackexchange.com



(a) (b)

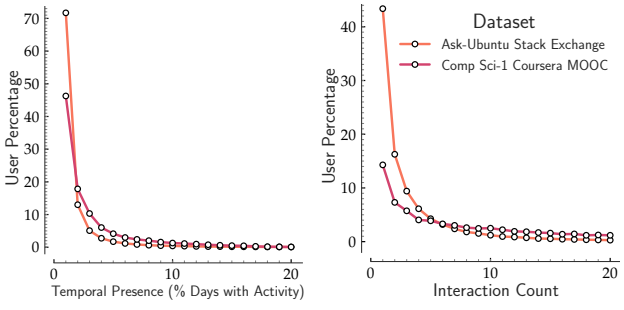
Figure 1: Illustration of (a) Dominant Action Skew and (b) Content Skew in our largest Stack-Exchange, Ask-Ubuntu

[9, 33]. In the Ask-Ubuntu[2] Q&A network, most users primarily engage by commenting on posts as seen in Figure 1(a). Subject experts who invest most of their time editing or answering questions are relatively infrequent. The extent of behavior skew is compounded by the presence of popular and niche subject areas (Figure 1(b)). For instance, users who comment on popular topics vastly outnumber those who edit or answer questions on niche subject areas.

Inconsistent participation and user-level data sparsity are other prominent challenges in most social media platforms [3, 13, 36]. In Community Q&A websites and MOOCs, a minority of participants dominate activity in a classic power-law distribution [3] as observed in Figure 2(a). Additionally, an overwhelming majority of users record activity on less than 10% days of observation in our datasets (Figure 2(b)). Temporal inconsistency in user participation renders evolutionary user modeling approaches [28, 29] ineffective for sparse or bursty participants in social learning platforms.

Despite several relevant lines of work, including user evolution modeling [28], behavior factorization [48], sparsity-aware tensor factorization [13] and contextual text mining [29], there are few systematic studies addressing these pervasive challenges in modeling user behavior across diverse platforms and applications. Evolutionary activity sequence based user modeling approaches [28] do not explicitly account for sparse or bursty users, and are best suited to temporally consistent user activity.

[2]https://askubuntu.com/



(a) (b)

Figure 2: Temporal Consistency and User Interaction Volume ($\eta \approx 3$) are highly skewed in Stack-Exchange/Coursera

Matrix factorization methods have been adapted to extract dynamic user representations and account for evolution of user interests. Jiang et al [13] develop a multi-facet tensor factorization approach for evolutionary analysis, employing facet-wise regularizer matrices to tackle sparsity. [48] discovers topical interest profiles via simultaneous factorization of action-specific matrices. Quadratic scaling imposes restrictive computational limits on these methods. Generative behavior models are controlled by introducing Dirichlet priors in user profile assignments [23, 29, 30]. However, this setting is limited in it's ability to model skew, and could merge infrequent behavior profiles with common ones. Furthermore, behaviors learned could be contaminated by the presence of several inactive users.

In contrast to these approaches, we propose to simultaneously partition and profile users in a unified latent framework to adapt to varying degrees of skew and data sparsity in our network. Our user-partioning scheme builds upon preferential attachment models [1, 27] to explicitly discount common activity profiles and favor exploration of diverse user partitions, learning fine-grained representations of behavioral signals. Mutual-enhancement of behavior profiling and user-partitioning can also bridge temporal inconsistencies or sparsity in participant activity. Users exhibiting similar engagement patterns are jointly grouped and profiled within partitions. Furthermore, our latent behavior profiles can be flexibly defined to integrate several facets of user behavior and evolution, hence generalizing to diverse social platforms and applications.

The main contributions of this paper are:

- **Partitioning and Profiling :** We simultaneously generate flexible user partitions and profile user behavior within partitions in a unified mutually-enhancing latent framework. Our partitioning scheme can adapt to varying levels of behavior skew, effectively uncover fine-grained or infrequent engagement patterns, and address user-level sparsity.
- **Generalizability :** Our model is generalizable to diverse platforms and applications. User profiles can be flexibly defined to integrate several facets of user behavior, social activity and temporal evolution of interests, providing comprehensive user representations.
- **User Evolution :** We formalize our evolutionary profiles to integrate the time-evolving content-action associations observed in user activity and social dynamics.
- **Efficiency :** We provide several optimizations for efficient model inference (see Section 5) and scale linearly in the size of our datasets compared to quadratic-time scaling in tensor factorization approaches.

Extensive experiments over large Coursera[3] datasets as well as Stack-Exchange websites indicate that our approach strongly outperforms state-of-the-art baselines. We perform three prediction tasks: certificate completion prediction (MOOCs), reputation prediction (Stack Exchange), and behavior distribution prediction. For certificate prediction, we outperform baselines on the AUC measure by 6.26%-15.97%; reputation prediction by 6.65%-21.43%; behavior prediction MOOCs (12%-25%) and Stack-Exchanges (9.5%-22%).

[3]https://coursera.org

On experiments related to activity sparsity, we see magnified gains on participants who offer limited data (10.2%-27.1%). We also examine the effects of reducing behavior skew: our approach still outperforms baselines on data with reduced skew. Our scalability analysis shows that the model scales well, and is amenable to a parallel implementation. Finally, we study the effects of model parameters and obtain stable performance over a broad range of parameter values, indicating that in practice our model requires minimal tuning.

We organize the rest of the paper as follows. In the next section, we formalize our data abstraction and user representation. In Section 3, we describe the datasets used and in Section 4, we present our approach. Section 5 describes a collapsed Gibbs sampler to infer model parameters, and experimental results are presented in Section 6. We discuss related work in Section 7 and conclude the paper in Section 8.

2 PROBLEM DEFINITION

We study networks where participants seek to primarily gain and exchange knowledge (e.g. MOOCs, Stack-Exchanges). In these networks, participants act (e.g. "post", "play video", "answer") on content and communicate with other participants. Content may either be participant-generated (e.g. in a forum), or external (e.g. MOOC lecture). Interactions with content encode latent knowledge and intent of participants - for instance, answering or editing published content is indicative of subject expertise. Furthermore, social exchanges between participants play an important role in profiling their activity.

Let $\mathcal{U}$ denote the set of all participants on the network. These participants employ a finite set of distinct actions $\mathcal{A}$ to interact with content generated from vocabulary $\mathcal{V}$. Atomic participant activity is refered to as an interaction. We define each interaction d as a tuple $d = (a, W, t)$, where the participant performs action $a \in \mathcal{A}$ on content $W = \{w \mid w \in \mathcal{V}\}$ at a normalized time-stamp $t \in [0, 1]$. We denote the set of all interactions of participant u as $\mathcal{D}_u$. Thus the entire collection of interactions in the network is given by, $\mathcal{D} = \cup_{u \in \mathcal{U}} \mathcal{D}_u$.

Inter-participant links are represented by a directed multigraph $G = (\mathcal{U}, E)$. A directed labeled edge $(u, v, l) \in E$ is added for each interaction of user u, $d_u \in \mathcal{D}_u$(e.g. "answer") that is in response to an interaction of user v, $d_v \in \mathcal{D}_v$ (e.g. "ask question") with edge label $l \in \mathcal{L}$ indicating the specific nature of the social exchange (e.g. "answer" $\longrightarrow$"question").

Our model infers a set of latent activity profiles R, where each profile $r \in R$ encodes a specific evolutionary pattern of user behavior and social engagement. Observable facets of user behavior, namely $(D_u, L_u), u \in \mathcal{U}$ are drawn from profile $r \in R$ with likelihood $p(D_u, L_u \mid r)$ which we abbreviate as $p(u \mid r)$. Each user is then represented by the vector of likelihoods over latent profiles, i.e. $\mathcal{P}_u = [p(u \mid r) \; \forall \; r \in R]$.

3 DATASET DESCRIPTION

We study datasets from two diverse real-world learning platforms. They encompass rich temporal behavior data in conjunction with textual content, and a community element whereby participants form social ties with each other.

Platform	Action	Description
MOOC	Play	First lecture segment view
	Rewatch	Repeat lecture segment view
	Clear Concept	Back and forth movement, pauses
	Skip	Unwatched lecture segment
	Create Thread	Create a forum thread with a question
	Post	Reply to existing threads
	Comment	Comment on existing posts
Stack Ex.	Question	Posting a question
	Answer	Authoring answer to a question
	Comment	Comment on a question/answer
	Edit	Modify posted content
	Favorite	Liking posted content

Table 1: User Action Description (Coursera/Stack-Exchange)

Stack-Exchange Q&A Networks: Stack-Exchanges are Q&A websites covering broad domains of public interest. Users interact by asking/answering questions, and editing, liking and commenting on published content (Table 1). Furthermore, users communicate by reacting to other users' activity, specifically liking and editing content, favorite, and answering questions hence setting up Editing, Liking and Answering links between the pair of users, indicative of their shared interests and knowledge. We apply our model on several Stack-Exchange websites from varied domains (Table 2).

Coursera MOOC Platform: Coursera MOOCs feature a structured learning environment driven by both, lecture content and communication between students and instructors via multiple course forums. Patterns of lecture viewing obtained from video clickstreams provide valuable cues on student learning behavior [28, 38], in addition to forum activity [24]. We combine these two sources to define the action set of students (Table 1). Lecture segment content is extracted from subtitle files. Students engage in social exchanges by commenting on or upvoting content from their peers. We study several MOOC datasets described in Table 2.

Platform	Dataset	#Users	#Interactions	η_t	S_N
Coursera	Math	10,796	162,810	-2.90	0.69
	Nature	6,940	197,367	-2.43	0.70
	Comp Sci-1	26,542	834,439	-2.51	0.67
	Comp Sci-2	10,796	165,830	-2.14	0.73
Stack-Ex	Ask-Ubuntu	220,365	2,075,611	-2.81	0.65
	Android	28,749	182,284	-2.32	0.56
	Travel	20,961	277,823	-2.01	0.66
	Movies	14,965	150,195	-2.17	0.67
	Chemistry	13,052	175,519	-2.05	0.63
	Biology	10,031	138,850	-2.03	0.71
	Workplace	19,820	275,162	-2.05	0.59
	Christianity	6,417	130,822	-1.71	0.64
	Comp. Sci.	16,954	183,260	-2.26	0.62
	Money	16,688	179,581	-1.72	0.63

Table 2: Preliminary Analysis of Behavior Skew and Temporal Inconsistency of participant activity in our datasets

To quantify data sparsity in our datasets, we compute the power-law index (η_t) that best describes the fraction of users against number of weeks with activity. A more negative index indicates that fewer users are consistently active over time. Behavioral skew can be quantified by grouping participants by their dominant action type (e.g. users who mostly comment), and computing normalized entropy (S_N) of the resulting distribution of users across actions. In large Stack-Exchanges such as Ask-Ubuntu, 'Answer' is the dominant action for less than 5% users while 'Comment' accounts for over 60% (Figure 1). In MOOCs, 'Play' is the most common action and forum interactions are rare (~10-15% participation), resulting in fewer social links. It is interesting to observe that large Stack-Exchanges have more inactive users in comparison to niche domains of discussion (Ask-Ubuntu vs Money, Table 2).

4 OUR APPROACH

In this section, we motivate our behavior profiling framework (Section 4.1) proceeding in two simultaneous mutually-enchancing steps. Discovering diverse homogenous partitions of users in the network (Section 4.2), and learning latent evolutionary profiles characterizing facets of their content interactions and social exchanges (Section 4.3).

4.1 Motivation

We develop our profiling framework with two key objectives. First, to account for behavior skew (participants are unevenly distributed across varying behavior patterns) as well as temporal inconsistency and sparsity in user-level data. Second, to learn informative evolutionary profiles simultaneously characterizing engagement with content and social exchanges between users.

Modeling inherent behavior skew necessitates the development of profiling approaches that adapt to the observed data and learn informative representations of user activity. Generative behavior models are traditionally controlled by introducing Dirichlet priors in the topic assignment process [23, 29, 30]. However, this setting is limited in it's ability to model inherent topical skew, where some outcomes significantly outnumber others. In the context of user behavior, it is necessary to explicitly account for the presence of skew in the proportions of behavior patterns and effectively separate users to learn discriminative evolutionary profiles of activity. There are two key advantages to mutually enhancing user paritioning and profile learning over conventional profile assignments:

- **Tackling Behavior Skew** : Conventional topic assignments tend to merge infrequent behavior patterns with common ones, resulting in uninformative profiles. Our approach explicitly discounts common profiles and favors diverse user partitions. Profile variables assigned to these partitions learn informative representations of infrequent behaviors.
- **Temporal Inconsistency and Sparsity** : Our profile assignment process enforces common profiles across inconsistent and active users within homogenous partitions. As our inference process converges, users with limited data are probabilistically grouped with the most similar users based on available interaction data.

We now proceed to formalize our user partitioning scheme.

Symbol	Description
$\mathcal{D}, \mathcal{U}, \mathcal{A}, \mathcal{V}$	Set of all content interactions, users, actions and content vocabulary
$\mathcal{D}_u, \mathcal{L}_u$	Content interactions and social links observed for user $u \in \mathcal{U}$
$d = (a, W, t)$	Interaction involving action a on content W at time t
$(s, u, l), (u, y, l') \in \mathcal{L}_u; l, l' \in \mathcal{L}$	l-label inward link from source s, l'-label outward link to target y; $\mathcal{L}$ - predefined link label set
R, K	The set of evolution profiles, and the set of behavior topics
$\phi_k^{\mathcal{V}}, \phi_k^{\mathcal{A}}$	Multinomial word and action distributions in behavior $k \in K$
ϕ_r^K	Multinomial distribution over behaviors for profile $r \in R$
α_{rk}, β_{rk}	Parameters of the temporal beta distribution for behavior k in profile r
$\phi_{r,r'}^{\mathcal{L}}$	Multinomial distribution over link labels for links directed from profile r to r'
γ, δ, G_0	Scale parameter, discount parameter and base distribution of Pitman-Yor process
$a, n_a, r_a; N, A_r$	Table ID, # seated users, profile served on it; # total seated users and # tables serving profile $r \in R$
$\alpha_{\mathcal{V}}, \alpha_{\mathcal{A}}, \alpha_K, \alpha_{\mathcal{L}}$	Dirichlet-Multinomial priors for $\phi_k^{\mathcal{V}}$, $\phi_k^{\mathcal{A}}$, ϕ_r^K and $\phi_{r,r'}^{\mathcal{L}}$ $\forall\, k \in K$ & $r, r' \in R$

Table 3: Notations used in this paper

4.2 Skew Aware User Partitioning

We first introduce a basic preferential attachment model based on the Pitman-Yor process [27] to generate skewed partitions of integeres. We proceed to develop our profile-driven user partitioning scheme, building upon the Chinese Restaurant perspective [1] of the Pitman-Yor process to group similar users within partitions. Our approach explicitly discounts common behavior profiles to generate diverse user partitions and learn subtle variations and infrequent behavior patterns in the network. Additionally, it jointly profiles sparse and temporally inconsistent users with best-fit partitions.

4.2.1 Basic Preferential Attachment. The Pitman-Yor process [27] (generalization of the Dirichlet process [41]) induces a distribution over integer partitions, characterized by a concentration parameter γ, discount parameter δ, and a base distribution G_0. An interpretable perspective is provided by the Chinese Restaurant seating process [1] (CRP), where users entering a restaurant are partitoned across tables. Each user is either seated on one of several existing tables $[1, \ldots, A]$, or assigned a new table $A + 1$ as follows,

$$p(a|u) \propto \begin{cases} \frac{n_a - \delta}{N+\gamma}, & a \in [1, A], \text{ existing table} \\ \frac{\gamma + A\delta}{N+\gamma}, & a = A + 1, \text{ new table} \end{cases} \tag{1}$$

where n_a is the number of users seated on existing tables $a \in [1, A]$, $A + 1$ denotes a new table, and $N = \sum_{a \in [1,A]} n_a$ is the total number of participants seated across all tables. Equation (1) thus induces a preferential seating arrangement proportional to the current size of each partition. The concentration (γ) and the discount (δ) parameters govern the formation of new tables.

This simplistic assignment thus captures the "rich get richer" power-law characteristic of online networks [3]. However a significant drawback is it's inability to account for similarities of users seated together. Our approach enforces a profile-aware seating arrangment to generate homogenous user partitions.

4.2.2 Profile-Driven Preferential Attachment. Let us now assume the presence of a set of evolutionary profiles, R describing temporal patterns of user engagement and their social ties. Each user u $\in \mathcal{U}$ is associated with a set of time-stamped interactions D_u, and social links L_u. The likelihood of generating these observed facets via profile $r \in R$ is given by p$(D_u, L_u \mid r)$, which we abbreviate to p$(u \mid r)$.

Thus, to continue the restaurant analogy above, we "serve" a table-specific profile $r_a \in R$ to participants seated on each table $a \in [1, A]$. When we seat participant u on a new table $A + 1$, a profile variable $r_{A+1} \in R$ is drawn on the new table to describe u,

$$r_{A+1} \sim p(u \mid r)p(r)$$

where $p(r)$ is parameterized by the base distribution G_0 of the Pitman-Yor process, acting as a prior over the set of profiles. We set $p(r)$ to a uniform distribution to avoid bias. A user u in our model is thus seated on table a as follows,

$$p(a|u) \propto \begin{cases} \frac{n_a - \delta}{N+\gamma} \times p(u \mid r_a), & a \in [1, A], \\ \frac{\gamma + A\delta}{N+\gamma} \times \frac{1}{|R|} \sum_{r \in R} p(u \mid r), & a = A + 1. \end{cases} \tag{2}$$

Thus, the likelihood of assigning a specific profile r to participant u, $p(r|u)$ is obtained by summing up over the likelihoods of being seated on existing tables serving r, and the likelihood of being seated on a new table $A + 1$ with the profile draw $r_{A+1} = r$,

$$p(r \mid u) = \Big(\sum_{a \in [1,A], r_a = r} \frac{n_a - \delta}{N + \gamma} p(u \mid r) \Big) + \frac{1}{|R|} \cdot \frac{\gamma + A\delta}{N + \gamma} p(u \mid r) \tag{3}$$

$$= \left(\frac{N_r - A_r \delta}{N + \gamma} + \frac{\gamma + A\delta}{|R|(N + \gamma)} \right) p(u \mid r) \tag{4}$$

where A_r is the number of existing tables serving profile r and N_r is the total number of participants seated on tables serving r.

Discount Factor : The extent of skew is jointly controlled by both, the number of users exhibiting similar activity patterns, encoded by $p(u \mid r)$ as well as the setting of the discount parameter δ. Common profiles are likely to be drawn on several tables, thus their probability masses are discounted by the product $A_r \delta$ in Equation (3). A higher setting of δ favors exploration by seating users on new tables and generating diverse partitions, learning subtle variations in profiles rather than merging them.

Temporal Inconsistency : Users offering limited evidence are likely to be assigned to popular profiles that well explain their limited interaction data. Our user partitioning scheme enforces a common profile assignment across users sharing a partition, thus ensuring proximity of their likelihood distributions over the latent space of the inferred evolutionary profiles.

Algorithm 1 Profile-Driven Preferential Attachment

for each user $u \in \mathcal{U}$ **do**
  Sit at existing table $a \in [1, A] \propto \frac{n_a - \delta}{N+\gamma} \times p(u \mid r_a)$
  Sit at new table $A+1 \propto \frac{\gamma + A\delta}{N+\gamma} \times (\sum_{r \in R} p(u \mid r) \times \frac{1}{|R|})$
  if $A+1$ is chosen **then**
    Draw $r_{A+1} \propto p(u \mid r) \times \frac{1}{|R|}, r_{A+1} \in R$

Simplistic prefential assignment (Equation (1)) can be interpreted as a specialization of our model where all evolutionary profiles $r \in R$ are equally likely for every user. Our model effectively generalizes Equation (1) to generate partitions of typed entities rather than integers. The resulting seating arrangement in our model can be shown to be exchangeable similar to that in [1] and hence amenable to efficient inference. Our partitioning approach can be extended to several diverse applications, depending on the precise formulation of p($u \mid r$). In the next subsection, we formalize our definition of evolutionary profiles $r \in R$.

4.3 Evolutionary Activity Profiling

We now formalize the notion of latent behaviors and evolutionary activity profiles. A behavior (or a behavioral topic) $k \in K$ is jointly described by $\phi_k^{\mathcal{V}}$, a $|\mathcal{V}|$ dimensional multinomial distribution over words, and $\phi_k^{\mathcal{A}}$, a $|\mathcal{A}|$ dimensional multinomial distribution over the set of actions. The combined probability of observing an action a (e.g. "play", "post") on content $W = \{w \mid w \in \mathcal{V}\}$ (e.g. a sentence on "Probability") conditioned on behavior k can be given by,

$$p(a, W \mid k) \propto \phi_k^{\mathcal{A}}(a) \prod_{w \in W} \phi_k^{\mathcal{V}}(w). \tag{5}$$

Each observed interaction is assumed to be drawn from a single behavior $k \in K$, thus learning consistent action-content associations.

Evolutionary profiles are temporal mixtures over behaviors. We describe each activity profile $r \in R$ jointly by a K dimensional multinomial-distribution parameterized by ϕ_r^K over the K latent behaviors, and a *Beta* distribution specific to each behavior $k \in K$, over normalized time $t \in [0, 1]$, parameterized by $\{\alpha_{rk}, \beta_{rk}\}$. Each component of the multinomial distribution $\phi_r^K(k)$ is the likelihood of observing behavior k in profile r. The α_{rk}, β_{rk} parameters of the *Beta* distributions capture the temporal trend of behavior k within profile r.

We draw an interaction $d = (a, W, t)$ within profile r by first drawing a behavior k in proportion to $\phi_r^K(k)$. We then draw action a and content W conditioned on behavior k, and time t conditioned on both r and k. Thus, the likelihood of observing interaction d in profile r, $p(d \mid r)$ is obtained by marginalizing over behaviors,

$$p(d \mid r) \propto \sum_k \phi_r^K(k) \times p(a, W \mid k) \times p(t \mid r, k). \tag{6}$$

where $p(a, W \mid k)$ is computed as in Equation (5) and $p(t \mid r, k)$ through the corresponding *Beta* distribution $Beta(t; \alpha_{rk}, \beta_{rk})$,

$$p(t \mid r, k) = \frac{t^{\alpha_{rk}-1}(1-t)^{\beta_{rk}-1}}{B(\alpha_{rk}, \beta_{rk})}. \tag{7}$$

The *Beta* distribution offers us flexibility in modeling temporal association. Prior behavior models [13, 28] used static time slicing to describe user evolution. Choosing an appropriate temporal granularity is challenging. A single granularity may be inadequate to model heterogeneous user activity. Since we analyze behavior data recorded over finite intervals, the parameterized Beta distribution is capable of learning flexible continuous variations over normalized time-stamps via parameter estimation [43].

We can now compute the likelihood of observing the entire set of interactions $\mathcal{D}_u$ of a user $u \in \mathcal{U}$ conditioned on profile r as,

$$p(D_u \mid r) \propto \prod_{d \in \mathcal{D}_u} p(d \mid r) \tag{8}$$

The above process is summarized in Algorithm 2. In addition to interaction set $\mathcal{D}_u$, we now proceed to exploit inter-participant link structure in the construction of activity profiles.

4.3.1 Modeling Participant Links. We create a directed multigraph between pairs of profiles to incorporate the nature of participant ties. Edge labels $l \in \mathcal{L}$ describe the nature of exchanges (e.g. "question", "answer") connecting users who perform them, and the directionality of the exchange is indicative of the implicit social relationship between their profiles. We thus associate a multinomial distribution parameterized by $\phi_{r,r'}^{\mathcal{L}}$ between each ordered pair of profiles r, r', setting up $|R|^2$ distributions in all.

The network structure also enhances modeling for participants with inconsistent interaction data to take advantage of more extensive interaction data of their neighbors. The probability $p(L_u \mid r_u)$ of the set of links L_u associated with user u is proportional to the likelihoods of independently observing each link, based on the current profile assignment $r_u \in R$, and the participant profiles of users linked to u,

$$p(L_u \mid r_u) \propto \prod_{(s,u,l) \in L_u} \phi_{r_s, r_u}^{\mathcal{L}}(l) \times \prod_{(u,y,l) \in L_u} \phi_{r_u, r_y}^{\mathcal{L}}(l), \tag{9}$$

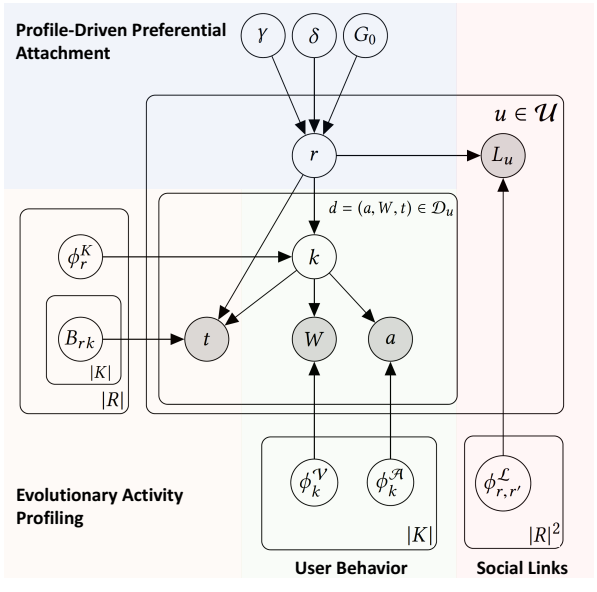


Figure 3: Graphical model illustrating our simultaneous partitioning and profiling framework

Algorithm 2 Generative process for drawing facets $\mathcal{D}_u, L_u$ from profile $r \in R$ assigned to the partition containing user $u \in \mathcal{U}$

for each behavior $k \in K$ **do**
  Draw word distribution $\phi_k^{\mathcal{V}} \sim Dir(\alpha_{\mathcal{V}})$
  Draw action distribution $\phi_k^{\mathcal{A}} \sim Dir(\alpha_{\mathcal{A}})$
for each profile $r \in R$ **do**
  Draw distribution over behaviors, $\phi_r^K \sim Dir(\alpha_K)$
  for each profile $r' \in R$ **do**
    Choose link distribution $\phi_{r,r'}^{\mathcal{L}} \sim Dir(\alpha_{\mathcal{L}})$
for each behavior interaction $d = (a, W, t) \in \mathcal{D}_u$ **do**
  Choose behavior $k \sim Multi(\phi_r^K)$
  for word $w \in W_d$ **do**
    Draw $w \sim Multi(\phi_k^{\mathcal{V}})$
  Draw action $a \sim Multi(\phi_k^{\mathcal{A}})$
  Draw normalized time $t \sim Beta(\alpha_{rk}, \beta_{rk})$
for each inward link (s,u,l) $\in L_u$ **do**
  Let r_s denote source user profile
  Draw $(s, u, l) \sim Multi(\phi_{r_s,r}^{\mathcal{L}})$
for each outward link (u,y,l) $\in L_u$ **do**
  Let r_y denote target user profile
  Draw $(u, y, l) \sim Multi(\phi_{r,r_y}^{\mathcal{L}})$

where $\phi_{r_s,r_u}^{\mathcal{L}}(l)$ is the likelihood of an l-labeled inward link to user u emerging from a source user s with profile r_s, and $\phi_{r_u,r_y}^{\mathcal{L}}(l)$, is that of the analogous outward link to user y. We thus encode observed social links as implicit relationships of the respective evolutionary profiles.

We combine social ties L_u and content interactions D_u (eqs. (8) and (9)), to compute the joint conditional probability $p(u \mid r)$,

$$P(u \mid r) \propto p(D_u \mid r) \times p(L_u \mid r). \tag{10}$$

The above equation provides the likelihood of describing user $u \in \mathcal{U}$ with a chosen profile $r \in R$. The generative process corresponding to eq. (10) is summarized by Algorithm 2.

In this section, we motivated our partitioning and profiling framework to tackle the issues of skew and sparsity in social learning environments. Next, we describe a collapsed Gibbs-sampling approach [19] for model inference, where we iteratively sample user seating arrangements and update profile parameters to reflect the resulting set of user partitions, until mutual convergence.

5 MODEL INFERENCE

In this section we describe a collapsed Gibbs-sampling [19] approach for model inference, an analysis of it's computational complexity, and propose an efficient parallel batch-sampler to scale to large datasets.

5.1 Inference via Gibbs Sampling

We exploit the widely used Markov-Chain Monte-Carlo(MCMC) algorithm, collapsed Gibbs-sampling, to sample user seating and learn profiles by iteratively sampling the latent profile variable r_u for each user $u \in \mathcal{U}$, latent behavior-topic assignments k_d for interactions $d \in \mathcal{D}_u$, and table a_u serving sampled profile r_u, on which user u is seated.

Symbol	Description
$n_k^{(w)}, n_k^{(a)}, n_k^{(.)}$	Number of times word w, action a were assigned to topic k, and respective marginals
$n_r^{(k)}, n_r^{(.)}$	Number of interactions of users in r assigned topic k, total interactions of all users in r
$n_{r,r'}^{(l)}, n_{r,r'}^{(.)}$	Number of l-label links across users in profile r with r', and total links between r and r'

Table 4: Gibbs-sampler count variables

5.1.1 Initialization: Randomized initializtion of user partitions and corresponding profile assignments could result in longer convergence times. We speed-up convergence by exploiting content tags and action distributions of users to generate a coherent initial seating arrangement. Users with similar action and content tag distributions are seated together to form homogenous partitions.

5.1.2 Sampling User Partitions: The likelihood of generating interaction $d = (a, W, t) \in \mathcal{D}_u$ from behavior $k \in K$ (Equation (5)) can be given by,

$$p(a, W \mid k) \propto \frac{n_k^{(a)} + \alpha_{\mathcal{A}}}{n_k^{(.)} + |\mathcal{A}|\alpha_{\mathcal{A}}} \times \prod_{w \in W} \frac{n_k^{(w)} + \alpha_{\mathcal{V}}}{n_k^{(.)} + |\mathcal{V}|\alpha_{\mathcal{V}}} \tag{11}$$

Thus the likelihood of observing interaction $d = (a, W, t)$ in profile $r \in R$ (Equation (6)) is,

$$p(d \mid r) \propto \sum_{k \in K} \frac{n_r^k + \alpha_K}{n_r^{(.)} + |K|\alpha_K} \times p(a, W \mid k) \times p(t \mid r, k) \tag{12}$$

where $p(t \mid r, k)$ is computed as in eq. (7). Link likelihood for source profile p, target p' and label l is computed as,

$$\phi_{p,p'}^{\mathcal{L}}(l) = \frac{n_{p,p'}^l + \alpha_{\mathcal{L}}}{n_{p,p'}^{(.)} + |\mathcal{L}|\alpha_{\mathcal{L}}} \tag{13}$$

Thus $p(u \mid r), u \in \mathcal{U}$ (eq. (8)) can be obtained as the product of eq. (12) over $d \in \mathcal{D}_u$ and eq. (13) over L_u respectively. Given $p(u \mid r)$ we can sample profile r_u for user u as in eq. (3),

$$P(r_u = r \mid u, a_{-u}, r_{-u}, k_{-u}) \sim \left(\frac{N_r - A_r\delta}{N + \gamma} + \frac{\gamma + A\delta}{|R|(N + \gamma)} \right) p(u \mid r) \tag{14}$$

where a_{-u}, r_{-u}, k_{-u} indicate the seating and profile assignments of all other users, and the behavior assignments for their interactions. Behavior assignments for each interaction $d \in \mathcal{D}_u$ are sampled in proportion to eq. (12) with $r = r_u$, the chosen profile for u, and the user is seated either on an existing table $a \in [1, A]$ serving r_u, or new table $A + 1$ with $r_{A+1} = r_u$,

$$\begin{cases} a \in [1, A] \propto \frac{n_a - \delta}{N + \gamma} \text{ if } r_a = r_u, \text{else } 0 \\ \text{New table } A + 1 \propto \frac{\gamma + (\delta \times A)}{N + \gamma} \times \frac{1}{|R|}, r_{A+1} = r_u \end{cases}$$

Note that $N = |\mathcal{U}| - 1$, i.e. all users except u.

5.1.3 Parameter Estimation: All counts (Table 4) corresponding to previous behavior and profile assignments of u are decremented and updated based on the new assignments drawn. At the end of each sampling iteration, Multinomial-Dirichlet priors $\alpha_{\mathcal{V}}$, $\alpha_{\mathcal{A}}$, α_K and $\alpha_{\mathcal{L}}$ are updated by Fixed point iteration [26] and profile parameters $(\alpha_{rk}, \beta_{rk})$ are updated by the method of moments [43].

All time-stamps are rounded to double-digit precision and values of $p(t \mid r, k)\ \forall\ t \in [0, 1], r \in R, k \in K$ are cached at the end of each sampling iteration. This prevents $R \times K$ scaling for $p(u \mid r)$ in Equation (14) by replacing computations with fetch operations. Pitman-Yor parameters can be estimated via auxiliary variable sampling with hyperparameters set to recommended values in [37, 40].

5.2 Computational Complexity

In each Gibbs iteration, Equations (11) and (12) involve $|\mathcal{D}| \times (K + R)$ computations. Equation (14) requires an additional $|\mathcal{U}| \times R$ computations. $R \times K$ scaling for $p(u \mid r)$ in Equation (14) is prevented by restricting temporal precision as described in section 5.1. The first product term of Equation (14) is cached for each $r \in R$, and updated only when tables of profile r are altered.
On the whole, our algorithm is linear in $|\mathcal{D}| + |\mathcal{U}|$, scaled by $R + K$ in both time and space complexity (results in Figure 6). We efficiently scale to massive datasets by parallelizing our algorithm across users via batch-sampling, described in the next subsection.

5.3 Parallelizing Model Inference

The Gibbs sampler described above samples each user's seating $P(r_u = r \mid u, a_{-u}, r_{-u}, k_{-u})$ conditioned on all other users, which necessitates iteration over $\mathcal{U}$. Instead, seating arrangements could be simultaneously sampled in batches $U \subset \mathcal{U}$ conditioned on all users outside the batch, i.e. $P(R_U = R \mid U, a_{\mathcal{U}-U}, r_{\mathcal{U}-U}, k_{\mathcal{U}-U})$.

Batch sampling is most efficient when each batch $U \subset \mathcal{U}$ is chosen such that users $u \in U$ entail comparable computational loads. We approximate computation load for $u \in \mathcal{U} \propto |\mathcal{D}_u| + |\mathcal{L}_u|$ to decide apriori batch splits for sampling iterations.

All assignment counts can be updated at the end of the sampling process for one batch. Note that social links between users in a given batch U are ignored since their profiles are drawn simultaneously. However, in practice the batch-size is a small value in comparison to $|\mathcal{U}|$, thus rendering this loss to be negligible.

6 EXPERIMENTAL RESULTS

In this section, we present our experimental results on datasets from MOOCs as well as Stack-Exchange. We first introduce the set of competing baselines. Then in Section 6.2, we discuss prediction tasks used to evaluate the different behavioral representation models. In Section 6.3, we demonstrate the impact of data sparsity on prediction tasks, and in Section 6.4, we discuss the effects of behavior skew. Next, we present results on the parameter sensitivity and scalability of our model in Sections 6.5 and 6.6 and conclude with a discussion on the limitations of our approach.

6.1 Baseline Methods

We compare our model (CMAP) with three state-of-the-art behavior models and two standard baselines.

LadFG [28]: LadFG is a dynamic latent factor model which uses temporal interaction sequences and demographic information of participants to build latent representations. We provide LadFG action-content data from interactions and all available user demographic information.

BLDA [29]: BLDA is an LDA based extension to capture actions in conjunction with text. It represents users as a mixture over latent content-action topics.

FEMA [13]: FEMA is a multifaceted sparsity-aware matrix factorization approach which employs regularizer matrices to tackle sparsity. Facets in our datasets are users, words and actions. We set User-User and Word-Word regularizers to link and co-occurrence counts respectively. We could not run FEMA on Ask-Ubuntu and Comp Sci-1 datasets owing to very high memory and compute requirements. Regularizer matrices in FEMA scale as $|\mathcal{U}|^2$, whereas our model scales as $|\mathcal{U}|$.

DMM (Only text) [47]: We apply DMM to the textual content of all interactions to learn topics. Users are represented by the probabilistic proportions of learned topics in their interaction content.

Logistic Regression Classifier (LRC) [17]: It uses logistic regression to train a classifier model for prediction. Input features include topics(obtained from DMM) that the user interacts with and respective action proportions for each topic.

We initialize Dirichlet priors for $\phi_k^{\mathcal{V}}, \phi_k^{\mathcal{A}}, \phi_r^K$ and $\phi_{r,r'}^{\mathcal{L}}$ following the common strategy [9, 46] ($\alpha_X = 50/|X|, X = \{\mathcal{A}, \mathcal{L}, K\}$, and $\alpha_{\mathcal{V}} = 0.01$) and all *Beta* parameters α_{rk}, β_{rk} are initialized to 1. We found CRP parameter initialization at $\delta = 0.5, \gamma = 1$ to perform consistently well across datasets. All models were implemented in Python, and experiments were performed on an x64 machine with 2.4GHz Intel Xeon cores and 16 GB of memory.

6.2 Prediction Tasks

In this section, we identify three prediction tasks, discuss evaluation metric and compare results with competing baseline methods. We focus on two platform specific user characterization tasks, and a common content-action prediction task.

Certificate Prediction: Students maintaining a minimum cumulative grade over course assignments are awarded certifications by Coursera. Connecting behavior to performance may help better online educational experiences. We attempt to predict students recieving certificates based on their behavioral data in each MOOC.

User Reputation Prediction: For Stack-Exchanges, we predict if participants have a high reputation. Participants receive a reputation score based on the quality of their participation. We define high reputation as the top quartile of all reputation scores on that stack-exchange.

Behavior Prediction: We predict the distribution of participant behavior across content and actions, in all Stack-Exchanges and MOOC datasets. Specifically, for each dataset, we extract $T = 20$ topics from the text of all interactions using DMM [47], and assign each participant interaction to the most likely topic learned by DMM.

We use standard classifiers and evaluation metrics to evaluate all models. Prediction tasks use linear kernel Support Vector Machine (SVM) classifier (with default parameters) in sklearn[4] and we compute results for each dataset through 10-fold cross validation.

[4] http://scikit-learn.org/

Method	Precision	Recall	F1-score	AUC
LRC	0.76 ± 0.04	0.71 ± 0.05	0.74 ± 0.04	0.72 ± 0.03
DMM	0.77 ± 0.03	0.74 ± 0.04	0.75 ± 0.03	0.74 ± 0.03
LadFG	0.81 ± 0.02	0.78 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
FEMA	0.78 ± 0.03	0.75 ± 0.04	0.76 ± 0.03	0.78 ± 0.03
CMAP	**0.86 ± 0.02**	**0.81 ± 0.03**	**0.83 ± 0.02**	**0.84 ± 0.02**
BLDA	0.80 ± 0.04	0.75 ± 0.03	0.77 ± 0.03	0.77 ± 0.04

Table 5: Certification Prediction ($\mu \pm \sigma$ across MOOCs). CMAP outperforms baselines by 6.26-15.97% AUC.

Method	Precision	Recall	F1-score	AUC
LRC	0.73 ± 0.04	0.69 ± 0.04	0.72 ± 0.03	0.73 ± 0.03
DMM	0.69 ± 0.05	0.65 ± 0.04	0.66 ± 0.04	0.70 ± 0.04
LadFG	**0.86 ± 0.03**	0.75 ± 0.03	0.79 ± 0.02	0.80 ± 0.03
FEMA	0.79 ± 0.04	0.73 ± 0.03	0.77± 0.03	0.79 ± 0.04
CMAP	0.85 ± 0.02	**0.83 ± 0.03**	**0.84 ± 0.02**	**0.86 ± 0.02**
BLDA	0.75 ± 0.04	0.71 ± 0.04	0.74 ± 0.03	0.74 ± 0.04

Table 6: Reputation Pred. ($\mu \pm \sigma$ across Stack-Exchanges). CMAP outperforms baselines by 6.65-21.43% AUC.

LRC is not used in behavior prediction since it does not build a user representation. We evaluated performance of all methods in the certificate and reputation prediction tasks via Precision, Recall, F1-Score and Area-Under-Curve (AUC). For the behavior prediction task, we measure the Root Mean Squared Error (RMSE) in the predicted user interaction proportions for *(topic, action)* pairs against the actual interaction proportions of users.

We show strong results across prediction tasks. In the certificate prediction task, our method improves on the baselines using the AUC measure by 6.26-15.97%, averaged across MOOCs (c.f. Table 5). In the reputation prediction task over Stack-Exchanges, we improve on all baselines on the AUC metric in the range 6.65-21.43%. In the behavior prediction task, our method improves on competing baselines using RMSE by 12%-25% on MOOCs and between 9.5%-22% on Stack-Exchange datasets (c.f. Table 9).

6.3 Effects of Data Sparsity

In order to study the gains of our algorithm on characterizing users with different levels of activity, we split participants in each dataset into four equal partitions based on their number of interactions (Quartiles 1-4, 1 being least active). We then sample participants from each quartile and evaluate all methods on prediction performance (AUC).

Our model shows magnified gains (Figure 4) in prediction performance over baseline models in the first and second quartiles

Method	DMM	LadFG	FEMA	CMAP	BLDA
MOOC	4.9 ± 0.4	4.2 ±0.3	4.1 ± 0.2	**3.6 ± 0.2**	4.4 ± 0.4
Stack-Ex	8.6 ± 0.6	7.9 ± 0.4	7.5 ± 0.3	**6.7 ± 0.4**	7.4 ± 0.5

Table 7: Behavior Prediction (RMSE ($\times 10^{-2}$) $\mu \pm \sigma$). CMAP ouperforms baselines in MOOCs (12%-25%) and Stack-Exchanges (9.5%-22%)

which correspond to sparse or inactive users. BLDA performs the weakest in Quartile-1 since it relies on interaction activity to build user representations. Our model effectively bridges gaps in sparse or inconsistent participant data by exploiting similar active users within user partitions.

6.4 Effects of Behavior Skew

We study the effect of behavioral skew on the prediction results, by subsampling users who predominantly perform the two most common activities in our two largest datasets, Ask-Ubuntu (Comments and Questions) and Comp Sci-1 (Play and Skip) in half, and retaining all other users. This reduces overall skew in the data. Baseline models are expected to perform better with de-skew. All models degrade in Ask-Ubuntu owing to significant content loss in the de-skew process.

Method	Ask-Ubuntu		CompSci1 MOOC	
	Original	Deskewed	Original	Deskewed
LRC	0.671	0.656	0.713	0.734
DMM	0.647	0.611	0.684	0.672
LadFG	0.734	0.718	0.806	0.830
BLDA	0.706	0.683	0.739	0.788
CMAP	**0.823**	**0.746**	**0.851**	**0.839**

Table 8: CMAP outperforms baselines (AUC) in original and de-skewed datasets. Performance gap reduces with de-skew.

We also investigate performance gains achieved by our approach in our most skewed and sparse Stack-Exchange (Ask-Ubuntu) vs least skewed (Christianity, Table 2). On average, our model outperforms baselines by 13.3% AUC for Ask-Ubuntu vs 10.1% for Christianity Stack-Exchange in the Reputation Prediction task.

Method	DMM	LRC	LadFG	FEMA	CMAP	BLDA
Ask-Ubuntu	0.647	0.671	0.734	-	**0.823**	0.706
Christianity	0.684	0.720	0.842	0.818	**0.856**	0.791

Table 9: Performance gains in Reputation Pred for most skewed/sparse dataset (Ask-Ubuntu) vs least (Christianity)

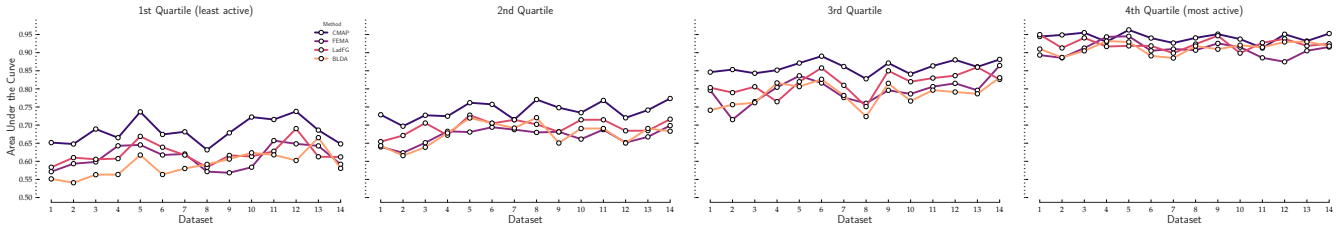


Figure 4: Effects of activity sparsity on prediction tasks (AUC) for Stack Exchanges (datasets 1-10) and MOOCs (datasets 11-14). CMAP has greatest performance gains in Quartile-1 (Sparse users), performance gap reduces for very active users (Quartile-4).

6.5 Parameter Sensitivity Analysis

Our model is primarily impacted by three parameter values: number of profiles R, number of behaviors K and discount parameter δ. We find results to be stable in a broad range of parameter values indicating that in practice our model requires minimal tuning (Figure 5). It is worth noting that while R primarily impacts the granularity of the discovered activity profiles, while K impacts the resolution of content-action associations. Dirichlet parameters and other hyper-parameters have negligible impact on the profiles and behaviors learned. We set $R = 20$, $\delta = 0.5$ and $K = 50$ for all datasets. Our inference algorithm is found to converge within 1% AUC in less than 400 sampling iterations across all datasets.

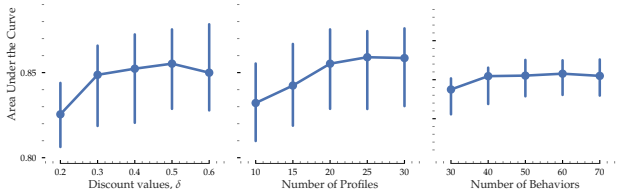


Figure 5: Mean performance(AUC) & 95% confidence interval with varying model parameters one at a time: δ, R, K. Stability is observed in broad ranges of parameter values.

6.6 Scalability Analysis

We compared the runtimes and memory consumption of our serial and batch-sampling (with 8 cores) inference algorithms with other models, for different volumes of interaction data obtained from random samples of the Ask-Ubuntu Stack-Exchange. BLDA is the fastest among the set of compared models. Our 8x batch sampler is comparable to BLDA in runtime. FEMA was the least scalable owing to the $|\mathcal{U}|^2$ growth of the User-User regularizer matrix. Figure 6 shows the comparisons between the algorithms.

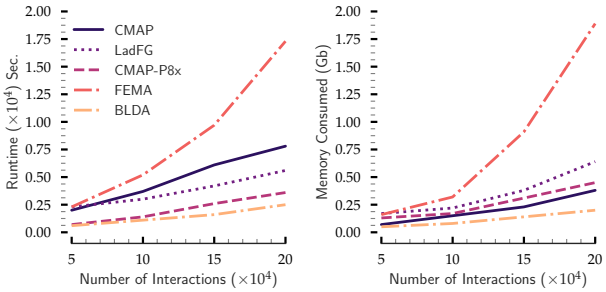


Figure 6: Effects of dataset size on algorithm runtime and memory consumption. BLDA is the fastest among the set of compared models.

6.7 Limitations

We identify two limitations. First, we make no assumptions about the structure of knowledge (e.g. a knowledge of "probability" is useful to understand "statistical models"); incorporating knowledge structure, perhaps in the form of an appropriate prior will help with better understanding participants with low activity. Second, we assume a bounded time range and our model is inapplicable on streaming data.

7 RELATED WORK

We categorize research related to our problem into four groups: Contextual Text Mining, Behavior Modeling, Temporal Behavior Dynamics, and Platform-specific work.

Contextual Text Mining: Generative models which combine contextual information with text have found success in generating discriminative combined trends. Topics Over Time [43] is a latent generative model over text and time-stamps of documents. Other temporal content models have been proposed [25, 45]. Link based models attempt to extract a static view of author communities and content [5, 21, 35]. Short-text approaches [31, 47] address content sparsity, absent publishing and consuming behaviors of users. Skew-aware models have also been developed for morphological structure analysis [10], topic modeling [16, 18, 37], dependency parsing [42] and query expansion [15, 22]. BLDA [29] is most closely related to our work since it attempts to integrate user actions with textual content, in the absence of a temporal factor.

Behavior Modeling: Matrix factorization has been a popular approach to study and predict human behavior. In the past, models have attempted to study two dimensional relations such as user-item affiliation [14, 49], and higher dimensional data via tensor models in web search and recommender systems [11, 39]. These models extract static views of user behavior.

Temporal Behavior Dynamics: This line of work integrates the temporal evolution of users with behavior modeling approaches. Previous works attempt to exploit historical user behavior data to predict future activity in online media [7, 20], recommender applications [8], and academic communities [44]. [28] attempts to build temporally-discretized latent representations of evolutionary learner behavior. These approaches do not explicitly address data sparsity at the user level. Jiang et al [13] have proposed a sparsity-aware tensor factorization approach to study user evolution. Their model however faces scalability challenges in massive real-world datasets, and relies on external regularizer data.

Platform-specific work: Characterization of users with generated content and actions has been studied in both settings, MOOCs [6, 24, 32] and community Question-Answering [12, 23]. [2] develops an engagement taxonomy for learner behavior. [4, 34] integrate social data to study dropout in MOOCs. In contrast to these approaches, our objective is to learn robust and generalizable representations to study participant behavior in diverse interactive social learning platforms.

8 CONCLUSION

In this paper, we address the challenge of characterizing user behavior on social learning platforms in the presence of data sparsity and behavior skew. We proposed a CRP-based Multi-facet Activity Profiling model (CMAP) to profile user activity with both, content interactions as well as social ties. Our experimental results on diverse real-world datasets show that we outperform state of the art baselines on prediction tasks. We see strong gains on participants with low activity, our algorithms scale well, and perform well even with de-skewing data.

We identify three rewarding future directions. Developing Incremental models for streaming data; incorporating priors to structure knowledge; allowing for infinite action spaces.

REFERENCES

[1] David J Aldous, Illdar A Ibragimov, and Jean Jacod. 2006. *Ecole d'Ete de Probabilites de Saint-Flour XIII, 1983*. Vol. 1117. Springer.

[2] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2014. Engaging with massive online courses. In *Proceedings of the 23rd international conference on World wide web*. ACM, 687–698.

[3] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *science* 286, 5439 (1999), 509–512.

[4] Jaroslav Bayer, Hana Bydzovská, Jan Géryk, Tomás Obsivac, and Lubomir Popelinsky. 2012. Predicting Drop-Out from Social Behaviour of Students. *International Educational Data Mining Society* (2012).

[5] Jonathan Chang and David M Blei. 2009. Relational topic models for document networks. In *International conference on artificial intelligence and statistics*. 81–88.

[6] Carleton Coffrin, Linda Corrin, Paula de Barba, and Gregor Kennedy. 2014. Visualizing patterns of student engagement and performance in MOOCs. In *Proceedings of the fourth international conference on learning analytics and knowledge*. ACM, 83–92.

[7] Peng Cui, Shifei Jin, Linyun Yu, Fei Wang, Wenwu Zhu, and Shiqiang Yang. 2013. Cascading outbreak prediction in networks: a data-driven approach. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 901–909.

[8] Peng Cui, Fei Wang, Shaowei Liu, Mingdong Ou, Shiqiang Yang, and Lifeng Sun. 2011. Who should share what?: item-level social influence prediction for users and posts ranking. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 185–194.

[9] Qiming Diao, Jing Jiang, Feida Zhu, and Ee-Peng Lim. 2012. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 536–544.

[10] Sharon Goldwater, Mark Johnson, and Thomas L Griffiths. 2006. Interpolating between types and tokens by estimating power-law generators. In *Advances in neural information processing systems*. 459–466.

[11] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. 2013. Personalized recommendation via cross-domain triadic factorization. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 595–606.

[12] Zongcheng Ji, Fei Xu, Bin Wang, and Ben He. 2012. Question-answer topic model for question retrieval in community question answering. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2471–2474.

[13] Meng Jiang, Peng Cui, Fei Wang, Xinran Xu, Wenwu Zhu, and Shiqiang Yang. 2014. Fema: flexible evolutionary multi-faceted analysis for dynamic behavioral pattern discovery. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1186–1195.

[14] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).

[15] Adit Krishnan, P Deepak, Sayan Ranu, and Sameep Mehta. 2017. Leveraging semantic resources in diversified query expansion. *World Wide Web* (2017), 1–27.

[16] Adit Krishnan, Aravind Sankar, Shi Zhi, and Jiawei Han. 2017. Unsupervised Concept Categorization and Extraction from Scientific Document Titles. *arXiv preprint arXiv:1710.02271* (2017).

[17] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*. ACM, 641–650.

[18] Robert V Lindsey, William P Headden III, and Michael J Stipicevic. 2012. A phrase-discovering topic model using hierarchical pitman-yor processes. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 214–222.

[19] Jun S Liu. 1994. The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem. *J. Amer. Statist. Assoc.* 89, 427 (1994), 958–966.

[20] Lu Liu, Jie Tang, Jiawei Han, Meng Jiang, and Shiqiang Yang. 2010. Mining topic-level influence in heterogeneous networks. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, 199–208.

[21] Yan Liu, Alexandru Niculescu-Mizil, and Wojciech Gryc. 2009. Topic-link LDA: joint models of topic and author community. In *proceedings of the 26th annual international conference on machine learning*. ACM, 665–672.

[22] Hao Ma, Michael R Lyu, and Irwin King. 2010. Diversifying Query Suggestion Results.. In *AAAI*, Vol. 10.

[23] Zongyang Ma, Aixin Sun, Quan Yuan, and Gao Cong. 2015. A Tri-Role Topic Model for Domain-Specific Question Answering.. In *AAAI*. 224–230.

[24] Jenny Mackness, Sui Mak, and Roy Williams. 2010. The ideals and reality of participating in a MOOC. In *Proceedings of the 7th International Conference on Networked Learning 2010*. University of Lancaster.

[25] Yasuko Matsubara, Yasushi Sakurai, B Aditya Prakash, Lei Li, and Christos Faloutsos. 2012. Rise and fall patterns of information diffusion: model and implications. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 6–14.

[26] Thomas Minka. 2000. Estimating a Dirichlet distribution. (2000).

[27] Jim Pitman and Marc Yor. 1997. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability* (1997), 855–900.

[28] Jiezhong Qiu, Jie Tang, Tracy Xiao Liu, Jie Gong, Chenhui Zhang, Qian Zhang, and Yufei Xue. 2016. Modeling and predicting learning behavior in MOOCs. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. ACM, 93–102.

[29] Minghui Qiu, Feida Zhu, and Jing Jiang. 2013. It is not just what we say, but how we say them: Lda-based behavior-topic model. In *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM, 794–802.

[30] Qiang Qu, Cen Chen, Christian S Jensen, and Anders Skovsgaard. 2015. Space-Time Aware Behavioral Topic Modeling for Microblog Posts. *IEEE Data Eng. Bull.* 38, 2 (2015), 58–67.

[31] Xiaojun Quan, Chunyu Kit, Yong Ge, and Sinno Jialin Pan. 2015. Short and Sparse Text Topic Modeling via Self-Aggregation.. In *IJCAI*. 2270–2276.

[32] Arti Ramesh, Dan Goldwasser, Bert Huang, Hal Daumé III, and Lise Getoor. 2013. Modeling learner engagement in MOOCs using probabilistic soft logic. In *NIPS Workshop on Data Driven Education*, Vol. 21. 62.

[33] Fatemeh Riahi, Zainab Zolaktaf, Mahdi Shafiei, and Evangelos Milios. 2012. Finding expert users in community question answering. In *Proceedings of the 21st International Conference on World Wide Web*. ACM, 791–798.

[34] Carolyn Penstein Rosé, Ryan Carlson, Diyi Yang, Miaomiao Wen, Lauren Resnick, Pam Goldman, and Jennifer Sherer. 2014. Social factors that contribute to attrition in MOOCs. In *Proceedings of the first ACM conference on Learning@ scale conference*. ACM, 197–198.

[35] Yiye Ruan, David Fuhry, and Srinivasan Parthasarathy. 2013. Efficient community detection in large networks using content and links. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 1089–1098.

[36] Hassan Saif, Yulan He, and Harith Alani. 2012. Alleviating data sparsity for twitter sentiment analysis. CEUR Workshop Proceedings (CEUR-WS. org).

[37] Issei Sato and Hiroshi Nakagawa. 2010. Topic models with power-law using Pitman-Yor process. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 673–682.

[38] Tanmay Sinha, Patrick Jermann, Nan Li, and Pierre Dillenbourg. 2014. Your click decides your fate: Inferring information processing and attrition behavior from MOOC video clickstream interactions. *arXiv preprint arXiv:1407.7131* (2014).

[39] Jian-Tao Sun, Hua-Jun Zeng, Huan Liu, Yuchang Lu, and Zheng Chen. 2005. Cubesvd: a novel approach to personalized web search. In *Proceedings of the 14th international conference on World Wide Web*. ACM, 382–390.

[40] Yee Whye Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 985–992.

[41] Yee Whye Teh. 2011. Dirichlet process. In *Encyclopedia of machine learning*. Springer, 280–287.

[42] Hanna Wallach, Charles Sutton, and Andrew McCallum. 2008. Bayesian modeling of dependency trees using hierarchical Pitman-Yor priors. In *ICML Workshop on Prior Knowledge for Text and Language Processing*. 15–20.

[43] Xuerui Wang and Andrew McCallum. 2006. Topics over time: a non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 424–433.

[44] Xiaolong Wang, Chengxiang Zhai, and Dan Roth. 2013. Understanding evolution of research themes: a probabilistic generative model for citations. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1115–1123.

[45] Jaewon Yang and Jure Leskovec. 2011. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 177–186.

[46] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. 2013. Lcars: a location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 221–229.

[47] Jianhua Yin and Jianyong Wang. 2014. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 233–242.

[48] Zhe Zhao, Zhiyuan Cheng, Lichan Hong, and Ed H Chi. 2015. Improving user topic interest profiles by behavior factorization. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1406–1416.

[49] Xiaodong Zheng, Hao Ding, Hiroshi Mamitsuka, and Shanfeng Zhu. 2013. Collaborative matrix factorization with multiple similarities for predicting drug-target interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1025–1033.