Insights from the Long-Tail : Learning Latent Representations of Online User Behavior in the Presence of Skew and Sparsity

Adit Krishnan
University of Illinois at Urbana-Champaign
aditk2@illinois.edu

Ashish Sharma*
Microsoft Research, India
t-asshar@microsoft.com

Hari Sundaram
University of Illinois at Urbana-Champaign
hs1@illinois.edu

ABSTRACT

This paper proposes an approach to learn robust behavior representations in online platforms by addressing the challenges of user behavior skew and sparse participation. Latent behavior models are important in a wide variety of applications: recommender systems; prediction; user profiling; community characterization. Our framework is the first to jointly address skew and sparsity across graphical behavior models. We propose a generalizable bayesian approach to partition users in the presence of skew while simultaneously learning latent behavior profiles over these partitions to address user-level sparsity. Our behavior profiles incorporate the temporal activity and links between participants, although the proposed framework is flexible to introduce other definitions of participant behavior. Our approach explicitly discounts frequent behaviors and learns variable size partitions capturing diverse behavior trends. The partitioning approach is data-driven with no rigid assumptions, adapting to varying degrees of skew and sparsity.

A qualitative analysis indicates our ability to discover niche and informative user groups on large online platforms. Results on User Characterization (+6-22% AUC); Content Recommendation (+6-43% AUC) and Future Activity Prediction (+12-25% RMSE) indicate significant gains over state-of-the-art baselines. Furthermore, user cluster quality is validated with magnified gains in the characterization of users with sparse activity.

KEYWORDS

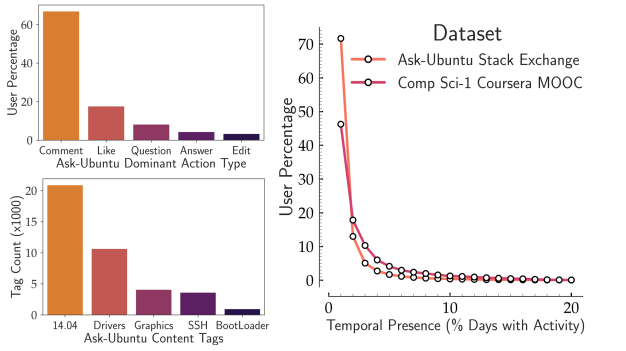
Behavior Analysis; Probabilistic Graphical Models; Behavior Skew; Data Sparsity; Interactive Media Platforms

ACM Reference Format:
Adit Krishnan, Ashish Sharma, and Hari Sundaram. 2018. Insights from the Long-Tail : Learning Latent Representations of Online User Behavior in the Presence of Skew and Sparsity. In *Proceedings of The 27th ACM International Conference on Information and Knowledge Management (CIKM '18).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3269206.3271706

1 INTRODUCTION

This paper addresses the challenge of learning robust statistical representations of participant behavior on online social networks.

Figure 1: Dominant Action Types and Content are highly skewed in Ask-Ubuntu, User presence exhibits steep power-law ($\eta \approx 3$) indicating several inconsistent or inactive users



Graphical behavior models have found success in several social media applications: content recommendation [16, 25], behavior prediction [17, 28], user characterization [11] and community profiling [5]. Despite the large sizes of these social networks (e.g. several million users), developing robust behavior profiles is challenging. We know from prior work [3] that activity on online networks is heavy tailed (a small set of users account for most interactions) with several temporally sparse users. Furthermore, user activity styles and topical interests are highly skewed (imbalanced) within the population, complicating the inference of prototypical behavior types. Figure 1 shows a typical example of behavior skew and temporal sparsity in `AskUbuntu`[1], a popular online Q&A forum.

Past works address one of the challenges (either sparsity or skew) separately in graphical behavior models, but do not adopt a unified approach to learn representations. Clustering is one common way to address sparsity [18, 24]. However, using clustering techniques in the presence of behavior skew can lead to uninformative results. For example, when topic models do not account for skew (e.g. Zipf's law), the resulting topics are less descriptive [19]. The use of suitable priors over the cluster sizes is a way to deal with skew. Beutel et al. [4] propose the use of the Pitman-Yor process [14] (visualized via Chinese Restaurant Process; CRP) to model skew in user data. However, a direct application of the CRP prior to behavior models cannot address sparsity. This is because behavior profiles are still learnt at the user-level and inactive users degrade the ability to learn robust latent representations; a lack of robust representations affects cluster quality.

Our main technical insight: We need to simultaneously address behavior skew and temporal sparsity of inactive users. While

*Work done during a summer internship at UIUC

[1] https://askubuntu.com/

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10…$15.00
https://doi.org/10.1145/3269206.3271706

we exploit the Pitman-Yor process (or CRP) as a prior, our key innovation in addressing sparsity and behavior skew lies in how we "seat" users onto tables. Our intuition is to associate inactive users with those active users to whom they were most similar, at the time these sparse users were active. Thus, to address sparsity we identify three concrete lines of attack: Profiles need to be learned from data at the granularity of a table (or equivalently, a group of users), *not* at the user-level; Behavioral similarity should guide user seating on these tables; We should discount common behavioral profiles to encourage identification of niche behaviors in the presence of skew. We refer to our model as CMAP (CRP-based Multi-Facet Activity Profiling) in the rest of this paper. To summarize our contributions:

- **Jointly address skew and sparsity:** To the best of our knowledge, this is the first work to *jointly* address behavior skew and sparsity with graphical behavior models. Our partitioning scheme can adapt to varying levels of behavior skew, effectively uncover fine-grained or niche behavior profiles, and address user-level sparsity.
- **Generalizability:** While in this work, we employ user activity and knowledge-exchanges, our framework generalizes well. The constituents of a behavioral profile can be easily adapted to new applications and platforms, while retaining skew and sparsity awarenesss in the learning process.
- **Efficiency:** Our model is efficient: the computational complexity is linear in the number of users and interactions. We employ caching optimizations to speed-up inference and scale to massive datasets with parallelized batch sampling.

We show strong quantitative and qualitative results on diverse datasets (public Stack-Exchange datasets and Coursera MOOCs[2]). We chose our datasets across technical/non-technical subject domains and varying population sizes, with all datasets seen to exhibit significant behavioral skew and sparsity (table 5). We evaluate CMAP against state-of-the-art baselines on three familiar task types: user characterization (reputation; certificate prediction on MOOCs), content recommendation and future activity prediction. Through extensive qualitative analysis, we find CMAP gains to be most significant for sparse users, and that the behavioral profiles learned are coherent and varied in size, capturing underlying behavioral skew. Our results have impact on the practical realities of large scale social network dataset analyses, since successfully addressing behavioral skew and sparsity is critical to familiar applications such as behavioral profiling and content recommendation.

We organize the rest of the paper as follows. In Section 2 we discuss related work. We formally define the problem and proposed approach in Section 3 and 4. We then discuss model inference, datasets and results in Sections 5, 6 and 7, concluding in Section 8.

2 RELATED WORK

At a high level, our motivations are shared with skew-aware topic models to improve document representation [19] by accounting for Zipf's law and short-text clustering methods [18, 27] to address content sparsity in text snippets. Graphical behavior models employ simple Dirichlet priors in user profile assignments [11, 16, 17]. However, this setting is limited in it's ability to model behavior skew and cannot cleanly separate niche and common behavior profiles. Our qualitative results (section 7.4) reflect this observation.

[2]https://stackexchange.com, https://coursera.org

Table 1: Comparing aspects addressed by baseline models with our proposed approach (CMAP)

Aspect	BLDA	LadFG	FEMA	CMAP
Skew-aware	No	No	No	Yes, via CRP
User-level Sparsity	No	No	External Regularizer	Profile-based Clustering
Multi-facet	Limited to Text/Action	Yes	Yes	Yes
Integrate with latent models	Limited to Text/Action	No	No	Yes
Runtime	Linear	Linear	Quadratic	Linear

In collaborative filtering, efforts have been made to transfer the user-item latent structure across platforms [7, 13] via consensus models to tackle sparsity. In the implicit feedback setting, this approach assumes alignment of user behavior across platforms. However, user interests and consumption trends vary not just by platform, but action-type and time as well [8, 29]. User anonymization (such as in MOOCs) can also pose difficulties in acquiring cross-platform data. We choose not to rely on external data.

Beutel et al. [4] propose a bayesian approach to group users with limited rating information and capture skewed product ratings. While the direct application of Pitman-Yor priors [14] to group users can capture skew in cluster sizes, it does not address the inactive user problem. In contrast, we factor in the latent behavior profiles in the seating to address sparsity via joint profiling of users [24]. The skew-aware partitioning and profile learning tasks are deeply coupled, unlike the superficial connection in past work.

Recently, Jiang et al [8] proposed sparsity-aware tensor factorization for user behavior analysis. User representations are regularized with external data such as author-author citations in academic networks, however not accounting for behavior skew. Behavior Factorization [29] simultaneously factorizes action-specific content affinities of users. Quadratic scaling imposes computational limits on these methods. Deep recurrent networks have also been explored to model temporal student behavior on MOOCs [15].

We choose FEMA [8] (Sparsity-aware Tensor Factorization), BLDA [16] (LDA-based generative user model) and LadFG [15] (Deep Recurrent network for temporal activity and attributes) as representative baselines for comparison with our approach. Table 2 provides a summary of the aspects addressed by each model.

3 PROBLEM DEFINITION

We apply our approach to learn representations of user behavior in multiple Coursera MOOCs and Stack-Exchange Q&A websites. The available facets of user activity include textual content, actions, time and inter-participant knowledge-exchanges.

Let $\mathcal{U}$ denote user set in a Stack-Exchange or MOOC dataset. Users employ a set of discrete actions $\mathcal{A}$ to interact with content generated from vocabulary $\mathcal{V}$. A user interaction d (atomic unit of participant activity) Is a tuple $d = (a, W, t)$, where the user performs action $a \in \mathcal{A}$ on content $W = \{w_1, w_2 \ldots \mid w_i \in \mathcal{V}\}$ at time-stamp $t \in [0, 1]$ (normalized over the time-span of the activity logs). We

denote the set of all interactions of $u \in \mathcal{U}$ as $\mathcal{D}_u$. Thus the collection of interactions in the dataset is $\mathcal{D} = \bigcup_{u \in \mathcal{U}} \mathcal{D}_u$. The action set for each dataset is described in table 4. Lecture interaction content for MOOC datasets is extracted from the respective transcripts.

Inter-participant knowledge-exchanges are represented by a directed multigraph $G = (\mathcal{U}, E)$. A directed labeled edge $(u, v, \ell) \in E$ represents an 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 label $\ell \in \mathcal{L}$ indicating the nature of the exchange (e.g. "answer" $\longrightarrow$"question"). We denote the set of all exchanges in which participant u is involved by L_u, so that $E = \bigcup_{u \in \mathcal{U}} L_u$.

Our goal is to obtain a set of temporal activity profiles R describing facets of user behavior, and infer user representations $\mathcal{P}_u, u \in \mathcal{U}$ as a mixture over the inferred behavior profiles $r \in R$.

4 OUR APPROACH

We begin in section 4.1 with intuitions to jointly address the behavior skew and sparsity challenges. In section 4.2, we describe a skew-aware user seating model guided by behavior profiles, concluding in section 4.3 with a description of our profile model.

4.1 Attacking the Skew-Sparsity Challenge

We begin by formally discussing the Pitman-Yor process [14] and then highlight challenges in the presence of sparsity.

Beutel et al. [4] employed the Pitman-Yor process via Chinese restaurant seating [1], as a simple prior over clusters to identify skewed user data trends. The conventional Chinese Restaurant arrangement induces a non-parametric prior over integer partitions (or indistinguishable entities), with concentration γ, discount δ, and base distribution G_0, to seat users across tables (partitions). Each user is either seated on an existing table $x \in \{1, \ldots, \chi\}$, or assigned a new table $\chi + 1$ as follows:

$$p(x \mid u) \propto \begin{cases} \frac{n_x - \delta}{N + \gamma}, & x \in \{1, \ldots, \chi\}, \text{ existing table,} \\ \frac{\gamma + \chi\delta}{N + \gamma}, & x = \chi + 1, \text{new table,} \end{cases} \tag{1}$$

where n_x is the user-count on existing tables $x \in \{1, \ldots, \chi\}$, $\chi + 1$ denotes a new table and $N = \sum_{x \in \{1, \ldots, \chi\}} n_x$ is the total user-count. A direct application of Equation (1) as a simple prior can address skew in profile proportions, but not sparsity. This is because, sparse users introduce noise into estimation of the corresponding behavioral profiles. To address sparsity, we need to find a way to "fill in the gaps" in our knowledge about inactive users.

Thus, to address sparsity we identify three concrete lines of attack: Profiles need to be learned from data at the granularity of a table (or equivalently, a group of users), *not* at the level of an individual; Behavioral similarity should guide seating on these tables; We should discount common behavioral profiles to encourage identification of niche behaviors and improve profile resolution.

4.2 Our Profile-Driven Seating

Now, we introduce our profile-driven seating approach that builds upon CRP to simultaneously generate partitions of similar users and learn behavior profiles describing users in these partitions. Consider a set of latent profiles $r \in R$ describing observed facets of user data with conditional likelihood $p(u \mid r)$ for $u \in \mathcal{U}$. We "serve" a profile $r_x \in R$ to users seated on each table $x \in \{1, \ldots, \chi\}$. A user u is seated on an existing table $x \in \{1, \ldots, \chi\}$ serving profile r_x or a new table $\chi + 1$ as follows,

Table 2: Notation for seating arrangement

Symbol	Description
N, R	Number of seated users, Set of profiles
$\{1, \ldots, \chi\}$, $\chi + 1$	Set of existing tables, New table
n_x, r_x	User count on table x, profile served on x
χ_r, N_r	Number of tables serving profile r, Total users seated on tables serving profile r

$$p(x \mid u) \propto \begin{cases} \frac{n_x - \delta}{N + \gamma} \times p(u \mid r_x), & x \in \{1, \ldots, \chi\}, \\ \frac{\gamma + \chi\delta}{N + \gamma} \times \frac{1}{|R|} \sum_{r \in R} p(u \mid r), & x = \chi + 1. \end{cases} \tag{2}$$

Note that the likelihood $p(x \mid u)$ of choosing an *existing* table $x \in \{1, \ldots, \chi\}$ for user u depends on the conditional $p(u \mid r_x)$ of the profile r_x served on the table and the number of users seated on table x. Further, the seating likelihoods for existing tables depend on the latent profiles served, while the latent profiles r_x are learned from the table x they are served on. This process introduces a mutual coupling between seating and profile learning. A larger setting of discount δ encourages the formation of new tables, leading to a preference on exploration over exploitation in profile learning.

The likelihood of assigning the user to a new table $x = \chi + 1$ depends on the sum of conditionals $p(u \mid r)$ with a uniform prior $\frac{1}{|R|}$, and the number of existing tables χ. Notice the effect of the discount factor δ: increasing δ favors exploration by forming new tables. Niche users are likely to be seated separately with a different profile served to them.

The main difference with basic CRP Equation (1), which partitions users based on the table size distribution, is that in our approach, we seat users based on the table size distribution, the profiles served on those tables, and the conditional probability of the user given behavioral profile. Equation (2) reduces to Equation (1) when all profiles $r \in R$ are equally likely for every user. We can show that our seating process is exchangeable, similar to [1]. The likelihood of a seating arrangement does not depend on the order in which we seat users on the tables.

When user u is seated on a new table $\chi + 1$, we draw profile variable $r_{\chi+1} \in R$ on the new table as follows:

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

where $p(r)$ is the Pitman-Yor base distribution G_0, or prior over the set of profiles. We set G_0 to be uniform to avoid bias.

The likelihood $p(r \mid u)$ of assigning profile r when seating user u, is proportional to the sum of likelihoods of seating the user on an existing table $x \in \{1, \ldots, \chi\}$ serving profile r (i.e. $r_x = r$), or seating on a new table $\chi + 1$ with the profile $r_{\chi+1} = r$. That is:

$$p(r \mid u) \propto \left(\sum_{\substack{x \in \{1, \ldots, \chi\}, \\ r_x = r}} \frac{n_x - \delta}{N + \gamma} p(u \mid r) \right) + \frac{1}{|R|} \cdot \frac{\gamma + \chi\delta}{N + \gamma} p(u \mid r), \tag{3}$$

$$\propto \left(\frac{N_r - \chi_r \delta}{N + \gamma} + \frac{\gamma + \chi\delta}{|R|(N + \gamma)} \right) p(u \mid r), \tag{4}$$

where χ_r is the number of existing partitions serving profile r and N_r is the total number of users seated on tables serving profile r.

Three insights stem from Equation (4). First, the skew in profile sizes depends on the counts of users exhibiting similar behavior patterns ($\propto p(u \mid r)$) enabling adaptive fits unlike [4]. Second, we discount common profiles served on multiple tables by the product $\chi_r \delta$. Since χ_r is larger for common profiles drawn on many tables, we discount common profiles more than niche profiles. This "common profile discounting" enables us to learn behavioral profile variations. Finally, not constraining the number of tables introduces stochasticity in profile learning and encourages exploration.

We assign users with limited activity to tables that well explain their data, biased by the priors in Equation (4). Our partitioning scheme assigns the *same* profile to users sharing a table, reducing the effect of inactive users since profiles describe behavioral groups.

In the next subsection, we introduce our temporal activity profiles $r \in R$ for representing user activity in our datasets.

4.3 Latent Profile Description

In this section, we formally define our behavioral profiles to describe user activity. We reiterate that our framework is flexible to other profile definitions depending on the requirements. In our datasets, behavioral profiles ($r \in R$) encode what actions users take (e.g. comment on a question), associated content (e.g. topic of the question), and when they take them. Furthermore, users participate in conversations (e.g. answer in response to a question), we term these directed links as "knowledge exchange."

Our profiles thus have two constituents: Joint associations of actions and words; referred to as "action-topics", and temporal distributions indicating when the action-topics are executed. Each action-topic $k \in K$ models user actions and the associated words, with $\phi_k^{\mathcal{V}}$ (multinomial over words with vocabulary $\mathcal{V}$) and $\phi_k^{\mathcal{A}}$ (multinomial over actions $\mathcal{A}$). Motivated by Wang and McCallum [23], we employ a continuous time model, Beta$(\alpha_{r,k}, \beta_{r,k})$ distributions, over a normalized time span to capture the temporal trend of each action-topic k within *each* profile r.

Thus for any interaction $d = (a, W, t)$, the probability $p(d \mid r, k)$ of a user interaction d given a profile r and topic k is:

$$p(d \mid r, k) \propto \underbrace{\phi_k^{\mathcal{A}}(a) \prod_{w \in W} \phi_k^{\mathcal{V}}(w)}_{\text{'what': profile independent}} \times \underbrace{\frac{t^{\alpha_{r,k}-1}(1-t)^{\beta_{r,k}-1}}{\mathrm{B}(\alpha_{r,k}, \beta_{r,k})}}_{\text{'when': profile dependent}}, \quad (5)$$

where B refers to the beta function. Notice that while the action-topics are shared between profiles, each profile r has a *different* temporal distribution associated with each action topic. Or simply, there are K action topics, but $R \times K$ temporal distributions. This modeling choice allows users with different overall behavioral profiles to participate in the same action topic at different times.

Since each behavioral profile r is a mixture over the K action topics and the associated temporal distributions, the likelihood $p(d \mid r)$ of user interaction d (as defined in section 3) for profile r is:

$$p(d \mid r) \propto \sum_k p(d \mid r, k) \times \phi_r^K(k), \quad (6)$$

where $\phi_r^K(k)$ is a K dimensional multinomial mixture over action-topics for each profile.

The next modeling step is to capture the exchange of knowledge between users. Instead of modeling it at the level of every pair of users, we model relationships between the pairs of profiles (r, r'), since every user is assigned to a single profile. This modeling choice is guided by sparsity. If we model every pair of users, we will develop a poor understanding of pairwise user interactions, owing to the heavy tailed activity distribution (i.e. most users contribute little; c.f. Figure 1).

Algorithm 1 Summary of the generative process to draw user data $\mathcal{D}_u$, L_u from profile $r \in R$ served on user's assigned table

for each action-topic $k \in K$ **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 activity profile $r \in R$ **do**
  Draw likelihood over action-topics, $\phi_r^K \sim Dir(\alpha_K)$
  for each profile $r' \in R$ **do**
    Draw knowledge-exchange likelihood $\phi_{r,r'}^{\mathcal{L}} \sim Dir(\alpha_{\mathcal{L}})$
for each content interaction $d = (a, W, t) \in \mathcal{D}_u$ **do**
  Choose action-topic $k \sim Multi(\phi_r^K)$
  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 time-stamp $t \sim Beta(\alpha_{r,k}, \beta_{r,k})$
for each inward exchange (s,u,ℓ) $\in L_u$ **do**
  Draw $\ell \sim Multi(\phi_{r_s,r}^{\mathcal{L}})$
for each outward exchange (u,y,ℓ) $\in L_u$ **do**
  Draw $\ell \sim Multi(\phi_{r,r_y}^{\mathcal{L}})$

We associate a label $\ell \in \mathcal{L}$ indicating the exchange type (e.g. Question → Answer, Comment → Answer etc.) between an ordered pair of users (u, v). To capture the knowledge exchange between profile pairs, we set-up $|R|^2$ multinomial distributions over exchange types $\phi_{r,r'}^{\mathcal{L}}$ between all ordered profile pairs (r, r').

Let L_u denote all exchanges for user u with other users v. Notice that sometimes u may initiate the exchange (e.g. ask a question) or respond (e.g. answer). Then, the likelihood $p(L_u \mid r)$ depends on the profiles being served to users involved in exchanges with u. Thus:

$$p(L_u \mid r) \propto \underbrace{\prod_{(s,u,\ell) \in L_u} \phi_{r_s,r}^{\mathcal{L}}(\ell)}_{\text{inbound exchange}} \times \underbrace{\prod_{(u,y,\ell) \in L_u} \phi_{r,r_y}^{\mathcal{L}}(\ell)}_{\text{outbound exchange}}, \quad (7)$$

where $\phi_{r_s,r}^{\mathcal{L}}(\ell)$ is the likelihood of an in-bound exchange from source user s served profile r_s, and $\phi_{r,r_y}^{\mathcal{L}}(\ell)$, for an out-bound exchange to user y served r_y.

The overall conditional likelihood $p(u \mid r)$ is the product of likelihood of exchanges $p(L_u \mid r)$ and likelihood of content interactions $p(d \mid r)$ of each user:

$$P(u \mid r) \propto p(L_u \mid r) \times \prod_{d \in D_u} p(d \mid r). \quad (8)$$

Algorithm 1 summarizes the generative process corresponding to Equation (8). We combine $p(u \mid r)$ from Equation (8) with $p(x \mid u)$ (Equation (2)) to seat users u on tables x, serving profile r_x.

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 times interactions of users served profile r was assigned topic k and marginal
$n_{r,r'}^{(\ell)}, n_{r,r'}^{(.)}$	Number of ℓ-labeled exchanges, all exchanges between users in tables serving r with r'

Table 3: Gibbs-sampler count variables

In this section, we identified challenges of using skew-aware partitions [4] when we have sparse users. Our intuition was to seat users based on their behavioral similarity, and to not learn profiles at the level of an individual, but that of a group. We discount common behaviors, encouraging identification of niche behavior. We introduced action-topics, and each profile is a mixture over these topics. Importantly, each profile learns a separate temporal distribution for each topic. Finally, we showed how user seating is guided by interaction between profiles—that is, users who behave similarly in their interaction with other groups are more likely to be seated together.

5 MODEL INFERENCE

In this section, we describe an efficient Gibbs-sampling approach for model inference, analyze its computational complexity and propose a parallel batch-sampling approach for speed-up. In each iteration of Gibbs-sampling, we unseat users one at a time and re-sample their seating as in Equation (2). Profile and Action-topic distributions are simultaneously updated, while hyper-parameters are modified between Gibbs iterations. We factor the seating sampler (Equation (12)) for efficiency since the number of tables is not fixed. We speed-up convergence times with coherent initial seating based on similar action distributions and content tags.

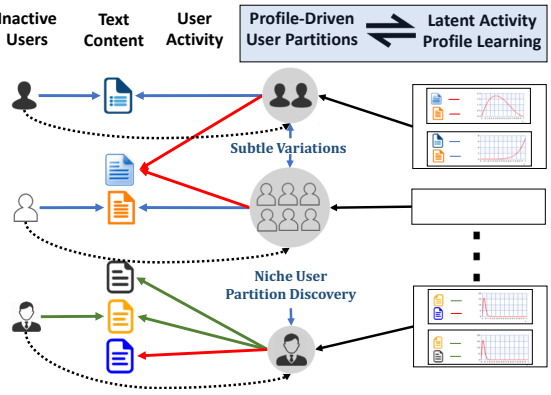


Figure 2: Our Gibbs-sampler simultaneously samples the seating arrangement of users (eq. 13) and learns profiles to describe the seated users (eq. 9, 10, 11). Users are grouped by behavioral similarity to overcome sparsity.

The likelihood of generating a user interaction $d = (a, W, t) \in \mathcal{D}_u$ conditional on action-topic $k \in K$ is:

$$p(a, W \mid k) \propto \frac{n_k^{(a)} + \alpha_{\mathcal{A}}}{n_k^{(.)} + |\mathcal{A}|\alpha_{\mathcal{A}}} \times \prod_{w \in W} \frac{n_k^{(w)} + \alpha_{\mathcal{V}}}{n_k^{(.)} + |\mathcal{V}|\alpha_{\mathcal{V}}}. \tag{9}$$

Thus, the likelihood $p(d \mid r)$ of interaction $d = (a, W, t)$ for a user served activity profile $r \in R$, Equation (6) is:

$$p(d \mid r) \propto \sum_{k \in K} \frac{n_r^k + \alpha_K}{n_r^{(.)} + |K|\alpha_K} \times p(a, W, t \mid k, r). \tag{10}$$

The likelihood that knowledge exchange occurs between profile pairs (r, r') on type ℓ is:

$$\phi_{r,r'}^{\mathcal{L}}(\ell) = \frac{n_{r,r'}^{\ell} + \alpha_{\mathcal{L}}}{n_{r,r'}^{(.)} + |\mathcal{L}|\alpha_{\mathcal{L}}}. \tag{11}$$

Thus, the conditional likelihood in Equation (8) can be obtained via Equation (10) over $\mathcal{D}_u$ and Equation (11) over L_u respectively. We can seat a user u either on an existing table $x \in \{1, \ldots, \chi\}$ serving profile r_x or on a new table $\chi + 1$; Equation (2), conditioned on the seating of all other users, denoted by x_{-u}. To avoid likelihood computation over all tables, we perform the draw in two factored steps. We first sample the profile served to u by marginalizing over tables via Equation (4),

$$P(r \mid u, x_{-u}) \sim \left(\frac{N_r - \chi_r \delta}{N + \gamma} + \frac{\gamma + \chi\delta}{|R|(N + \gamma)} \right) p(u \mid r), \tag{12}$$

and then sample from the set of tables serving the sampled profile (including the possibility of a new table with this profile draw),

$$P(x \mid r, u, x_{-u}) \sim \begin{cases} \frac{n_x - \delta}{N+\gamma} \times \mathbb{1}(r_x = r), & x \in \{1, \ldots, \chi\}, \\ \frac{\gamma + \chi\delta}{N+\gamma} \times \frac{1}{|R|}, & x = \chi + 1 \end{cases} \tag{13}$$

Note that $N = |\mathcal{U}| - 1$, i.e. all users except u. If we draw a new table $\chi + 1$, we assign the sampled profile variable r. We update all counts (Section 5) corresponding to prior profile and action-topic assignments for u.

We use well known techniques to update parameters. At the end of each sampling iteration, we update Multinomial-Dirichlet priors $\alpha_{\mathcal{V}}$, $\alpha_{\mathcal{A}}$, α_K and $\alpha_{\mathcal{L}}$ by Fixed point iteration [12]. We update Beta parameters $(\alpha_{r,k}, \beta_{r,k})$ by the method of moments [23]. We round all time-stamps to double-digit precision and we cache probability values $p(t \mid r, k)\ \forall\ t \in [0, 1], r \in R, k \in K$ at the end of each sampling iteration, thus avoiding $R \times K$ scaling for $p(u \mid r)$ in Equation (12). While we fix the Pitman-Yor parameters in our experiments for simplicity, if needed, we can estimate them via auxiliary variable sampling [19, 20].

Computational Complexity. Our inference is linear in the number of users $|\mathcal{U}|$ and interactions $|\mathcal{D}|$, scaled by $R+K$ (see empirical results in Figure 6). To see this, notice that in each Gibbs iteration, computing Equations (9) and (10) involves $|\mathcal{D}| \times (K + R)$ computations. Equation (12) requires an additional $|\mathcal{U}| \times R$ computations. We prevent $R \times K$ scaling for $p(u \mid r)$ in Equation (12) by caching. We cache the first product term of Equation (12) for each $r \in R$, and update it only when there is a change in the seating arrangements on tables serving profile r.

Parallelization with Batch Sampling. We scale to massive datasets by parallelizing inference via batch-sampling. The Gibbs sampler described above samples each user's seating $P(x_u \mid u, x_{-u})$ in Equation (13), conditioned on all other users. This 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. $P(x_U \mid U, x_{\mathcal{U}-U})$ where x_U denotes the table assignments to users in batch U. For efficiency, batches must be chosen with comparable computation. We approximate computation for $u \in \mathcal{U} \propto |\mathcal{D}_u| + |\mathcal{L}_u|$ to decide apriori batch splits for sampling iterations. Note that when batch sampling, we can only exploit knowledge exchange links between users in the batch and users *not* in the batch. In practice, since $|U| \ll |\mathcal{U}|$, the impact of not using knowledge exchanges between users in the same batch turns out to be negligible.

6 DATASET DESCRIPTION

We now provide a brief description of the Coursera MOOC and Stack-Exchange datasets that we use in our experiments and characterize them in terms of skew and sparsity.

Stack-Exchanges are community Q&A websites where participants discuss a wide range of topics. Users interact with each other and perform a range of actions (e.g. post question, answer, comment etc.). We experiment on 10 Stack-Exchanges, chosen for thematic diversity and size variation. Coursera MOOCs feature video lectures for students to watch and a forum where students and instructors can interact. We analyze the actions (e.g. play, skip, rewind etc.) on the videos, lecture content via subtitles, and the forum interaction for four MOOCs, chosen for thematic diversity. The user action types and datasets are summarized in Table 4 and Table 5 respectively.

To get a feel for these datasets, let us examine sparsity and behavior skew. To understand sparsity, we compute the power-law ($f_w = c.w^{\eta_t}$) exponent η_t that best describes the fraction of users f_w who were active for w-weeks. A more negative index indicates that fewer users are consistently active. As a reference point, when $\eta_t = 0$, a constant fraction of users are always active. Thus when we notice that $\eta_t = -2.81$ for `Ask Ubuntu` Stack Exchange in Table 5, it means that the number of users who are active for two weeks is just 14% of those active for one week. Table 5 indicates that larger Stack Exchanges tend to have greater sparsity.

Platform	Action	Description
Coursera 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 forum thread for inquiries
	Post	Reply to existing threads
	Comment	Comment on existing posts
Stack-Exchange	Question	Posting a question
	Answer	Authoring answer to a question
	Comment	Comment on a question/answer
	Edit	Modify posted content
	Follow	Following posted content

Table 4: User Action Description (Coursera/Stack-Exchange)

Table 5: Preliminary analysis indicates significant behavior skew and inactive user proportion, although slightly reduced in specialized domains like Christianity

Platform	Dataset	Users	Interactions	η_t	S_N
Coursera MOOC	Comp Sci-1	26,542	834,439	-2.51	0.67
	Math	10,796	162,810	-2.90	0.69
	Nature	6,940	197,367	-2.43	0.70
	Comp Sci-2	10,796	165,830	-2.14	0.73
Stack-Exchange	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

We measure skew by first identifying the dominant action type or style of each user (e.g commenter, editor) and then compute the normalized entropy S_N of the resulting user distribution. In a large Stack-Exchange such as `Ask-Ubuntu`, while less than 5% (c.f Figure 1) of the users have 'Answer' as their dominant type, over 60% of the users have 'Comment' as their dominant action. This does not consider content topics, which result in even greater skew. When $S_N = 1$, all dominant action types are equally likely; in contrast $S_N = 0$ indicates a single dominant action type. In MOOCs, 'Play' is the dominant action type with low forum participation (participation rates ~10-15% in our MOOC forums).

7 EXPERIMENTAL RESULTS

In this section, we present extensive quantitative and qualitative analysis of our model. We begin by introducing baseline methods (Section 7.1), followed by prediction tasks undertaken (Section 7.2), and present impressive quantitative results for CMAP in Section 7.3. Then in Section 7.4, we qualitatively analyze the reasons for CMAP's gains over baselines. In Section 7.5 we examine a counterfactual: *what if the data had less skew?* Finally, we analyze scalability (Section 7.6), parameter sensitivity (Section 7.7) and discuss limitations in Section 7.8.

7.1 Baseline Methods

We compare our model (CMAP) with user representations from three state-of-the-art models and two standard baselines. We list the baselines below.

LadFG [15]: LadFG is a deep recurrent approach to learn behavior representations from temporal activity and demographic information of users. We provide LadFG action-content data from interactions and all available user demographic information.

BLDA [16]: BLDA is an LDA-based extension to capture latent associations of user actions and content. It represents users as a mixture over these content-action topics.

FEMA [8]: FEMA is a multifaceted sparsity-aware tensor factorization approach employing external regularizers for smoothing. Facets in our datasets are users, words and actions. We set User

and Word regularizers to their exchanges and co-occurrence counts respectively. We could not run FEMA on Ask-Ubuntu and Comp Sci-1 datasets due to very high memory and compute requirements (Regularizer matrices in FEMA scale quadratically $O(|\mathcal{U}|^2)$).

DMM (Only text) [27]: We apply DMM to the textual content of all interactions to learn topics. We represent users by the proportions of topics in their interaction content.

Logistic Regression Classifier (LRC) [10]: Logistic regression based classification model. Input features are DMM topics that the user interacts with and actions in each topic (Answer, Edit etc.).

We construct user representations for models as follows: For CMAP (Ours), we use the $|R|$-dimensional normalized conditionals $P(r \mid u)$ for each user as given by Equation (4); For BLDA, we use normalized conditionals over the set of behaviors for each user as computed by the authors [16]; for FEMA, we use respective rows of user projection matrix, $\mathbf{A}_t$ [8]; for LadFG, we use latent user embeddings learned upon training; for DMM, we use topic proportions for user generated text. We use LRC only for prediction tasks, as it does not build user representations.

For fair comparison, the user representations from baselines were the same dimensionality as the profile count $|R|$ for our model. We use $|R| = 20$ and 40 Action-Topics ($|K|$) for all datasets. We initialize Dirichlet priors as: $\phi_k^{\mathcal{V}}, \phi_k^{\mathcal{A}}, \phi_r^K$ and $\phi_{r,r'}^{\mathcal{L}}$ with the common strategy [6, 9, 26] ($\alpha_X = 50/|X|, X = \{\mathcal{A}, \mathcal{L}, K\}$, and $\alpha_{\mathcal{V}} = 0.01$) and *Beta* parameters α_{rk}, β_{rk} to 1. CRP parameter initialization $\delta = 0.5, \gamma = 1$ performed well consistently. Our experiments were performed on a single x64 XSEDE compute node[3] [21] (Intel Xeon E5-2680v3, 64 GB Memory). Our implementations are available online[4].

7.2 Prediction Tasks

We identify three distinct task types for evaluating the quality of user representations across methods. We focus on two User Characterization tasks, a Future Activity Prediction task, and Question Recommendation in Stack-Exchanges. Below, we list the tasks.

User Characterization (MOOC) - Certificate Earner: Coursera awards certifications to students maintaining high cumulative grades over assignments. We predict students obtaining certificates with the user representations obtained from each model.

User Characterization (Stack-Exchange) - Reputed User: For Stack-Exchanges, we predict if participants have a high reputation with user representations from each model. We define users in a Stack Exchange to have high reputation if they lie in the top quartile (25%) of all reputation scores.

Question Recommendation (Stack-Exchange): For popular questions in Stack-Exchanges, we identify suitable users to answer them. In each dataset, we choose a set of 100 held-out popular questions & learn user representations by applying models to their remaining activity. We then perform 5-fold Cross-Validation for each held-out question with the known users who answered the question and equal number of negative users chosen at random.

Future Activity Prediction (All Datasets): We obtain topic assignments for user interactions with DMM [27] ($T = 20$). For each user, we predict their future activity mixture over topics & actions given user representations with their past activity from each model (6-month data held-out). LRC is not used in Future Activity Prediction as it does not build a user representation.

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

[4]https://github.com/ash-shar/CMAP

Table 6: Reputed User Prediction ($\mu \pm \sigma$ across Stack-Exchanges). We obtain improvements of 6.65-21.43% 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
BLDA	0.75 ± 0.04	0.71 ± 0.04	0.74 ± 0.03	0.74 ± 0.04
CMAP	0.85 ± 0.02	**0.83 ± 0.03**	**0.84 ± 0.02**	**0.86 ± 0.02**

We use standard classifiers and evaluation metrics. Characterization and Recommendation use linear-kernel SVM evaluated with Precision, Recall, F1-Score and Area-Under-Curve (AUC). Future Activity Prediction uses Linear Regression. Both were implemented with default parameters in sklearn [5]. For the activity prediction task, we measure the Root Mean Squared Error (RMSE) in predicted activity proportions for *(topic, action)* pairs against actual proportions of users in the held-out future activity. We compute results with 5-fold cross-validation for each dataset. Statistically significant gains (Paired t-test, $p < 0.05$) are in bold.

7.3 Results

We examine the experimental results for each of the three tasks—User characterization, question recommendation and future activity prediction in this section.

Our method improves on the baselines in the *reputation prediction task* by 6.26-15.97% AUC averaged across the Stack-Exchanges; Table 6 shows the results with statistically significant improvements in bold. LadFG performs slightly better on the overall precision in reputation prediction (not statistically significant), likely due to over-fitting of the embeddings to user-level data resulting in low recall. Our ability to discover more distinct user clusters even with the same latent dimensions as baselines (refer fig. 5) is the main reason for our gains in predicting reputation. Similarly, we improve on *certification prediction* (see Table 7) by 6.65-21.43% AUC averaged over MOOCs.

For the *question recommendation task* Table 8, we see gains between 6-47% AUC over the baselines. To do well in this task, we require the model to make finer distinctions between the topical preferences of users; user reputation and action style are also important in this task.

For the *future activity prediction task*, our method shows gains over baselines in RMSE by 12%-25% on MOOCs and between 9.5%-22% on Stack-Exchanges; (see Table 9). Gains are explained by our

[5]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
BLDA	0.80 ± 0.04	0.75 ± 0.03	0.77 ± 0.03	0.77 ± 0.04
CMAP	**0.86 ± 0.02**	**0.81 ± 0.03**	**0.83 ± 0.02**	**0.84 ± 0.02**

Table 7: Certificate Earner Prediction ($\mu \pm \sigma$ across MOOCs); CMAP improves upon baselines by 6.65-21.43% AUC

Table 8: Question Recommendation ($\mu \pm \sigma$ across Stack-Exchanges)with 6.30-47.45% AUC gains for CMAP. DMM performs quite well owing to importance of content in this task.

Method	Precision	Recall	F1-score	AUC
LRC	0.65 ± 0.06	0.57 ± 0.08	0.60 ± 0.06	0.57 ± 0.05
DMM	0.72 ± 0.04	0.81 ± 0.05	0.75 ± 0.04	0.74 ± 0.04
LadFG	0.88 ± 0.03	0.60 ± 0.02	0.71 ± 0.02	0.76 ± 0.04
FEMA	0.79 ± 0.05	0.73 ± 0.06	0.77± 0.05	0.79 ± 0.03
BLDA	0.70 ± 0.04	**0.84 ± 0.04**	0.77 ± 0.03	0.75 ± 0.04
CMAP	**0.89 ± 0.03**	0.81 ± 0.02	**0.85 ± 0.03**	**0.84 ± 0.02**

model's ability to make finer distinction on action styles, and to be able to make better distinctions between profiles assigned to users.

In this section, we showed impressive performance gains on three types of tasks for our model over state of the art baselines. In the next section, we qualitatively analyze the reasons for it's success.

7.4 Why does CMAP work well?

To interpret the gains obtained by CMAP, we examine the extracted clusters in Section 7.4.1 and then look at users responsible for the performance gains of our model in Section 7.4.2.

7.4.1 ***The Impact of Profile Driven Seating.*** We now compare clusters obtained through CMAP seating against conventional generative assignments in BLDA [16] on Stack-Exchanges. Both models group users best described by the same profile to form clusters. We use average user reputations of the clusters (appropriately normalized) as an external validation metric for cluster quality. We also run our model excluding time and knowledge-exchanges to see the effect on the clusters. Figure 3 shows the result from the `Ask-Ubuntu` Stack Exchange, and Table 10 shows the main activities and topics of the top three CMAP clusters. We make the following key observations from the clusters:

The Mean-Shift problem: The Dirichlet-Multinomial setting in BLDA tends to merge profiles and hence shift cluster sizes and average participant reputation closer to the mean. Figure 3 shows that 15 of 20 BLDA clusters have nearly the same size and average reputation. Both variants of CMAP show diversity in cluster size and high reputation variability across tables. Our cluster assignments appear to mirror the behavior skew for `Ask-Ubuntu` (c.f. Figure 1).

Profile Quality: CMAP learns finer variation in the topic affinities and actions of expert users. We can observe these variations from Figure 3 and from Table 10. The top three profiles are of higher reputation, smaller in size, and from Table 10, each of these clusters shows distinct activities different from the mean activity. CMAP clusters appear to better reflect skewed user activity (c.f. Table 5) and content preference (c.f. Figure 1) with flexible profile-driven seating.

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

Table 9: Future Activity Prediction (RMSE ($\times 10^{-2}$) $\mu \pm \sigma$), Lower RMSE is better. CMAP ouperforms baselines in MOOCs (12%-25%) and Stack-Exchanges (9.5%-22%)

We observe a similar trend in the aggregated clusters obtained from all the other Stack-Exchange datasets (c.f. Figure 5). The Dirichlet-Multinomial setting in BLDA results in similarly sized clusters which cannot model highly skewed content and action affinities of users. Note the fewer high-reputation clusters of BLDA in comparison to the finer distinctions of reputed users in our model. Our performance in prediction and recommendation reflect these observations, we see significant gains in our ability to characterize reputed users and recommend suitable content (Section 7.2).

7.4.2 ***Making gains on inactive users.*** We now investigate the source of our gains. We split users in each Stack-Exchange and MOOC into four quartiles based on interaction count (Quartile 1 is least active, 4 most). Then, we evaluate each method on Reputation and Certificate Prediction AUC in each quartile of Stack Exchange and MOOC datasets respectively.

Our model shows large gains (Figure 4) in Quartiles 1,2 that contain sparse users. We attribute these gains due to our joint profile learning to describe similar users seated on tables. The decision to address skew and sparsity jointly has two advantages: better profile fits for sparse users; more distinct and informative profiles in skewed scenarios. In contrast, models building representations at the user level perform weakly in Quartiles-1,2 since these methods rely on interaction volume. We make smaller gains in Quartiles 3,4.

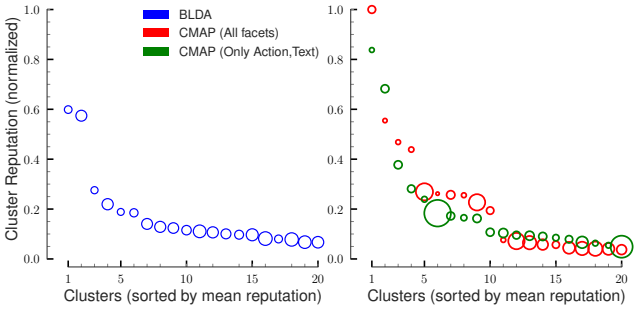


Figure 3: Bubbles denote user clusters discovered by each model in the Ask-Ubuntu dataset (Bubble size $\propto$ Users in Cluster). CMAP discovers fine distinctions of reputed users (Profiles 1,2,3,4) by content preference and activity (Table 10). BLDA clusters are mean-sized and close to the population average in reputation. In contrast, our assignments better reflect the behavior skew of users in the dataset.

Cluster	Action Style	Common Topics
1	+31% Answer, +24% Edits, -9% Questions	Graphics Drivers, Booting Issues, Disk Partitions
2	+67%Answer, -3% Edits, -21% Questions	Gnome, Desktop, Package Install
3	+11% Answer, -4% Edits, +47% Questions	Script, Application, Sudo Access

Table 10: Action and Content description of users in top-3 clusters discovered by CMAP in Ask-Ubuntu, +/- values of action proportions against the average Ask-Ubuntu user

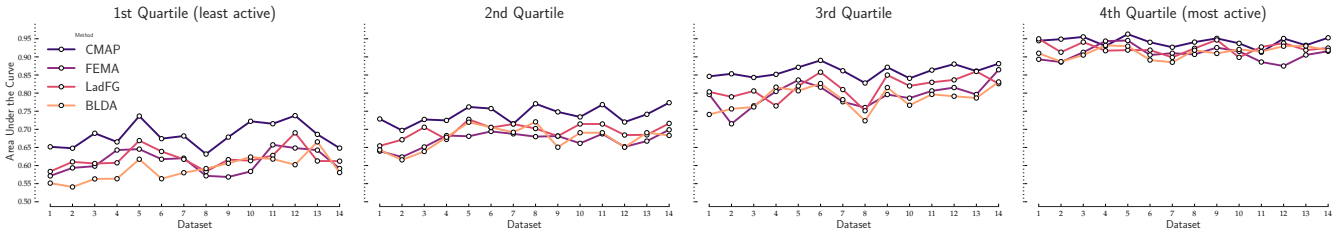


Figure 4: Effects of activity sparsity on prediction tasks (AUC) for Stack Exchanges (datasets 1-10) and MOOCs (datasets 11-14). CMAP has greatest performance gains in Quartile-1 (Sparse users), performance gap reduces for very active users (Quartile-4).

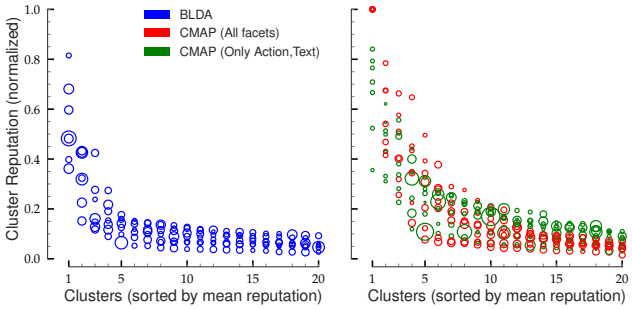


Figure 5: Bubbles denote clusters in other Stack-Exchanges (Bubble size $\propto$ Users in Cluster). CMAP discovers highest reputation clusters in all datasets (Thick red dot, Top-left). BLDA clusters tend to mean reputation, size (Mean-Shift) not capturing disparities. In our case, profiles 1,2,3,4 appear to capture niche, highly reputed user behaviors

To summarize: jointly addressing sparsity and skew by profile driven seating is responsible for our gains. Importantly, the clusters are coherent, the model learns fine distinctions in behavioral profiles and exhibits behavior skew found in the underlying data.

7.5 What if there was less Skew?

In this section, we study a counterfactual: what if the real-world datasets were less skewed? To study this question, we sub-sample users who predominantly perform the two most common actions in our largest datasets, `Ask-Ubuntu` (Comments and Questions) and `Comp Sci-1 MOOC` (Play and Skip). These users are sub-sampled by half while retaining all other users, hence reducing overall action skew in the data. Baseline models are expected to perform better with reduced skew. All models degrade in Ask-Ubuntu owing to significant loss of content.

Table 11 shows that CMAP still maintains a lead owing to inactive users. We also investigate performance gains in a highly skewed and sparse Stack-Exchange (Ask-Ubuntu) vs least skewed (Christianity)

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**	0.849

Table 11: CMAP outperforms baselines (AUC) in the de-skewed datasets, but with smaller gains

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 12: We see greater gains for User Characterization in a high-skew dataset (Ask-Ubuntu) vs low-skew (Christianity)

in Table 12. On average, we outperform baselines by 13.3% AUC for Ask-Ubuntu vs 10.1% for Christianity Stack-Exchange in User Characterization.

7.6 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 $O(|\mathcal{U}|^2)$ growth of the User-User regularizer matrix. Figure 6 shows the comparisons between the algorithms.

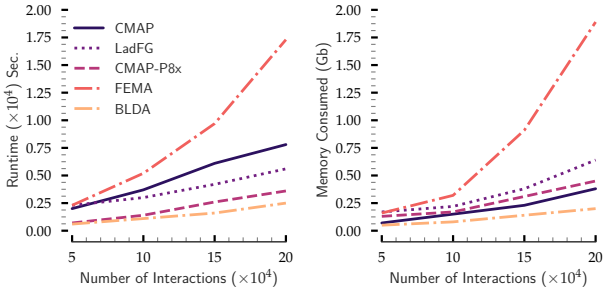


Figure 6: Effects of dataset size on algorithm runtime and memory consumption. BLDA is the fastest among the set of compared models.

7.7 Parameter Sensitivity Analysis

Our model is primarily impacted by three parameter values: number of profiles R, number of Action-Topics K and discount δ. We find results to be stable in a broad range of parameter values indicating that in practice our model requires minimal tuning (Figure 7). It is worth noting that while R primarily impacts the granularity of the discovered activity profiles, K impacts the resolution of content-action associations. Dirichlet and other hyper-parameters have negligible impact on the profiles and seating arrangement learned. Our inference algorithm converges within 1% AUC in less than 400 sampling iterations across all datasets.

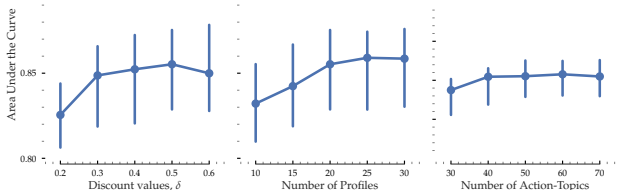


Figure 7: Mean performance(AUC) & 95% confidence interval with varying model parameters one at a time: δ, R, K. Stability is observed in broad ranges of parameter values.

7.8 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. Development of latent profiles for streaming activity can lead to deployment with real-time data.

It would be an interesting exercise to observe the effect of dynamic updates to Pitman-Yor hyper-parameters over sampling iterations [20]. Although such an approach has been explored for LDA [22], it is unclear how over-fitting in our approach can be avoided in case of hyper-parameter drift. We plan this study for future work.

8 CONCLUSION

This paper proposes an approach to learn descriptive statistical representations of user behavior on interactive platforms, robust to behavior skew and sparsity. Our framework is the first to jointly address skew and sparsity across graphical behavioral models. Our main technical contribution: jointly partition users using a Pitman-Yor process prior and assign temporal behavioral profiles to each partition. User profile models can be flexibly chosen depending on requirements. Extensive experiments over large online forums reveal the informativeness of our behavior profiles in diverse recommendation and profiling tasks. A qualitative analysis indicates our ability to discover niche and discriminative user groups and strong gains for sparse participants. Furthermore, our algorithms scale linearly and do not require supervision or auxiliary data.

We identify three rewarding future directions to enhance applications of our model. Developing Incremental models for streaming data could enable application to real-time online platforms. Incorporating knowledge priors on expected behavior patterns (e.g. how students without a strong probability might test in an IR class) in the context of the MOOC platform [2] is also a possible future direction. Finally, we plan to incorporate continuous user action spaces, which is particularly challenging in skewed and sparse settings.

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-Laszlo Barabasi. 2005. The origin of bursts and heavy tails in human dynamics. *Nature* 435, 7039 (2005), 207–211.

[4] Alex Beutel, Kenton Murray, Christos Faloutsos, and Alexander J Smola. 2014. Cobafi: collaborative bayesian filtering. In *Proceedings of the 23rd international conference on World wide web*. ACM, 97–108.

[5] Hongyun Cai, Vincent W Zheng, Fanwei Zhu, Kevin Chen-Chuan Chang, and Zi Huang. 2017. From community detection to community profiling. *Proceedings of the VLDB Endowment* 10, 7 (2017), 817–828.

[6] 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.

[7] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. 2015. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 278–288.

[8] 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.

[9] Adit Krishnan, Ashish Sharma, and Hari Sundaram. 2017. Improving Latent User Models in Online Social Media. *arXiv preprint arXiv:1711.11124* (2017).

[10] 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.

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

[12] Thomas Minka. 2000. Estimating a Dirichlet distribution.

[13] Weike Pan, Evan Wei Xiang, Nathan Nan Liu, and Qiang Yang. 2010. Transfer Learning in Collaborative Filtering for Sparsity Reduction.. In *AAAI*, Vol. 10. 230–235.

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

[15] 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.

[16] 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.

[17] 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.

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

[19] 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.

[20] 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.

[21] John Towns, Timothy Cockerill, Maytal Dahan, Ian Foster, Kelly Gaither, Andrew Grimshaw, Victor Hazlewood, Scott Lathrop, Dave Lifka, Gregory D Peterson, et al. 2014. XSEDE: accelerating scientific discovery. *Computing in Science & Engineering* 16, 5 (2014), 62–74.

[22] Hanna M Wallach, David M Mimno, and Andrew McCallum. 2009. Rethinking LDA: Why priors matter. In *Advances in neural information processing systems*. 1973–1981.

[23] 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.

[24] Gui-Rong Xue, Chenxi Lin, Qiang Yang, WenSi Xi, Hua-Jun Zeng, Yong Yu, and Zheng Chen. 2005. Scalable collaborative filtering using cluster-based smoothing. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 114–121.

[25] Chunfeng Yang, Huan Yan, Donghan Yu, Yong Li, and Dah Ming Chiu. 2017. Multi-site User Behavior Modeling and Its Application in Video Recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 175–184.

[26] 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.

[27] 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.

[28] Zhijun Yin, Liangliang Cao, Jiawei Han, Chengxiang Zhai, and Thomas Huang. 2011. Geographical topic discovery and comparison. In *Proceedings of the 20th international conference on World wide web*. ACM, 247–256.

[29] 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.