A Collective Bayesian Poisson Factorization Model for Cold-start Local Event Recommendation

Wei Zhang[†]   Jianyong Wang[†♮]

[†]Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[♮]Jiangsu Collaborative Innovation Center for Language Ability, Jiangsu Normal University, Xuzhou 221009, China
zwei11@mails.tsinghua.edu.cn, jianyong@tsinghua.edu.cn

ABSTRACT

Event-based social networks (EBSNs), in which organizers publish events to attract other users in local city to attend offline, emerge in recent years and grow rapidly. Due to the large volume of events in EBSNs, event recommendation is essential. A few recent works focus on this task, while almost all the methods need that each event to be recommended should have been registered by some users to attend. Thus they ignore two essential characteristics of events in EBSNs: (1) a large number of new events will be published every day which means many events have few participants in the beginning, (2) events have life cycles which means outdated events should not be recommended. Overall, event recommendation in EBSNs inevitably faces the cold-start problem.

In this work, we address the new problem of cold-start local event recommendation in EBSNs. We propose a collective Bayesian Poisson factorization (CBPF) model for handling this problem. CBPF takes recently proposed Bayesian Poisson factorization as its basic unit to model user response to events, social relation, and content text separately. Then it further jointly connects these units by the idea of standard collective matrix factorization model. Moreover, in our model event textual content, organizer, and location information are utilized to learn representation of cold-start events for predicting user response to them. Besides, an efficient coordinate ascent algorithm is adopted to learn the model. We conducted comprehensive experiments on real datasets crawled from EBSNs and the results demonstrate our proposed model is effective and outperforms several alternative methods.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering, Retrieval Models*

General Terms

Algorithms, Experimentation

Keywords

Event Recommendation, Bayesian Poisson Factorization, Event-Based Social Networks, Cold-start Recommendation

1. INTRODUCTION

Along with the trend of combining online and offline interactions among users in the mobile Internet era, event-based social networks (EBSNs) have emerged in recent years and enjoyed a booming development. Meetup[1] and Douban Event[2] are two standard EBSNs which prove to be widely used by many users. The core goal of EBSNs is to gather neighbors (users located in the same city) together to do what they are commonly interested in.

Among all the elements in EBSNs, event is the most significant one which bridges the gap of online and offline interaction. Formally, a event consists of the following core elements: 1) content, which provides introduction of the event theme, 2) organizer, who launches and organizes the event, 3) location, where the event will be held, and 4) time, when the event will start. As users always prefer to participate in the events nearby [28], many EBSNs divide events by cities and provide users with the events which are located in the same city to attend. Due to a large volume of events, personalized event recommendation is essential for avoiding the information overload problem. Moreover, it is beneficial for EBSNs as the better user experience can attract more users to register on their websites. Although various recommendation problems have been studied in the last decade, only a few recent works study event recommendation in EBSNs. Moreover, event recommendations implemented in popular EBSNs are very simple for only ranking events by their popularity, chronological order, and location distance from users.

Two most recent works [21, 4] are proposed to explore this task. Both of them have one essential assumption that each event has already been registered by some users for attending. Based on this assumption, they further directly associate each event with a latent factor and learn it from corresponding users' participation records. Nevertheless, this assumption does not conform to the real scenario of event recommendation in EBSNs since it ignores the fact that events

---

[1]http://www.meetup.com
[2]http://beijing.douban.com/events

have life cycles. The outdated events whose starting time are past should be removed from the event candidate list. Besides, many new events which are published in a short time are registered by only a few users or not even one. They account for a certain proportion in all candidates without the outdated events. As a consequence, event recommendation in EBSNs inevitably faces a serious cold-start challenge [24].

To address the above issues, we formulate a new problem called cold-start local event recommendation in event-based social networks. The substantial distinction from the task studied by [21, 4] is that our problem concentrates on cold-start event recommendation with each candidate event having no registered users for attending. Thus new event recommendation results can be generated as soon as the events are published. The main challenge lies in how to learn the representations of the cold-start events without interaction behaviors with users. For overcoming this challenge, we propose a Collective Bayesian Poisson Factorization (CBPF) model. CBPF combines the merits of Bayesian Poisson factorization (BPF) [6] and collective matrix factorization (CMF) [25].

First of all, it takes Bayesian Poisson factorization as its basic unit and each unit is responsible for reconstructing different types of data when modeling. In our problem setting, user response, social relation, and event content text should be modeled. Then CBPF jointly connects these units to construct a unified model inspired by the idea of collective matrix factorization. To learn representations of cold-start events, it associates each event with its content introduction, organizer, and location information. Thus the unit of modeling user response is more complex than BPF as it involves interaction betweens factors of user, event content, organizer, and location. An efficient coordinate ascent algorithm is adopted and corresponding parameter update formulas are derived for CBPF. After the model learning stage, optimal factors except content factor of new events can be acquired. To generalize to the cold-start events, CBPF naturally infers their content factor based on content text and optimal word topic factors learned from training data. Finally, we can get recommendation results by sorting the new events according to the predicted user response to them.

We should emphasize that start time information of events is also considered in this work, yet they seem to be not effective for the future event recommendation task, which will be discussed in the experiments.

Contributions. In summary, the main contributions of this paper lie in the following three aspects:

- To the best of our knowledge, we are the first to propose the problem of cold-start local event recommendation in EBSNs and explore the cold-start challenge in this problem.

- We propose the CBPF model which integrates event content, organizer, location and user social relation together. Moreover, it naturally enables to infer content topic factor of cold-start events in a unified model. Besides, an efficient coordinate ascent algorithm is adopted for this model.

- We crawled real event datasets from EBSNs and conducted comprehensive experiments. The results demonstrate the advantage of CBPF over several alternative methods.

In the rest of the paper, we first discuss the related work in Section 2. Then we formulate the problem we studied and give some preparations in Section 3. Section 4 detailedly introduce the proposed model and the learning algorithm. In Section 5, the experimental comparisons among between adopted methods and some analysis are provided. Finally, we draw a conclusion about this paper.

2. RELATED WORK

In this section, we expand the related work in two directions, i.e., either they study problems similar to the cold-start local event recommendation task or they propose methods relevant to it.

2.1 Event Recommendation in EBSNs

There are only a few research works studying on recommending events published online by organizers and held offline. Before proceeding, we should first emphasize that some researches have also mentioned event recommendation. Nevertheless, the concept of event in their work is different from what we study. For example, the activities in [29] mean broad human behaviors like shopping, tourism, and so on. The events in [12] and [18, 14] mean daily happened news and academic reports, respectively. As the carrier of events, event-based social networks are first analyzed in data mining field in [16]. In [28, 20], event-based group recommendation in EBSNs and their variants are formulated. However, as the group information is necessary and they do not consider textual content of events, the problems are still different from the event recommendation problem. Until recently, two works [21, 4] appear to address this problem. In [21], a standard matrix factorization approach which jointly models event, location, and social relation is proposed. Yet they ignore content and organizer information of events. There is no clear difference between their method and other methods applied to other recommendation problems such as location recommendation [13]. Du et al. further considers event content information [4]. However, their content modeling part is only based on topic distributions inferred from the standard topic model [2] and the learning process is separated from the final model. Moreover, their problem setting is binary classification for judging whether a user will attend an event and thus is a little different from the recommendation problem. Most of all, for the above two works, both of them associate a latent factor directly with each event and try to learn them from training data, which is not very demanding in real scenario. As we mentioned before, they ignore the life cycles of outdated events and newly published events which cause cold-start event recommendation. Hence, learning the latent factors directly from training data is impossible and their methods cannot be applied to our problem setting.

2.2 Related Methods for Recommendation

As the cold-start local event recommendation in EBSNs is a new problem and there is no standard method to solve this. We introduce several lines of research methods that are relevant to some aspects of the problem.

Textual content based methods. These methods focus on how to effectively model textual content information of users and items for recommendation. They are often utilized in cold-start recommendation. Word-based similarity methods [19] recommend items based on textual content similarity in word vector space. In [4], standard topic model [2]

is utilized to learn topics of users based on the content of their attended events, and then the similarity between topic factor of user and events is calculated, which is an important component of their method. In [26], CTR is proposed to combine standard topic model with matrix factorization for recommendation. Gopalan et al. [5] recently propose a Bayesian Poisson factorization (BPF) approach for modeling content in recommending articles.

Location based methods. There are many works adopting location information for recommendation in recent years. Some of them utilize distance information between locations [27, 28] and the idea is also adopted by [4] in event recommendation. Moreover, latent factor models such as matrix factorization [29] can model location information by associating each location with a latent factor. In [21], their partial method is to model location for event recommendation, which first clusters locations of events into regions and then assigns a latent factor to each region.

Multiple factor models. Latent factor models tend to integrate multiple factors to handle more complex relations in recommender systems in recent years. Tensor factorization [11] reconstructs the elements in a tensor by getting inner product of three factors, but not two factors in traditional factor models. In [1], factors from item side are enhanced by other factors. It addresses the pairwise interaction between multiple factors. Word latent factor is further incorporated into the multiple latent factor models in [3].

We adapt the above related methods to the cold-start local event recommendation problem. By comparing these alternatives with CBPF in the experiments, we demonstrate the superiority of our proposed model for the new problem.

3. PRELIMINARIES

We first provide some necessary definitions and formulate the cold-start local event recommendation problem. Then we describe two methods that are related to the proposed model.

3.1 Problem Formulation

Event based social network connects online and offline world through events. In the following, we formally define event based social network from the centric view of events.

Definition 1 (Event Based Social Network (EBSN)) *An event based social network is a heterogeneous graph* $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ *mainly containing six types of nodes* $\mathbb{V} = (E, U, O, C, L, T)$. *Among them, event set* E *is the most significant one which associates other nodes together. For an event* $e \in E$, *it should be published by an organizer* o_e ($o \in O$) *online. Meanwhile, the location* l_e ($l \in L$) *where it will be held and the timestamp* t_e *when it will start are also specified. Besides, event* e *should have a content text* c_e ($c \in C$) *to introduce itself. The textual document* c_e *consists of multiple words from a vocabulary* V *and* $C_{c_e v}$ *denotes the occurrence count of word* $v \in V$ *in the document* c_e. *For a user* $u \in U$, *if he is interested in the event* e, *he will register online for attending. Thus the event has a user attendance list* $U(e)$. *All the above relations are directly relevant to event node, which can be regarded as event-oriented relations. Besides, each user* $u \in U$ *may have a friend list* $F(u)$. *Hence, event-oriented relations and social relations form the edge set* $\mathbb{E}$.

Figure 1 provides a simple example to illustrate the diverse relations existed in event based social networks. As
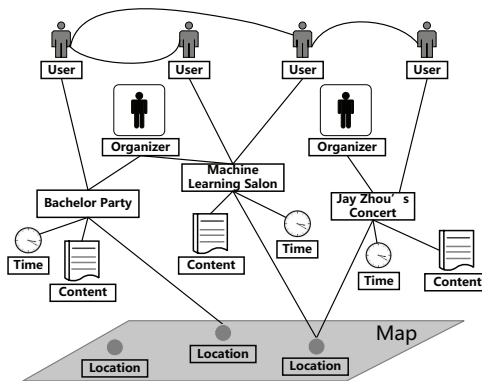

Figure 1: A toy example showing diverse relations in EBSNs.

we mentioned, each event connects to a unique organizer, textual content document, location address, and start time moment. However, for an organizer, he may organize several different events, such as the organizer who has organized the event of Bachelor Party and Machine Learning Salon shown in the figure. For a user, he may also attend more than one events and have many friends. It is also similar for a location at which several events are held. In summary, there are 1-to-1, 1-to-n, and n-to-n relations in the networks. Event recommendation plays a significant role for EBSNs. Usually, events to be recommended for each user should be held in the city where the user stays. Moreover, cold-start events are the main targets studied in this work. According to the above two requirements for events, we formally define the cold-start local event as follows,

Definition 2 (Cold-start Local Event) *Given a target city* $\mathbf{x}$, *a cold-start local event* $e \in E$ *not only should be held in this city, but also has been published online recently and thus has no registered users for attendance currently, i.e.,* $U(e) \subseteq \emptyset$.

The two requirements for cold-start local events are rational. Location requirement makes it possible for users to go to the locations where events held. And cold-start status requirement is realistic for many new events. Now based on the above definitions, we formally define the new problem studied in this work as below,

Problem (Cold-start Local Event Recommendation) *In an event based social network, given a target city* $\mathbf{x}$, *its historical local events list is denoted as* $E_{old}(\mathbf{x})$, *cold-start local event list is represented as* $E_{new}(\mathbf{x})$ *and user list is* $U(\mathbf{x})$. *For each user* $u \in U(\mathbf{x})$, *the goal is to rank every event* $e \in E_{new}(\mathbf{x})$ *according to the user response* $R_{u,e}$ *which is computed by a suitable predictive model. The model is constructed based on the known user response to historical events* $e \in E_{old}(\mathbf{x})$. *Finally, different top-n ranked events are recommended for each user.*

Naturally, how to compute the user response $R_{u,e}$ is the core of the problem. It is intuitive to utilize users' event attendance histories, social relations, and events' related information to construct an effective predictive model.

3.2 Bayesian Poisson Factorization

Bayesian Poisson factorization is proposed recently for implicit feedback and content based recommender system [6,

5]. Although Poisson factorization is already utilized for recommendations [17, 15], the key difference is that BPF combines Poisson factorization with Bayesian learning which can handle sparse data well and is more robust to the issue of overfitting. It shows promising results compared to traditional factorization models such as matrix factorization [22].

Specifically, the Poisson distribution for the rating $R_{u,v}$ of user u to item v is defined to be

$$\text{Poisson}(R_{u,v}; \theta_u^{\mathrm{T}}\theta_v) = (\theta_u^{\mathrm{T}}\theta_v)^{R_{u,v}} \exp(-\theta_u^{\mathrm{T}}\theta_v)/R_{u,v}! \quad (1)$$

where $\theta_u \in \mathbb{R}^{\mathrm{K}}$ denotes the latent factor of user u and $\theta_v \in \mathbb{R}^{\mathrm{K}}$ represents the latent factor of item v. $\theta_u^{\mathrm{T}}\theta_v$ is regarded as the shape parameter of Poisson factorization. Each component of the above two factors is assumed to drawn from a Gamma distribution defined as

$$\text{Gamma}(\theta_{\cdot,k}; \lambda_a, \lambda_b) = \frac{\lambda_b^{\lambda_a}}{\Gamma(\lambda_a)}\theta_{\cdot,k}^{\lambda_a-1}\exp\left(-\lambda_b\theta_{\cdot,k}\right) \quad (2)$$

where λ_a is the shape parameter of the Gamma distribution and λ_b is the rate parameter of the Gamma distribution.

The goal of Poisson factorization is to learn optimal θ_u and θ_v to reconstruct original training data. Under Bayesian learning framework, θ_u and θ_v should be marginalized and it is intractable to directly optimize them. To address the issue, Gibbs sampling [23] and variational Bayesian inference [6] are proposed. In this work, our proposed model CBPF builds on Bayesian Poisson factorization by taking it as its basic unit to model different types of data.

3.3 Collective Matrix Factorization

To jointly model multiple relational matrices together, collective matrix factorization is proposed [25]. The core idea of this model is to simultaneously reconstruct the several relation matrices through designed objective functions. All the matrices are associated with some shared elements.

For example, suppose there are two relation matrices M_1 and M_2. A_i denotes the factor of row i in matrix M_1 and B_j represents the factor of column j in the same matrix. It is similar for M_2 that B_n and C_m corresponds the factor of row n and column m, respectively. Hence B are the shared latent factors. To jointly model the two matrices, the following hybrid objective function is defined as,

$$\mathcal{L} = \alpha_1\mathcal{L}_1(M_1; A, B) + \alpha_2\mathcal{L}_2(M_2; B, C) \quad (3)$$

where α_1 and α_2 are relative weights to control the two sub-objectives. They are commonly tuned on validation datasets.

Unlike traditional settings of recommender system where only a user-item matrix needs to be modeled, multiple matrices exist in the cold-start local event recommendation problem, such as social relation matrix and event-word matrix. Therefore, we resort to the idea of collective matrix factorization. Particularly, we adopt Poisson factorization instead of matrix factorization as a basic unit and connect them through the idea of collective reconstruction.

4. PROPOSED CBPF MODEL

In this section, we first give an overview of the model. Then we describe the model in detail, including mathematical formulations. In what follows, the optimization approach based on coordinate ascent algorithm is provided. Finally, the way to infer the content topic factors of new events and predict user response to the events are introduced.
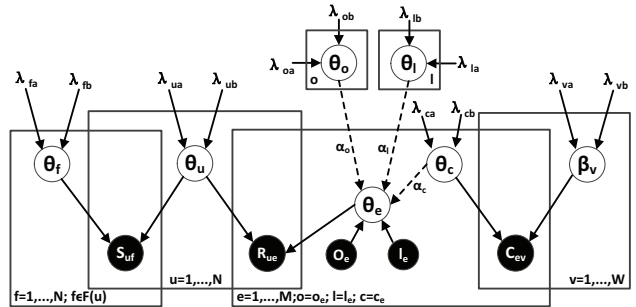

Figure 2: Graphical model of the CBPF model.

4.1 Model Overview

From a high-level perspective, CBPF combines the merits of Bayesian Poisson factorization and collective matrix factorization. As the graphical model of CBPF shown in Figure 2, the new model first utilizes BPF as its basic unit to model social relation, user response to events, and event content text separately. Then it connects each unit through the idea from CMF. The components in social relation and response matrices take binary value, but not in content word matrix, where the components take positive integer.

In latent factor models such as matrix factorization and Poisson factorization, each row and column of matrices associates with a K-dimensional latent factor. The main goal of latent factor models is to learn these latent factors in a model learning stage and predict the missing elements in matrix by inner product of their corresponding row and column factors in a prediction stage. However, in our problem setting, cold-start events which will be recommended to users do not occur in user-event and event-word matrices in the learning stage. As a result, we cannot directly associate a latent factor with each of them. The above problem is called out-of-matrix prediction in [26].

It is intuitive to utilize organizer, introduction textual content, location, and starting time information of events to overcome the cold-start issue. In CBPF model, event latent factor is replaced by the summarization of event organizer latent factor, event location latent factor, and event introduction content latent factor. We also try temporal latent factor in the experiments, nonetheless it is ineffective for this task, which will be discussed in the experimental part.

The number of organizers is much smaller than that of events and users. Besides, the total count of locations in a city is limited. Therefore, the cold-start degree of organizers and locations is minor. Based on this phenomenon, we assume each organizer and location occurs in training data at least once, and thus their latent factors can be learned when training model. Unlike the above two types of latent factors, the content factor of cold-start events should be inferred in the prediction stage. CBPF achieves this by modeling introduction textual content in both stages. More specifically, optimal word factors can first be obtained when learning the model. Then in the prediction stage, word factors are fixed and the knowledge contained in them are transferred to introduction content factors when modeling word occurrence count of content text.

Moreover, social relations are also considered in CBPF. In the binary social relation matrix we mentioned before, value one denotes users have social relation while zero not. CBPF aims at reconstructing the social matrix in model learning

stage. The goal of modeling social relations is to ensure latent factors of friends are similar.

4.2 Model Specification

We assume the dataset for a city is given. We denote the historical event set as E_{old} and cold-start event to be recommended as E_{new} according to the Definition 1, We shall use a user event pair (u, e) as an example for later introduction. A social friend f_u of the user, organizer o_e, introduction content c_e, and location l_e of the event e are also considered. We initiate the detailed specification of CBPF by the order of data generation process.

For latent factors shown in Figure 2, we assume they are drawn from Gamma distributions. This is because Gamma distributions are the conjugate priors for the shape parameters of Poisson distributions and it will facilitate Bayesian learning. Specifically, they are defined to be

$$
\begin{aligned}
\theta_{f_u,k} &\sim \text{Gamma}(\lambda_{fa}, \lambda_{fb}) & \theta_{u,k} &\sim \text{Gamma}(\lambda_{ua}, \lambda_{ub}) \\
\theta_{o_e,k} &\sim \text{Gamma}(\lambda_{oa}, \lambda_{ob}) & \theta_{l_e,k} &\sim \text{Gamma}(\lambda_{la}, \lambda_{lb}) \\
\theta_{c_e,k} &\sim \text{Gamma}(\lambda_{ca}, \lambda_{cb}) & \beta_{v,k} &\sim \text{Gamma}(\lambda_{va}, \lambda_{vb})
\end{aligned}
\tag{4}
$$

where $\theta_{f_u} \in \mathbb{R}^K$ is the social factor of user u, $\theta_u \in \mathbb{R}^K$ is the preference factor of user u. Hence, each user is associated with two types of factors. $\theta_o \in \mathbb{R}^K$ is the latent factor of organizer o, and it is similar for $\theta_l \in \mathbb{R}^K$ of location l, $\theta_c \in \mathbb{R}^K$ of event introduction textual content c, and $\beta_v \in \mathbb{R}^K$ of word v. The expression of the Gamma distribution is shown in Equation 2.

To represent latent factor θ_e of event e, we utilize organizer factor, textual content factor, and location factor. Specifically, we incorporate relative weights between three factors to adjust their final contributions to the preference response $R_{u,e}$. Formally, the formula of computing θ_e is defined as below,

$$
\theta_e = \alpha_o \theta_{o_e} + \alpha_c \theta_{c_e} + \alpha_l \theta_{l_e}
\tag{5}
$$

where α_o, α_c, and α_l are relative weights and can be tuned based on recommendation performance on validation datasets. One simple way to tune them is to first determine the most significant factor and set its relative weight to be one. Then we can constrain the other two relative weights to be within the range of zero to one. Finally a grid search approach with a fixed step size can be applied to determine them. The detailed settings of the these three relative weights can be seen in the experimental part.

Suppose $S_{uu'}$ denotes the binary value of social relation between user u and u'. We specify the values of $S_{uu'}$, $R_u e$, and $C_{c_e v}$ to be generated from Poisson distributions. Particularly, they can be expressed as

$$
\begin{aligned}
S_{uu'} &\sim \text{Poisson}(\theta_u^{\mathrm{T}} \theta_{f_{u'}}), & R_{ue} &\sim \text{Poisson}(\theta_u^{\mathrm{T}} \theta_e) \\
C_{c_e v} &\sim \text{Poisson}(\theta_{c_e}^{\mathrm{T}} \theta_v)
\end{aligned}
\tag{6}
$$

Based on the above description, we can conclude the generative story of CBPF as following,

1. For each user u,
   (a) Draw latent factor $\theta_u \sim \text{Gamma}(\lambda_{ua}, \lambda_{ub})$.
   (b) Draw social factor $\theta_{f_u} \sim \text{Gamma}(\lambda_{fa}, \lambda_{fb})$.
2. For each organizer o, draw latent factor
   $$\theta_o \sim \text{Gamma}(\lambda_{oa}, \lambda_{ob}).$$
3. For each location l, draw latent factor
   $$\theta_l \sim \text{Gamma}(\lambda_{la}, \lambda_{lb}).$$

4. For each word v, draw topic factor
   $$\beta_v \sim \text{Gamma}(\lambda_{va}, \lambda_{vb}).$$
5. For each event e,
   (a) Draw content topic factor, $\theta_{c_e} \sim \text{Gamma}(\lambda_{ca}, \lambda_{cb})$.
   (b) For each word v in the introduction content, draw word occurrence count, $C_{c_e v} \sim \text{Poisson}(\theta_{c_e}^{\mathrm{T}} \theta_v)$.
6. For each user-user pair (u, u'), draw the binary social relation, $S_{uu'} \sim \text{Poisson}(\theta_u^{\mathrm{T}} \theta_{f_{u'}})$.
7. For each user-event pair (u, e), draw the preference response, $R_{ue} \sim \text{Poisson}(\theta_u^{\mathrm{T}} \theta_e)$, where θ_e is calculated through Equation 5.

For convenience, we denote all the Gamma priors of latent factors with an integrated expression $p(\Theta, \beta; \lambda_{\cdot,a}, \lambda_{\cdot,b})$ where $\Theta = \{\theta_u, \theta_o, \theta_{c_e}, \theta_l, \theta_{f_{u'}}\}$, $\forall u \in U, \forall o \in O, \forall e \in E, \forall l \in L, \forall u' \in U$. The joint probability of generating all visible data is defined to be,

$$
\begin{aligned}
p(R, S, C) &= p(\Theta, \beta; \lambda_{\cdot,a}, \lambda_{\cdot,b}) \\
&\prod_{e=1}^{E} \prod_{v=1}^{V} \int \int \text{Poisson}(C_{c_e v}; \theta_{c_e}^{\mathrm{T}} \beta_v) d\theta_{c_e} d\beta_v \\
&\prod_{u=1}^{U} \prod_{u' \neq u}^{U} \int \int \text{Poisson}(S_{uu'}; \theta_u^{\mathrm{T}} \theta_{f_{u'}}) d\theta_u d\theta_{f_{u'}} \\
&\prod_{u=1}^{U} \prod_{e=1}^{E} \int \int \text{Poisson}(R_{ue}; \theta_u^{\mathrm{T}} \theta_e) d\theta_u d\theta_e
\end{aligned}
\tag{7}
$$

4.3 Optimization Approach

The core goal in the model learning stage is to get the optimal latent factors, i.e., $\{\Theta, \beta\}$, for predicting user response to events. As the events to be recommended are cold-start, their content factor $\theta_{c_e}(e \in E_{new})$ cannot be obtained in the learning stage, we leave the details of inferring θ_{c_e} in the prediction stage. Given a set of training data, the way to achieve the goal under Bayesian learning is to compute the posterior distribution $p(\Theta, \beta | R, S, C)$. However, it is intractable to directly compute the posterior because the normalization term shown in Equation 7 contains the coupling integration variables. We resort to variational Bayesian inference [10, 2, 5] to address this issue.

The general idea of variational Bayesian inference is to derive a lower bound of the normalization term in this work, i.e, $p(R, S, C; \lambda_{\cdot,a}, \lambda_{\cdot,b})$. and then optimizes the lower bound through standard learning algorithms. The lower bound is usually obtained by applying Jensen's inequality though a new designed variational distribution $q(\Theta, \beta)$,

$$
\begin{aligned}
\log p(R, S, C) &= \log \int p(R, S, C, \Theta, \beta) d\Theta d\beta \\
&= \log \int p(R, S, C, \Theta, \beta) \frac{q(\Theta, \beta)}{q(\Theta, \beta)} d\Theta d\beta \\
&\geq q(\Theta, \beta) \log \int \frac{p(R, S, C, \Theta, \beta)}{q(\Theta, \beta)} d\Theta d\beta \\
&= \mathcal{L}(q)
\end{aligned}
\tag{8}
$$

To implement variational Bayesian inference for CBPF, we should first incorporate several types of auxiliary latent variables to facilitate inference like [5]. Specifically, for user-user pair (u, u'), we add K latent variables $s_{uu',k} \sim \text{Poisson}(\theta_{u,k} \theta_{f_{u'},k})$ and the characteristics of Poisson distributions ensure they are integers whose sum is equal to $S_{uu'}$.

Similar operation is repeated for $z_{c_e v,k} \sim \text{Poisson}(\theta_{c_e,k}\beta_{v,k})$, $C_{ev} = \sum_k z_{c_e v,k}$. The operation for user-event pair (u,e) is different as θ_e consists of three types of latent variables. Following [5], we construct a 3K latent space by adding K latent variables $r_{uo_e,k} \sim \text{Poisson}(\alpha_o \theta_{u,k}\theta_{o_e,k})$, $r_{uc_e,k} \sim \text{Poisson}(\alpha_c \theta_{u,k}\theta_{c_e,k})$, and $r_{ul_e,k} \sim \text{Poisson}(\alpha_l \theta_{u,k}\theta_{l_e,k})$, respectively. Their sum should satisfies the requirement $R_{ue} = \sum_k r_{uo_e,k} + r_{uc_e,k} + r_{ul_e,k}$. Thus for R_{ue} which equals 0, $r_{uo_e,k}$, $r_{uc_e,k}$, and $r_{ul_e,k}(\forall k \in K)$ are all constrained to be 0. And it is the same for $s_{uu',k}$ and $z_{c_e v,k}$. This feature helps decreasing the complexity of variational parameter space and complexity of learning algorithm. After adding these new latent variables, the variational distribution becomes $q(\Theta, \beta, Z)$ where Z denotes all the added latent variables.

Before we continue, we first get the complete conditional distribution [7] for each latent variable, which will be used for later parameter updating. We divide the latent variables into two categories. The first category includes Θ and β whose priors are Gamma distributions. We take $\beta_{v,k}$ as an example and fix other latent variables. After extracting the relevant terms of $\beta_{v,k}$ from Equation 7, we can derive the complete condition distribution of $\beta_{v,k}$ as below,

$$p(\beta_{v,k}|\mathbf{z}, \theta_c, \lambda_{va}, \lambda_{vb})$$

$$\propto \beta_{v,k}^{\lambda_{va}-1} \exp(-\lambda_{vb}\beta_{v,k}) \prod_e (\theta_{c_e,k}\beta_{v,k})^{z_{c_e v,k}} \exp(-\theta_{c_e,k}\beta_{v,k})$$

$$\propto \beta_{v,k}^{\lambda_{va}+\sum_e z_{c_e v,k}-1} \exp\left(-(\lambda_{vb} + \sum_e \theta_{c_e,k})\beta_{v,k}\right)$$

$$= \text{Gamma}(\lambda_{va} + \sum_e z_{c_e v,k}, \lambda_{vb} + \sum_e \theta_{c_e,k})$$

$$(9)$$

It is similar to derive the conditionals for other latent variables belonging to the first category. The second category covers all the auxiliary latent variables whose priors are Poisson distributions. It is a little more complex to derive them than the variables in first category. We utilize the conclusion from [9, 6] that given the sum of a set of latent variables drawn from Poisson distributions, the conditional distribution of the variables is a multinomial whose parameters are the normalized values of their priors. For example, the conditional distribution of $z_{c_e v}$ is

$$p(z_{ev}|C, \theta_c, \beta) = \text{Mult}(C_{c_e v}; \frac{\theta_{c_e} \cdot \beta_v}{\theta_{c_e}^{\mathsf{T}}\beta_v}) \qquad (10)$$

where $\cdot$ denotes the element-wise product operation. The integrated list of conditional distributions for all latent variables is shown in Table 1, in which the parameter form of Multinomial distribution corresponds to its formula in an exponential family.

To maximize the lower bound $\mathcal{L}(q)$, we first define $q(\Theta, \beta, Z)$ with a mean-field variational form [10],

$$q(\Theta, \beta, Z) = \prod_{u,k} p(\theta_{u,k}|\tilde{\theta}_{u,k}^{shp}, \tilde{\theta}_{u,k}^{rte})p(\theta_{f_u,k}|\tilde{\theta}_{f_u,k}^{shp}, \tilde{\theta}_{f_u,k}^{rte})$$

$$\prod_{v,k} p(\beta_{v,k}|\tilde{\beta}_{v,k}^{shp}, \tilde{\beta}_{v,k}^{rte}) \prod_{o,k} p(\theta_{o,k}|\tilde{\theta}_{o,k}^{shp}, \tilde{\theta}_{o,k}^{rte}) \prod_{l,k} p(\theta_{l,k}|\tilde{\theta}_{l,k}^{shp}, \tilde{\theta}_{l,k}^{rte})$$

$$\prod_{e,k} p(\theta_{c_e,k}|\tilde{\theta}_{c_e,k}^{shp}, \tilde{\theta}_{c_e,k}^{rte}) \prod_{e,v,k} p(z_{c_e v,k}|\delta_{c_e v,k}) \prod_{u,u',k} p(s_{uu',k}|\psi_{uu',k})$$

$$\prod_{u,e,k} p(r_{uo_e,k}, r_{uc_e,k}, r_{ul_e,k}|\kappa_{ue,k}^o, \kappa_{ue,k}^c, \kappa_{ue,k}^l)$$

$$(11)$$

Table 1: Conditional distribution of latent variables.

Variable	Type	Conditional Parameter
$\beta_{v,k}$	Gamma	$\lambda_{va} + \sum_e z_{c_e v,k}, \quad \lambda_{vb} + \sum_e \theta_{c_e,k}$
$\theta_{f_{u'},k}$	Gamma	$\lambda_{fa} + \sum_{u \neq u'} s_{uu',k}, \quad \lambda_{fb} + \sum_{u \neq u'} \theta_{u,k}$
$\theta_{u,k}$	Gamma	$\lambda_{ua} + \sum_{u' \neq u} s_{uu',k} + \sum_e (r_{uo_e,k} + r_{uc_e,k} + r_{ul_e,k}),$ $\lambda_{ub} + \sum_{u' \neq u} \theta_{f_{u'},k} + \sum_e (\alpha_o \theta_{uo_e,k} + \alpha_c \theta_{uc_e,k} + \alpha_l \theta_{ul_e,k})$
$\theta_{o,k}$	Gamma	$\lambda_{oa} + \sum_u \sum_e \mathbb{I}(o=o_e)r_{uo_e,k},$ $\lambda_{ob} + \sum_u \sum_e \mathbb{I}(o=o_e)\theta_{u,k}$
$\theta_{l,k}$	Gamma	$\lambda_{la} + \sum_u \sum_e \mathbb{I}(l=l_e)r_{ul_e,k},$ $\lambda_{lb} + \sum_u \sum_e \mathbb{I}(l=l_e)\theta_{u,k}$
$\theta_{c_e,k}$	Gamma	$\lambda_{ca} + \sum_u r_{uc_e,k} + \sum_v c_{c_e v,k},$ $\lambda_{cb} + \sum_u \theta_{u,k} + \sum_v \beta_{v,k}$
$z_{c_e v,k}$	Mult	$\log \theta_{c_e,k} + \log \beta_{v,k}$
$s_{uu',k}$	Mult	$\log \theta_{u,k} + \log \theta_{f_{u'},k}$
$r_{ue,k}$	Mult	$\begin{cases} \log \alpha_o + \log \theta_{u,k} + \log \theta_{o_e,k}, & \text{if } k \leq K \\ \log \alpha_c + \log \theta_{u,k} + \log \theta_{c_e,k}, & \text{if } K < k \leq 2K \\ \log \alpha_l + \log \theta_{u,k} + \log \theta_{l_e,k}, & \text{if } 2K < k \leq 3K \end{cases}$

where each p denotes the corresponding type of its variables in Table 1. The parameters with superscript shp represent the shape parameters of Gamma distributions while the parameters with superscript rte mean the rate parameters. $\delta_{c_e v,k}$ and $\psi_{uu',k}$ are the multinomial parameters for $z_{c_e v,k}$ and $s_{uu',k}$, respectively. $\kappa_{ue,k}^o$ is one part of multinomial parameter for $r_{ue,k}$ which belongs to the dimension of 1 to K. Analogously, $\kappa_{ue,k}^c$ belongs to $K+1$ to $2K$ and $\kappa_{ue,k}^l$ to $2K+1$ to $3K$. Overall, each latent variable is independent on the others. If the optimal variational parameters can be obtained, then approximate posterior distribution, i.e., $p(R, S, C; \lambda_{\cdot,a}, \lambda_{\cdot,b})$, can be calculated.

After substituting the variational distribution in Equation 8 with Equation 11, we optimize the lower bound to get optimal variational parameters through a coordinate ascent algorithm adopted in [7, 6, 5]. The central idea of the algorithm is to optimize one variable each time while fixing all other variables. The conclusion from [7] shows that if the complete conditional distribution of a latent variable is in an exponential family and its corresponding variational distribution has the same form, then its variational parameters have a closed-form solution using coordinate ascent algorithm. More specifically, the variational parameter equals the expectation of the conditional parameter in its corresponding posterior distribution under the complete variational distribution $q(\Theta, \beta, Z)$. Luckily, Bayesian Poisson factorization satisfies the requirements [5] and it is also suitable for the proposed CBPF.

We choose the variational parameters, i.e., $\tilde{\theta}_{c_e,k}$ and $\kappa_{ue,k}$, as examples to derive their closed-form solution in each update. The solutions for other variational parameters can be educed similarly. For $\tilde{\theta}_{c_e,k}^{shp}$ and $\tilde{\theta}_{c_e,k}^{rte}$, based on the conclusion above, the closed-form solutions can be represented as,

$$\tilde{\theta}_{c_e,k}^{shp} = E_q\left[\lambda_{ca} + \sum_u r_{uc_e,k} + \sum_v c_{c_e v,k}\right]$$

$$\tilde{\theta}_{c_e,k}^{rte} = E_q\left[\lambda_{cb} + \sum_u \theta_{u,k} + \sum_v \beta_{v,k}\right]$$

$$(12)$$

where $E_q[x]$ denotes the expectation of variable x under the probability distribution q. After solving the expectation terms, we can get the following update expressions,

$$\tilde{\theta}_{c_e,k}^{shp} = \lambda_{ca} + \sum_u R_{ue}\kappa_{ue,k}^c + \sum_v C_{c_e v}\delta_{c_e v,k}$$

$$\tilde{\theta}_{c_e,k}^{rte} = \lambda_{cb} + \sum_u \frac{\tilde{\theta}_{u,k}^{shp}}{\tilde{\theta}_{u,k}^{rte}} + \sum_v \frac{\tilde{\beta}_{v,k}^{shp}}{\tilde{\beta}_{v,k}^{rte}}$$

$$(13)$$

For $\kappa_{ue,k}$, it has a similar expectation like Equation 12. However, its update expression is more complex since the expectation contains logarithms and $\kappa_{ue,k}$ lies in a $3K$ probability space. Formally, the derived update formula is

$$
\begin{cases}
\kappa_{ue,k}^o \propto \exp\{\Psi(\alpha_o) + \Psi(\tilde{\theta}_{u,k}^{shp}) - \log(\tilde{\theta}_{u,k}^{rte}) \\
\qquad + \Psi(\tilde{\theta}_{oe,k}^{shp}) - \log(\tilde{\theta}_{oe,k}^{rte})\} & \text{if } 0 < k \le K \\
\kappa_{ue,k}^c \propto \exp\{\Psi(\alpha_c) + \Psi(\tilde{\theta}_{u,k}^{shp}) - \log(\tilde{\theta}_{u,k}^{rte}) \\
\qquad + \Psi(\tilde{\theta}_{ce,k}^{shp}) - \log(\tilde{\theta}_{ce,k}^{rte})\} & \text{if } K < k \le 2K \\
\kappa_{ue,k}^l \propto \exp\{\Psi(\alpha_l) + \Psi(\tilde{\theta}_{u,k}^{shp}) - \log(\tilde{\theta}_{u,k}^{rte}) \\
\qquad + \Psi(\tilde{\theta}_{le,k}^{shp}) - \log(\tilde{\theta}_{le,k}^{rte})\} & \text{if } 2K < k \le 3K
\end{cases}
\tag{14}
$$

where $\Psi(\cdot)$ denotes the Digamma function. We should emphasize that the above parameters should be normalized together to ensure their sum to be one. The above updates utilize the conclusion about the expectation of a logarithm variable x with a Gamma prior, i.e., $E_q[\log x] = \Psi(x) - \log x$. Relative weights such as α_o can be incorporated here by assuming they are drawn from Gamma distributions with the shape parameters being α_o, α_c, or α_l, and the rate parameters just being value one.

Based on the above update formulas and their variants for other variational parameters, we can update each of them one time in a circulation and repeat this process iteratively until the values of the parameters converge.

4.4 Prediction

The prediction stage of CBPF for the cold-start local event recommendation task mainly consists of two steps. The first step is to infer the variational parameters of content topic factors for new events and the second step is to predict user response to the events.

First, we only update $\tilde{\theta}_{ce,k}$ and $\delta_{c_ev,k}(e \in E_{new})$ and keep other variational parameters fixed. While the update formula of $\delta_{c_ev,k}$ is similar to that in the model learning stage, the way to compute $\tilde{\theta}_{ce,k}$ is different and shown below,

$$
\tilde{\theta}_{ce,k}^{shp} = \lambda_{ca} + \sum_v C_{c_ev}\delta_{c_ev,k}
$$
$$
\tilde{\theta}_{ce,k}^{rte} = \lambda_{cb} + \sum_v \frac{\tilde{\beta}_{v,k}^{shp}}{\tilde{\beta}_{v,k}^{rte}}
\tag{15}
$$

where user related terms such as $\kappa_{ue,k}^c$ vanish. It is intuitive since the real responses of users to the new events are unknown. $\tilde{\theta}_{ce,k}$ and $\delta_{c_ev,k}$ are updated iteratively and only several iterations are necessary to achieve good results.

Then the preference response of user u to the new event e can be predicted based on the following formula,

$$
\tilde{R}_{u,e} = \sum_k \frac{\tilde{\theta}_{u,k}^{shp}}{\tilde{\theta}_{u,k}^{rte}}\Big(\alpha_o \frac{\tilde{\theta}_{oe,k}^{shp}}{\tilde{\theta}_{oe,k}^{rte}} + \alpha_c \frac{\tilde{\theta}_{ce,k}^{shp}}{\tilde{\theta}_{ce,k}^{rte}} + \alpha_l \frac{\tilde{\theta}_{le,k}^{shp}}{\tilde{\theta}_{le,k}^{rte}}\Big)
\tag{16}
$$

Finally, the events are ranked according to the response scores of each user and then personalized top-n event recommendations are delivered to users.

5. EXPERIMENTS

In this section, our goal is to evaluate the effectiveness of the proposed model. To achieve this, we first describe the real datasets we used in the experiments. Then we introduce the evaluation metrics, adopted comparison methods, and parameter settings of the proposed model. Finally, we compare the results of CBPF and the comparison methods.

5.1 Datasets

5.1.1 Data Introduction

Because no benchmark datasets are available for evaluating performance on the event recommendation task, we collected real datasets for events and users by crawling from Douban Event in 2012. For each event, we get its organizer, content introduction, geographical address (including location name, longitude and latitude), start time information, and a list of registered users for attending. For each user, we acquire his event attendance list and social friend list.

Table 2: Introduction of experimental datasets.

Data	User	Event	Organizer	Location
Beijing	64113	12955	509	3212
Shanghai	36440	6753	328	1990

5.1.2 Data Preprocessing

To simulate real scenarios, we first partition all events according to their corresponding cities. We then choose Beijing[3] and Shanghai[4], the two largest cities in China, to create two local event datasets. As home addresses of users are private, we choose users for both cities just based on whether they have attended the events in them. We further remove users who attended less than five events to filter noisy data. To test the proposed model, we divide both cities' events into training and prediction set based on chronological order with a common ratio of 7:3. The user register list of events in the prediction sets are unknown when learning models, and thus we can regard events in prediction set as cold-start events. We further partition the prediction set into validation and test set with a ratio of 1:2. For event content information, we first conducted Chinese word segmentation and removed stop words from it. Then we construct a word vocabulary by filtering some noisy words which occur very few times in the datasets. Finally, the basic statistics of the datasets we used are shown in Table 2.

5.2 Evaluation Settings

5.2.1 Evaluation Metrics

We adopt Precision and Normalized Discounted Cumulative Gain at position n (P@n and NDCG@n), both of which are widely used in the top-n recommendation task. In our task, P@n measures the ratio of the recommended events that are really attended by users. NDCG@n further assumes the events appearing earlier in a recommendation list are more important and assigns more weights to the ground-truth events that are ranked higher.

In real scenario of recommender systems, it is desirable that the first event a user is willing to attend should appear as early as possible in a top-n recommendation list. To measure this point, we employ Mean Reciprocal Rank (MRR) which measures the reciprocal of the first occurrence position of ground truth event for each user.

All the three metrics are first calculated on each user's recommendation list separately and then taken an average among all users. If the values of the three metrics are larger, the recommendation performance is better.

---

[3]http://en.wikipedia.org/wiki/Beijing
[4]http://en.wikipedia.org/wiki/Shanghai

5.2.2 Comparison Methods

As there is no standard method to solve this new problem, we adopt several alternative location-based, content-based, and multiple factor methods. For the multiple factor models we adopted, we also incorporate relative weights into them as we do for CBPF to ensure fair comparisons. The hyper-parameters of all the adopted methods, such as regularization parameter and relative weights, are tuned according to performances on validation datasets. For latent factor models, we set the dimensions to be 50 uniformly, which is large enough for comparing different results.

L-Dis. We implement distance-based method as [27] by learning an exponential decay function about distance between users' visited locations and target locations.

L-HeSig. Although HeSig proposed in [21] cannot be directly applied to our task, its partial method which divides events' locations into regions and learn user preferences to the regions can be compared here. The number of clusters is set to be 100 experimentally.

L-BPF. As the basic Bayesian Poisson factorization [6] is the basic unit of our method, we adopt it here for event recommendation by utilizing events' location information.

Word-based Similarity (WBS). WBS constructs word vectors for new events based on their introduction content and for users based on content of their attended events. The events whose word vectors are more similar to users' will be recommended.

Topic-based Similarity (TBS). Unlike word space adopted in WBS, TBS utilizes topic model to get low-dimensional topic vectors for users and new events. It is the partial method adopted in [4] for using content information.

C-BPF. Bayesian Poisson factorization [5] is also utilized in content-based recommendation. Hence we should compare CBPF with it.

O-BPF. It utilizes organizer information with Bayesian Poisson factorization. The modeling process is similar to L-BPF.

CTR. CTR [26] is a standard content-based recommendation algorithm which combines topic model with matrix factorization for recommending content-based items.

Tensor Factorization (TF). 3-way tensor factorization is employed in recommender systems related to three different types of factors [11]. Here we specify them as user, organizer, and location factor.

MLFM. Multiple latent factor model is adopted in many works such as [1]. It addresses the pairwise interactions between user factor and other factors.

WMLFM. Word-enhanced multiple latent factor model is similar to MLFM by additionally adding word-based latent factors. It is similar to the models proposed in [3, 8]. However, it is not very efficient as latent factors of all words in an event should be summarized in every computation.

5.2.3 Settings of CBPF

We follow the hyper-parameter settings in [5]. Specifically, the Gamma prior parameters of all the latent factors are fixed to be 0.3. We initialize the variational parameters of θ_c and β_v using the topic factors learned from the standard topic model [2]. The variational parameters of θ_u, θ_o, θ_l, and θ_f are set to be their Gamma priors with a small random noise. By setting α_o to be 1, α_c and α_l are tuned to be 0.4 and 0.1, respectively.

5.3 Effectiveness Study

5.3.1 Performance Comparison

In this section, we analyze the results of all the adopted comparison methods and CBPF on metrics of NDCG@3, Precison@3, NDCG@5, Precision@5, and MRR. The results of them on both datasets are shown in Figure 3.

We first compare the performance of the location-based methods. From the Figure 3(a) and 3(b) we can see L-HeSig performs a little worse than the basic method, i.e., L-Dis. Thus it is not very effective in this setting. L-BPF outperforms the above two methods significantly. Particularly, by comparing L-BPF with L-HeSig, we can conclude that a more fine-grained location factor is more suitable than a region based latent factor in this task. The results provide the intuition of modeling location information through a latent factor model such as BPF.

We then analyze the results of content-based methods, i.e., WBS, TBS, C-BPF, and CTR. As expected, WBS performs worst among the four methods due to the fact that the word vectors of users and events are very long (vocabulary size) and relatively sparse, which may lead to inaccurate similarities. TBS overcomes the data sparsity issue by utilizing topic model to learn low-dimensional topic vectors and gain notable improvements over WBS on both datasets. Hence, topic-based similarity is better than bag-of-words based similarity for event recommendation. It is a surprise that CTR performs not well in this task, with only better results than WBS. C-BPF performs best among the related algorithms. [5] also shows its improvement over CTR. In summary, utilizing BPF as a basic unit in CBPF to model content information of events is promising.

Now we study the results of O-BPF. By comparing it with L-BPF and C-BPF, we find that O-BPF behaves better than the two methods in Figure 3(e) and 3(f). It reveals the organizer information is more important to the cold-start local event recommendation task. We also see that O-BPF even outperforms MLFM which additionally incorporates location factor into the model slightly. This is mainly explained by the reason that BPF is more suitable for modeling implicit user feedback than matrix factorization methods [6] and user response to events is one kind of such implicit feedback.

Finally, we make comparisons between CBPF and the other methods, especially the adopted multi-factor models. TF does not behave very well due to the sparsity of user-organizer-location cube. WMLFM performs the third best because it integrates all related factors. CBPF-S is the sub-method of CBPF by removing the social factor from CBPF. Although CBPF is slightly better than CBPF-S, it can still indicate the rationality of incorporating social relations into the model. CBPF achieves the best results consistently in both datasets, which demonstrates it is effective and better than the other alternative methods for the new cold-start local event recommendation problem.

5.3.2 Factor Contribution to CBPF

Organizer, content, and location factor are three main factors for events in this task. Although the results of L-BPF, C-BPF, and O-BPF shown in Figure 3(e) and 3(f) can indicate their effectiveness for the task alone, the contribution of each factor to CBPF should also be explored. This is because combining multiple latent factors to form a unified
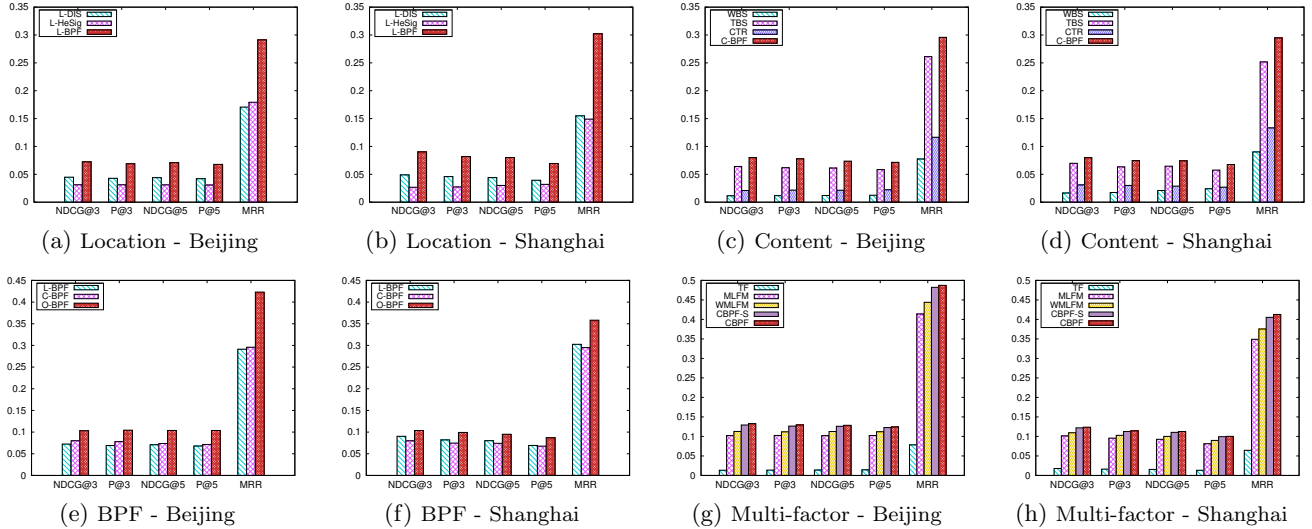
(a) Location - Beijing (b) Location - Shanghai (c) Content - Beijing (d) Content - Shanghai

(e) BPF - Beijing (f) BPF - Shanghai (g) Multi-factor - Beijing (h) Multi-factor - Shanghai

Figure 3: Comparisons of different types of methods on NDCG@n, Precision@n, and MRR metrics.

model does not mean the results of the new model is the performance summarization of each factor.

We adopt the strategy of removing one factor from CBPF each time to test the contribution of the removed factor to CBPF. Specifically, we test three sub-methods, i.e., CBPF-O, CBPF-C, and CBPF-L. The results of them are displayed in Figure 4. We find CBPF-O performs clearly worse than the other methods, which again indicates the importance of organizer information to the task. The reason may be attributed to the characteristics of offline meeting in events. Users are more cautious to make decisions and inclined to attend the events held by the organizers their trust. CBPF-L achieves better results than CBPF-C, which reveals location information makes a smaller contribution to CBPF than content information. This is because organizers get used to holding events in several fixed locations, which leads to the decrease of information gain when location factors are added to the integrated model.

Lastly, we provide some more results of modeling social relations collectively as a complementary to the comparison of CBPF and CBPF-S shown in Figure 3(g) and 3(h). We construct SC-BPF which is an extension of C-BPF by incorporating social relations and observe the improvements of SC-BPF over C-BPF presented in Figure 4(c) and 4(d). This also indicates that considering social relations is effective for the task, although the improvements are minor.

5.3.3 Time Factor Influence on Performance

We try to utilize the start time information of events for recommendation. As time is continuous, we should first discretize the time space. Specifically, we create a 48-dimensional time vector. Each dimension corresponds to a one-hour period in a weekday or weekend. Bayesian Poisson factorization is applied to model user preference to the time periods. However, the performances of event recommendation are relatively low on both datasets, with 0.0109 on NDCG@3 of Beijing dataset and 0.0113 on NDCG@3 of Shanghai dataset. It reveals that the start time information of events is not very effective for this task. One main reason is that the users' online register behaviors for attending events mainly reflect their interests. They may not

consider whether they are available at the start time of a specific event. Actually, it is not easy for users to determine it so long before the events will be held.

5.3.4 Complexity Analysis

As [6] indicates, the computational cost mainly depends on non-zero elements in matrices such as user-friend, user-event, and event-word count matrices in this work. Here we provide an illustration for our method. For coordinate ascent algorithm, its computational cost is mainly determined by the space complexity of parameters to be updated. We take κ_{ue} shown in Equation 14 as an example to illustrate the variational parameters of added latent factors as they dominate the space complexity of all latent factors. Suppose the number of non-zero elements in user-event interaction matrix is $|A_{ue}|$ which satisfies $|A_{ue}| \ll |U||E|$. For R_{ue} which is equal to 0, its corresponding enhanced latent variables $r_{uo_e,k}, r_{uc_e,k}$, and $r_{ul_e,k}(\forall k \in K)$ are constrained to be 0 as we discussed before. Thus there is no need to learn $\kappa_{ue,k}$ for these latent variables as they are already known. As a result, the space complexity of $\kappa_{ue,k}$ to be learned is $|A_{ue}||K|$ instead of $|U||E||K|$. It is the same for ψ_{uf} and $\delta_{c_e v}$ that their parameter space complexities are based on the number of non-zero elements in user-user social matrix and event content-word matrix as well. For the other variational parameters shown in Equation 11, their complexities are much lower than the above variational parameters and some of them are also based on the number of non-zero elements, such as $\tilde{\theta}_{c_e,k}^{shp}$ shown in Equation 13. Overall, the learning algorithm is efficient.

Table 3: Learning time cost in one iteration.

Data	WMLFM	CBPF
Beijing	446.6s	11.7s
Shanghai	192.8s	4.8s

To quantifiably verify the efficiency of CBPF, the learning time cost comparison with WMLFM is provided in Table 3. As we can see, CBPF costs much less time than WMLFM, which confirms to the expectation.
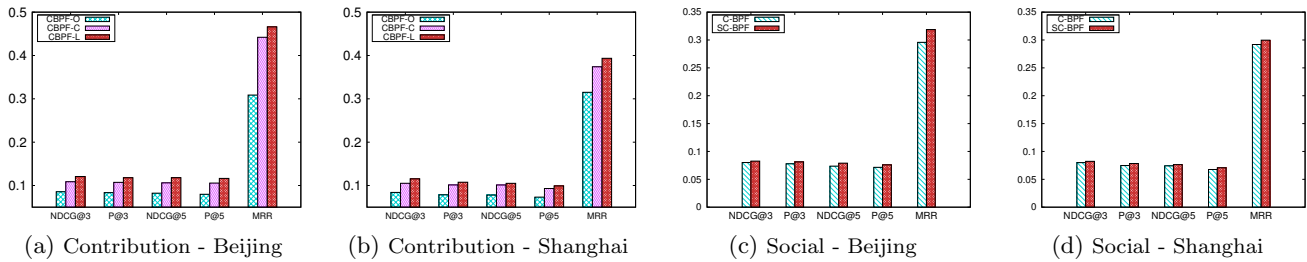
Figure 4: Results of sub-methods of CBPF and effectiveness of incorporating social relation.

6. CONCLUSION

In this paper, we have studied a new problem on cold-start local event recommendation in event-based social social networks. We propose a new model called collective Bayesian Poisson factorization to handle this problem. The new model collectively integrates user response, social relation, and event content information through Bayesian Poisson factorization. To address the cold-start issue, the model further utilizes events' organizer, location, and textual content information to learn representations for the cold-start events. An efficient coordinate ascent algorithm is adopted to learn the optimal parameters fo the model. The experimental results on real event datasets have shown that our model is effective and outperforms several alternative methods.

Acknowledgments

7. REFERENCES

[1] N. Aizenberg, Y. Koren, and O. Somekh. Build your own music recommender by modeling internet radio streams. In *WWW, Lyon, France, April 16-20*, pages 1–10, 2012.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu. Collaborative personalized tweet recommendation. In *SIGIR, Portland, OR, USA, August 12-16*, pages 661–670, 2012.

[4] R. Du, Z. Yu, T. Mei, Z. Wang, Z. Wang, and B. Guo. Predicting activity attendance in event-based social networks: content, context and social influence. In *Ubicomp, Seattle, WA, USA, September 13-17*, pages 425–434, 2014.

[5] P. Gopalan, L. Charlin, and D. M. Blei. Content-based recommendations with poisson factorization. In *NIPS, December 8-13, Montreal, Quebec, Canada*, pages 3176–3184, 2014.

[6] P. Gopalan, J. M. Hofman, and D. M. Blei. Scalable recommendation with poisson factorization. *CoRR*, vol. abs/1311.1704, 2013.

[7] M. D. Hoffman, D. M. Blei, C. Wang, and J. W. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[8] L. Hu, A. Sun, and Y. Liu. Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In *SIGIR, Gold Coast, QLD, Australia, July 06-11*, pages 345–354, 2014.

[9] N. L. Johnson, Z. Kotz, and A. W. Kemp. *Univariate Discrete Distributions, 2nd Edition*. Wiley & Sons, New York, 1993.

[10] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

[11] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *RecSys, Barcelona, Spain, September 26-30*, pages 79–86, 2010.

[12] H. Khrouf and R. Troncy. Hybrid event recommendation using linked data and user diversity. In *RecSys, Hong Kong, China, October 12-16*, pages 185–192, 2013.

[13] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui. Geomf: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *KDD, New York, NY, USA, August 24-27*, pages 831–840, 2014.

[14] G. Liao, Y. Zhao, S. Xie, and P. S. Yu. An effective latent networks fusion based model for event recommendation in offline ephemeral social networks. In *CIKM, San Francisco, CA, USA, October 27-November 1*, pages 1655–1660, 2013.

[15] B. Liu, Y. Fu, Z. Yao, and H. Xiong. Learning geographical preferences for point-of-interest recommendation. In *KDD, Chicago, IL, USA, August 11-14*, pages 1043–1051, 2013.

[16] X. Liu, Q. He, Y. Tian, W. Lee, J. McPherson, and J. Han. Event-based social networks: linking the online and offline social worlds. In *KDD, Beijing, China, August 12-16*, pages 1032–1040, 2012.

[17] H. Ma, C. Liu, I. King, and M. R. Lyu. Probabilistic factor models for web site recommendation. In *SIGIR, Beijing, China, July 25-29*, pages 265–274, 2011.

[18] E. Minkov, B. Charrow, J. Ledlie, S. J. Teller, and T. Jaakkola. Collaborative future event recommendation. In *CIKM, Toronto, Ontario, Canada, October 26-30*, pages 819–828, 2010.

[19] M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The Adaptive Web, Methods and Strategies of Web Personalization*, pages 325–341, 2007.

[20] T.-A. N. Pham, X. Li, G. Cong, and Z. Zhang. A general graph-based model for recommendation in event-based social networks. In *ICDE, Seoul, Korea, April 13-17*, 2015.

[21] Z. Qiao, P. Zhang, Y. Cao, C. Zhou, L. Guo, and B. Fang. Combining heterogenous social and geographical information for event recommendation. In *AAAI, July 27 -31, Québec City, Québec, Canada.*, pages 145–151, 2014.

[22] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS, Vancouver, British Columbia, Canada, December 3-6*, pages 1257–1264, 2007.

[23] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *ICML, Helsinki, Finland, June 5-9*, pages 880–887, 2008.

[24] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR, August 11-15, Tampere, Finland*, pages 253–260, 2002.

[25] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *SIGIR, Las Vegas, Nevada, USA, August 24-27*, pages 650–658, 2008.

[26] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *KDD, San Diego, CA, USA, August 21-24*, pages 448–456, 2011.

[27] M. Ye, P. Yin, W. Lee, and D. L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR, Beijing, China, July 25-29*, pages 325–334, 2011.

[28] W. Zhang, J. Wang, and W. Feng. Combining latent factor model with location features for event-based group recommendation. In *KDD, Chicago, IL, USA, August 11-14*, pages 910–918, 2013.

[29] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with GPS history data. In *WWW, Raleigh, North Carolina, USA, April 26-30*, pages 1029–1038, 2010.