Learn to Recommend Local Event Using Heterogeneous Social Networks

Shaoqing Wang[1,2], Zheng Wang[1], Cuiping Li[1](✉), Kankan Zhao[1], and Hong Chen[1]

[1] Key Lab of Data Engineering and Knowledge Engineering of MOE, Renmin University of China, Beijing, China
{wsq,zheng.wang,licuiping,zhaokankan,chong}@ruc.edu.cn
[2] School of Computer Science and Technology, Shandong University of Technology, Zibo, China

Abstract. Event-based social networks (EBSNs), which link the online and offline social networks, are increasing popular online services. Along with dramatic rise of the users and events in EBSNs, it is necessary to recommend event to users. Taking full advantage of social networks information can significantly improve predictive accuracy in recommender systems. The intuition here is that the user's response to events are determined by his/her instinct and behaviours of friends. We propose a Heterogeneous Social Poisson Factorization(HSPF) model which combines online and offline social networks into one framework, and integrates the tie strength of online and offline friend relationships to the model. We test HSPF on Meetup dataset. Experimental results demonstrate that HSPF outperforms state-of-the-art recommendation methods.

Keywords: Event recommendations · Social recommendation · Poisson factorization · Event-based social networks

1 Introduction

With the rapid development of event-based social network services, it becomes increasing popular to participate local events through the online services such as Meetup (www.meetup.com) and Douban Events (beijing.douban.com/events). Event-based social networks (EBSNs) not only contain online social networks by joining the same groups where users can organize, participate, comment,

This work is supported by National Basic Research Program of China(973)(No. 2014CB340403, No.2012CB316205), National High Technology Research and Development Program of China (863) (No.2014AA015204) and NSFC under the grant No.61272137, 61033010, 61202114, 61532021, 61502421 and NSSFC (No.12&ZD220), and the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China(15XNLQ06). It was partially done when the authors worked in SA Center for Big Data Research in RUC. This Center is funded by a Chinese National 111 Project Attracting.

F. Li et al. (Eds.): APWeb 2016, Part I, LNCS 9931, pp. 169–182, 2016.
DOI: 10.1007/978-3-319-45814-4_14

share and advertise events, but also include offline social networks where users make friends through attending the same face-to-face activities. The core goal of EBSNs is to make easy for neighbors (users located in the same city) together to do what they are commonly interested in [20]. To better organize events, these services allow users to join online groups, in which a user can publish and announce events to other group members [14]. For example, the online social networks on Meetup are the groups that organizers create and other numbers join, while the offline social networks on Meetup are captured in offline activities. Consequently, the users in EBSNs have two kinds of heterogeneous friend relationships that one is the online friend relationships and another is offline friend relationships.

Along with dramatic rise of the users and events in EBSNs, how to choose the interesting events for users is become very important and difficult. Meetup, the world's largest network of local groups, currently has 25.72 million users with more than 240,000 Meetup groups and 580,000 monthly events[1]. Therefore, it becomes essential to recommend from so many dazzling events to solve information overload problem. In contrast to traditional social network services (SNS), user behavior in EBSNs is predominantly driven by offline activities and highly influenced by a set of unique factors, such as spatio-temporal constraints and special social relationships [3]. As a result, event recommendation in EBSNs becomes very difficult. In addition, the event recommendation problem is arguably more challenging than classic recommendation scenarios (e.g. movies, books), since events have time limited efficacy, which means that event recommender systems have to recommend the event after it created and before it terminate.

Generally, users can only participate local events because of the limitation of distance. Different from existing recommendation problems, the characteristic of users' friend relationships plays very important role in event recommendation. Particularly worth mentioning is that online and offline friend relationship are not identical in EBSNs. On Meetup, the online friends mean the users who are in the same online group, and the offline friends mean the users who participate the same offline activities.

In this paper, we introduce online and offline friend relationships to recommend events to users and also introduce the tie strength of online and offline friend relationships. Based on these factors and the multi-factor model, we propose a novel method, named Heterogeneous Social Poisson Factorization (HSPF). In summary, the main contributions of this work lie in the following three aspects:

1. We propose the Heterogeneous Social Poisson Factorization(HSPF) model which combines online and offline social networks into one framework.
2. We integrate the tie strength of online and offline friend relationships to proposed HSPF model and use the coordinate ascent alogrithm for inference of the HSPF model parameters.
3. Our experiments on the Meetup dataset show that for the task of event recommendation, our HSPF model outperforms other models.

[1] http://www.meetup.com/about.

2 Related Works

2.1 Social Recommendation

Recent studies have proposed various methods to include social information in matrix factorization process [19]. For instance, [8,11,12] include trust in recommendation process; zhou et al. [21] exploits users interactions to improve recommendation qualities; duan et al. [4] uses locations to build a personalized recommendation system; and chaney et al. [2] proposes SPF (Social Poisson Factorization) model using trusted friend relationships. All the above methods for recommender systems only include a single factor. In our work, we not only divide trusted friends into online and offline friends, but also introduce the tie strength of trusted friends.

2.2 Event Recommendation

Currently, there are a small amount research works studying on event recommendation in EBSNs. EBSNs are first analyzed in data mining field in [10]. Du et al. [3] explores the modeling of EBSNs users by utilizing content preference, spatio-temporal context, and social influence (the event organizer) features. Pham et al. [14] transforms the recommendation problems into node proximity calculation problem and proposes a general graph-based model. Macedo et al. [13] takes several context-aware recommenders as input features such as content-based, social, locational and temporal signals. Zhang et al. [20] proposes CBPF (collective Bayesian Poisson factorization) model utilizing users' relationship information and events organizer, location, and textual content information to recommend local events. However, all the above methods do not introduce the heterogenous online+offline social relationships or cannot reflect the difference between online and offline social relationships. Qiao et al. [15] presents a Bayesian latent factor mode that unify the data, i.e., the geographical features, heterogenous online+offline social relationships and user implicit rating, for event recommendation. But it ignores the tie strength of the social relationships.

3 Preliminaries

Bayesian Poisson Factorization (BPF), proposed by Gopalan [5], is a probabilistic model of users and items for recommendation. BPF assumes that an observed rating matrix y_{ui} comes from a Poisson distribution:

$$y_{ui} \sim Poisson(\theta_u^T \beta_i)$$

where θ_u is a non-negative K-vector of user preference and β_i is a non-negative K-vector of item attributes. θ_u and β_i are hidden variables with Gamma priors:

$$\theta_{u,k} \sim Gamma(\lambda_{ua}, \lambda_{ub})$$
$$\beta_{i,k} \sim Gamma(\lambda_{ia}, \lambda_{ib})$$

where λ_{ua} and λ_{ia} are the shape parameters of the Gamma distribution and λ_{ub} and λ_{ib} are the rate parameters of the Gamma distribution.

BPF can handle sparse data well and is more robust to the issue of overfitting [20]. We build on BPF to develop a model of data where users attend face-to-face offline events and the same users are organized in a online network.

4 Proposed HSPF Model

In this section, we describe the Heterogeneous Social Poisson Factorization(HSPF). We are given data about users, events and groups, where each user who belongs to some groups has attend some events. The groups mean the online social networks while the events represent the offline social networks.

HSPF is a latent variable model of user-event interaction and user-group interaction. HSPF uses Poisson factorization to model both of the interactions that are typically sparse. The user's responses to events are determined by his/her instinct and behaviours of friends. That is the users would attend the events that are not consistent with his/her preference, just because that his/her friends attend it. The intuition is that the closer the relationship between of them, the greater the impact of the choices of RSVPs(Reply, if you please). EBSNs link the online and offline social worlds [10]. For different users, the impact of online and offline social worlds is different. As the graphical model of HSPF shown in Fig. 1, HSPF captures this intuition because there are three parts in the model: user's preference θ_u, online social network influence δ_{uv}, and offline social network influence τ_{uf}. Then, we provide modeling details for the Heterogeneous Social Poisson Factorization(HSPF).

Heterogeneous Social Poisson Factorization(HSPF). We assume that the rating y_{ui} of a (user u, event i) pair equals one if the choice of RSVP is yes, i.e., the user u attends event i, and is zero otherwise. The relation r_{uv} of a (user u, online friend v) pair is regarded as the degree of online closeness between them, which can be computed by the number of groups that both of them belong to. The relation r_{uf} of a (user u, offline friend f) pair is considered as the degree of offline closeness between them, which is in proportion to the number of events that both of them attend. Each user u is represented by a vector of K latent preference θ_u and each event i by a vector of K latent attributes β_i. $N^{on}(u)$ and $N^{off}(u)$ are the set of online and offline social friends of user u respectively. δ_{uv} and τ_{uf} are the influences of online friend v and offline friend f respectively.

The distribution of the observation y_{ui} is denoted by

$$y_{ui}|y_{\neg ui} \sim Poisson(\theta_u^T\beta_i + \sum_{f\in N^{off}(u)} \tau_{uf} r_{uf} y_{fi} + \sum_{v\in N^{on}(u)} \delta_{uv} r_{uv} y_{vi}), \qquad (1)$$

where $y_{\neg ui}$ denotes the responses of other users. To complete the specification of the variables, we place Gamma priors with shape and rate parameters on the user's preference θ_{uk}, item's attribute β_{ik}, online social network influence δ_{uv}, and offline social network influence τ_{uf}. This is because that Gamma distribution

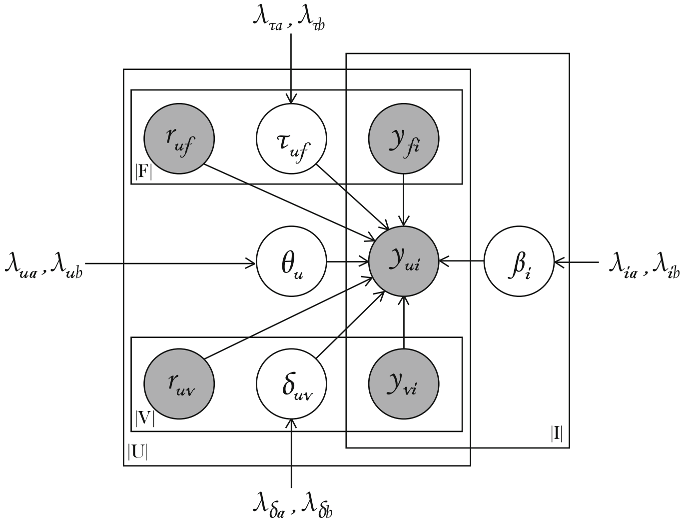

Fig. 1. Graphical model of the Heterogeneous Social Poisson Factorization(HSPF)

is the conjugate prior for Poisson distribution. It is very facilitate to Bayesian learning of model parameters. Furthermore, the Gamma prior encourages sparse representations of users, events and influences. More specifically, by setting the shape parameters, i.e., λ_{ua},λ_{ia},$\lambda_{\tau a}$ and $\lambda_{\delta a}$, to be small (e.g., 0.3), most of the generated values will be close to zero. Note that λ_{ub},λ_{ib},$\lambda_{\tau b}$ and $\lambda_{\delta b}$ are rate parameters.

Based on the above description, the generative process of the Heterogeneous Social Poisson Factorization(HSPF) is as follows:

(1) For each user $u = 1, ..., U$ and each component $k = 1, ..., K$, draw latent factor
$$\theta_{uk} \sim Gamma(\lambda_{ua}, \lambda_{ub}).$$
(2) For each event $i = 1, ..., I$ and each component $k = 1, ..., K$, draw latent factor
$$\beta_{ik} \sim Gamma(\lambda_{ia}, \lambda_{ib}).$$
(3) For each influence δ_{uv} of (user u, online-friend v) pair, draw latent factor
$$\delta_{uv} \sim Gamma(\lambda_{\delta a}, \lambda_{\delta b}).$$
(4) For each influence τ_{uf} of (user u, offline-friend f) pair, draw latent factor
$$\tau_{uf} \sim Gamma(\lambda_{\tau a}, \lambda_{\tau b}).$$
(5) For each (user u, event i) pair, draw response y_{ui} through Eq. (1).

5 Inference Algorithm

The key inferential problem that we need to solve in order to use HSPF is posterior inference. We denote all the Gamma priors of latent factors with λ. Given $\Theta = \{\theta_{uk}, \beta_{ik}, \delta_{uv}, \tau_{uf}\}$, $\mathrm{Y} = \{y_{ui}, r_{uv}, r_{uf}\}$, then the posterior distribution,

$$p(\Theta|\mathrm{Y}, \lambda) = \frac{p(\Theta, \mathrm{Y}|\lambda)}{p(\mathrm{Y}|\lambda)} = \frac{p(\Theta, \mathrm{Y}|\lambda)}{\int p(\Theta, \mathrm{Y}|\lambda)d\Theta} \tag{2}$$

which is intractable for exact inference due to the coupling between integration variables of the normalization term shown in Eq. (2).

We adopt the variational inference to approximate the posterior distribution because that the variational inference tends to scale better than alternative sampling based algorithm such as Markov chain Monte Carlo. Variational inference algorithms approximate the posterior by defining a parameterized family of distributions over the hidden variables, i.e., $q(\Theta|\gamma)$, and then fitting the parameters of $q(\Theta|\gamma)$ that is close to the posterior, i.e., $p(\mathrm{Y}|\lambda)$.

The basic idea of convexity-based variational inference is to take advantage of Jensen's inequality to obtain an evidence lower bound on the log likelihood [9].

$$\begin{aligned}
&logp(\mathrm{Y}|\lambda)\\
&= log\int p(\Theta, \mathrm{Y}|\lambda)d\Theta\\
&= log\int \frac{q(\Theta|\gamma)}{q(\Theta|\gamma)}p(\Theta, \mathrm{Y}|\lambda)d\Theta\\
&\geq \int q(\Theta|\gamma)logp(\Theta, \mathrm{Y}|\lambda)d\Theta - \int q(\Theta|\gamma)logq(\Theta|\gamma)d\Theta\\
&= E_q[logp(\Theta, \mathrm{Y}|\lambda)] - E_q[logq(\Theta|\gamma)]\\
&= \mathcal{L}(\gamma; \lambda)
\end{aligned} \tag{3}$$

The difference between $logp(\mathrm{Y}|\lambda)$ and $\mathcal{L}(\gamma; \lambda)$ of Eq. (3) is the Kullback-Leibler(KL) divergence between $q(\Theta|\gamma)$ and $p(\mathrm{Y}|\lambda)$, that is

$$logp(\mathrm{Y}|\lambda) = \mathcal{L}(\gamma; \lambda) + D(q(\Theta|\gamma)||p(\mathrm{Y}|\lambda)). \tag{4}$$

This shows that maximizing the evidence lower bound $\mathcal{L}(\gamma; \lambda)$ equivalent to minimizing the KL divergence, thus the problem of posterior inference becomes an optimization problem.

5.1 Auxiliary Variables

Our inference algorithm for HSPF makes use of general results about the class of conditionally conjugate models [5,7]. We first give an alternative formulation of HSPF in which we add some auxiliary variables to facilitate derivation and description of the algorithm. Without the auxiliary variables, HSPF is not conditionally conjugate model.

Note that a sum of independent Poisson random variables is itself a Poisson with rate equal to the sum of the rates. We introduce the auxiliary latent variables z^M_{uik}, z^{off}_{uif} and z^{on}_{uiv} for each (user u, event i) pair, each (user u, offline friend f) pair and each (user u, online friend v) pair respectively such that

$$y_{ui}|y_{\neg ui} = \sum_{k=1}^{K} z^M_{uik} + \sum_{f=1}^{F} z^{off}_{uif} + \sum_{v=1}^{V} z^{on}_{uiv}, \tag{5}$$

where

$$\begin{aligned} z^M_{uik} &\sim Poisson(\theta_{uk}\beta_{ik}), \\ z^{off}_{uif} &\sim Poisson(\tau_{uf} r_{uf} y_{fi}), \\ z^{on}_{uiv} &\sim Poisson(\delta_{uv} r_{uv} y_{vi}), \end{aligned}$$

and $F = |N^{off}(u)|$, $V = |N^{on}(u)|$.

After adding the auxiliary variables, the variational distribution, i.e., $q(\Theta|\gamma)$, turns to $q(\Theta, Z|\gamma, \phi)$ where Z and ϕ denote all the auxiliary variables and added parameters respectively.

5.2 Mean-Field Variational Family

We resort to the mean-field variational family, where each latent variable is independent and governed by its own variational parameter. Omitting the parameters γ and ϕ for simplicity, the mean-field variational family is

$$q(\Theta, Z) = \prod_{u,k} q(\theta_{uk}) \prod_{i,k} q(\beta_{ik}) \prod_{u,f} q(\tau_{uf}) \prod_{u,v} q(\delta_{uv}) \prod_{u,i,k} q(z_{uik}),$$

Each factor in the mean-field family usually is set to the same type of distribution as its *complete conditional* [2,5,6,20]. A *complete conditional* which is the conditional distribution of a latent variable given the observations and other latent variables in the model.

Complete Conditional. Firstly, we compute the *complete conditional*s of all the latent variables in the model. For the user preferences θ_{uk}, the *complete conditional* is a Gamma shown as follows,

$$\begin{aligned} &p(\theta_{uk}|\lambda, \beta, \tau, \delta, z, y) \\ &\propto p(\theta_{uk}|\lambda_{ua}, \lambda_{ub}) \prod_i p(z^M_{uik}|\theta_{uk}, \beta_{ik}) \\ &= Gamma(\theta_{uk}; \lambda_{ua}, \lambda_{ub}) \prod_i Poisson(z^M_{uik}; \theta_{uk}, \beta_{ik}) \\ &= Gamma(\lambda_{ua} + \sum_i z^M_{uik}, \lambda_{ub} + \sum_i \beta_{ik}). \end{aligned} \tag{6}$$

We can similarly derive the *complete conditional*s for β_{ik}, τ_{uf}, and δ_{uv}.

$$\beta_{ik}|\lambda, \theta, \tau, \delta, z, y \sim Gamma(\lambda_{ia} + \sum_u z^M_{uik}, \lambda_{ib} + \sum_u \theta_{uk}), \tag{7}$$

$$\tau_{uf}|\lambda,\theta,\beta,\delta,z,y \sim Gamma(\lambda_{fa}+\sum_i z_{uif}^{off}, \lambda_{fb}+\sum_i r_{uf}y_{fi}), \quad (8)$$

$$\delta_{uv}|\lambda,\theta,\beta,\tau,z,y \sim Gamma(\lambda_{va}+\sum_i z_{uiv}^{on}, \lambda_{vb}+\sum_i r_{uv}y_{vi}) \quad (9)$$

The conditional distribution of a set of Poisson variables, given their sum, is a multinomial for which the parameter is their normalized set of rates [1]. So the *complete conditionals* for $z_{ui} = (z_{ui}^M, z_{ui}^{off}, z_{ui}^{on})$ is multinomial, i.e., $z_{ui} \sim Mult(y_{ui}, \psi_{ui})$, where $\psi_{ui} = (\psi_{ui}^M, \psi_{ui}^{off}, \psi_{ui}^{on})$ is a point in the $(K+F+V)$-simplex, i.e.,

$$\psi_{ui}^M \propto \langle \theta_{u1}\beta_{i1}, ..., \theta_{uK}\beta_{iK}\rangle,$$
$$\psi_{ui}^{off} \propto \langle \tau_{u1}r_{u1}y_{1i}, ..., \tau_{uF}r_{uF}y_{Fi}\rangle,$$
$$\psi_{ui}^{on} \propto \langle \delta_{u1}r_{u1}y_{1i}, ..., \delta_{uV}r_{uV}y_{Vi}\rangle,$$

Note that the parameters of $\psi_{ui}^M, \psi_{ui}^{off}$ and ψ_{ui}^{on} should be normalized together to ensure their sum to be one.

Variational Parameter. Then, we set each factor in the mean-field family to be the same type of distribution as its complete conditional.

The complete conditionals of $\theta_{uk}, \beta_{ik}, \delta_{uv}$, and τ_{uf} are Gamma distributions, so their variational parameters are Gamma parameters, i.e., variational distributions for $\theta_{uk}, \beta_{ik}, \tau_{uf}$, and δ_{uv} are $Gamma(\gamma_{uk}^{shp}, \gamma_{uk}^{rte})$, $Gamma(\gamma_{ik}^{shp}, \gamma_{ik}^{rte})$, $Gamma(\gamma_{uf}^{shp}, \gamma_{uf}^{rte})$, and $Gamma(\gamma_{uv}^{shp}, \gamma_{uv}^{rte})$ respectively. We can similarly deduce the variational distribution for $z_{ui} = (z_{ui}^M, z_{ui}^{off}, z_{ui}^{on})$ is $Mult(y_{ui}, \phi_{ui})$, where $\phi_{ui} = (\phi_{ui}^M, \phi_{ui}^{off}, \phi_{ui}^{on})$.

We set each variational parameter equal to the expected parameter (under q) of the complete conditional because of the conditionally conjugate models [7].

For variational Gamma distributions, we take θ_{uk} as an example to derive the close-form update solution of parameters, i.e., shape parameter γ_{uk}^{shp} and rate parameter γ_{uk}^{rte}.

$$\gamma_{uk}^{shp} = E_q[\lambda_{ua} + \textstyle\sum_i z_{uik}^M] = \lambda_{ua} + \displaystyle\sum_i y_{ui}\phi_{uik}^M, \quad (10)$$

$$\gamma_{uk}^{rte} = E_q[\lambda_{ub} + \textstyle\sum_i \beta_{ik}] = \lambda_{ub} + \displaystyle\sum_i \frac{\gamma_{ik}^{shp}}{\gamma_{ik}^{rte}}, \quad (11)$$

The update solutions for the parameters of β_{ik}, τ_{uf}, and δ_{uv} are similarly derived, we omit the details and provide the final results.

$$\gamma_{ik}^{shp} = \lambda_{ia} + \sum_u y_{ui}\phi_{uik}^M, \quad (12)$$

$$\gamma_{ik}^{rte} = \lambda_{ib} + \sum_u \frac{\gamma_{uk}^{shp}}{\gamma_{uk}^{rte}}, \quad (13)$$

$$\gamma_{uf}^{shp} = \lambda_{fa} + \sum_i y_{uf} \phi_{uif}^{off}, \tag{14}$$

$$\gamma_{uf}^{rte} = \lambda_{fb} + \sum_i r_{uf} y_{fi}, \tag{15}$$

$$\gamma_{uv}^{shp} = \lambda_{va} + \sum_i y_{uv} \phi_{uiv}^{on}, \tag{16}$$

$$\gamma_{uv}^{rte} = \lambda_{vb} + \sum_i r_{uv} y_{vi}. \tag{17}$$

For variational multinomial distribution, we take z_{ui}^M as an example to derive update solution of parameter, i.e., ϕ_{ui}^M.

$$\begin{aligned}
\phi_{uik}^M &\propto G_q[\theta_{uk}\beta_{ik}] \\
&= exp\{E_q[log(\theta_{uk}) + log(\beta_{ik})]\} \\
&= exp\{E_q[log(\theta_{uk})] + E_q[log(\beta_{ik})]\} \\
&= exp\{\Psi(\gamma_{uk}^{shp}) - log\gamma_{uk}^{rte} + \Psi(\gamma_{ik}^{shp}) - log\gamma_{ik}^{rte})\},
\end{aligned} \tag{18}$$

where $k = 1, ..., K$; $\Psi(\cdot)$ is the digamma function and $G_q[\cdot] = exp(E_q[log(\cdot)])$ denotes the geometric expectation [17]. Similarly,

$$\phi_{uif}^{off} \propto exp\{\Psi(\gamma_{uf}^{shp}) - log\gamma_{uf}^{rte}\} + r_{uf} y_{fi}, \tag{19}$$

$$\phi_{uiv}^{on} \propto exp\{\Psi(\gamma_{uv}^{shp}) - log\gamma_{uv}^{rte}\} + r_{uv} y_{vi}, \tag{20}$$

where $f = 1, ..., F$ and $v = 1, ..., V$. Note that the parameters of $\phi_{ui}^M, \phi_{ui}^{off}$ and ϕ_{ui}^{on} should be normalized together to ensure their sum to be one.

5.3 Coordinate Ascent Algorithm

The coordinate ascent algorithm is illustrated in Fig. 2, which iteratively optimize each variational parameter while holding the others fixed.

5.4 Event Recommendation Using HSPF

Once the posterior is fit, we use the HSPF to recommend events, which are unattend and ongoing events, to users who will like to attend. Firstly, we compute the predicting scores $\hat{y}_{ui}$ (i.e., Eq. 21) for each (user u, unattend and ongoing event i) pair by their posterior expected Poisson parameters. Then, we rank the events for each user using the scores. Lastly, Top-N events will be recommended to each user.

$$\begin{aligned}
\hat{y}_{ui} &= E_q[\theta_u^T \beta_i + \sum_{f \in N^{off}(u)} \tau_{uf} r_{uf} y_{fi} + \sum_{v \in N^{on}(u)} \delta_{uv} r_{uv} y_{vi}] \\
&= \sum_{k=1}^K E_q[\theta_{uk}\beta_{ik}] + \sum_{f \in N^{off}(u)} E_q[\tau_{uf}] r_{uf} y_{fi} + \sum_{v \in N^{on}(u)} E_q[\delta_{uv}] r_{uv} y_{vi} \\
&= \sum_{k=1}^K \frac{\gamma_{uk}^{shp}}{\gamma_{uk}^{rte}} \frac{\gamma_{ik}^{shp}}{\gamma_{ik}^{rte}} + \sum_{f \in N^{off}(u)} \frac{\gamma_{uf}^{shp}}{\gamma_{uf}^{rte}} r_{uf} y_{fi} + \sum_{v \in N^{on}(u)} \frac{\gamma_{uv}^{shp}}{\gamma_{uv}^{rte}} r_{uv} y_{vi}
\end{aligned} \tag{21}$$

Initialize parameters of variational Gamma distributions.
Repeat until convergence:

1. For each observation $y_{ui} > 0$ of (user u, event i) pair, update the multinomial using Equation (18), (19) and (20).
2. For each user u, update the parameter using Equation (10) and (11).
3. For each event i, update the parameter using Equation (12) and (13).
4. For influence of each offline friend f, update the parameter using Equation (14) and (15).
5. For influence of each online friend v, update the parameter using Equation (16) and (17).

Fig. 2. The coordinate ascent algorithm for HSPF

6 Experiments

6.1 Dataset

We conduct experiments on the Meetup dataset [13]. The dataset contains all public activities on Meetup from January, 2010 to April, 2014, which we use to evaluate the HSPF model proposed in this work. We select two cities located in the USA, namely Chicago and Phoenix Table 1.

Table 1. Statistics of the Meetup dataset.

City	|G|	|U|	|E|	RSVPs
Chicago	2,138	133,357	100,701	810,213
Phoenix	842	43,112	64,255	326,913

6.2 Setup and Metrics

We select data during each of last 10 months as the test set for each city. The data during preceding 6 months of each test set is regarded as the corresponding training set. We firstly compute the scores for each (user u, event i) pair in the test set, then sort the triplets by these scores according to user, lastly recommend Top-N events to each user. We report the average value of the predicted performance over the 10 test set. We choose NDCG@N and Precision@N as the metrics.

6.3 Comparison Methods

To demonstrate the effectiveness of our HSPF model, we compare it to the following state-of-the-art recommenders:

- **CMF** [18] solves the recommendation task by reconstructing multiple relation matrices which are associated with some shared elements. In this work, the users are the shared elements.
- **FM** [16] a general predictor and is easily applicable to a wide variety of contexts, which include user, event, online and offline social networks in Meetup dataset, by specifying only the input data.
- **BPF**, proposed by Gopalan [5], is a probabilistic model of users and items for recommendation. BPF can handle sparse data well and models the long-tail of users and items.
- **SPF-on** and **SPF-off** stem from SPF [2] which aims to bridge the gap between preference- and social-based recommendations. However, SPF only incorporates one kind of social networks information into a Poisson factorization method and can't capture the tie strength of friend relationships. **SPF-on** and **SPF-off** means only incorporates online and offline social networks respectively.
- **HSPF-on** and **HSPF-off** are two variants of our HSPF model and can be considered as two special cases of our model. The difference of them is that they only consider the influence of online and offline social networks respectively.

6.4 Performance Comparison

For HSPF, the latent dimensionality, i.e., K, is set to 50 for both Chicago and Phoenix dataset. The Gamma priors parameters, i.e., shape and rate parameters, of all the latent factors are fixed to be 0.3.

Effectiveness Comparison. The detailed comparison results are shown in Fig. 3. From the Fig. 3, we can observe that:

- The performance of **HSPF**,**CMF** and **FM** are better than **SPF-off** and **SPF-on**. The reason is that either **SPF-off** or **SPF-on** can only integrate one kind of social networks. The performance of event recommendation can be improved by integrating the heterogeneous social networks.

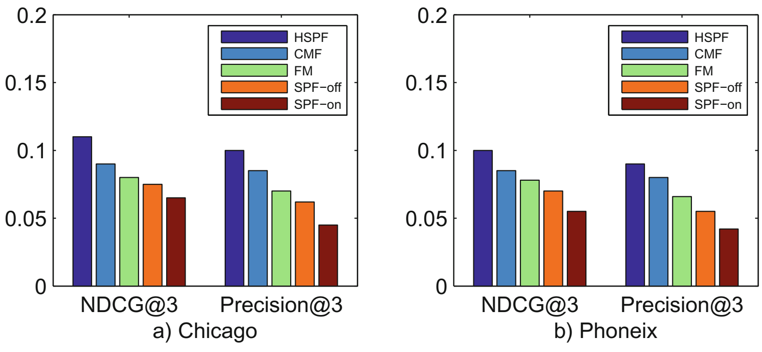

Fig. 3. Comparisons of different methods on NDCG@N and Precision@N

– The proposed **HSPF** model achieves the best performance comparing to the other models. It can be explained that the user responses to events are implicit feedback [5] while HSPF is more suitable for modeling implicit user feedback.

Factor Contribution. From the detailed comparison results which are shown in Fig. 4, we can draw the conclusions that:

– The performance of **HSPF**,**HSPF-off** and **HSPF-on** outperforms that of **BPF** which can be considered as a special case of HSPF without social network information. It shows that the performance of recommender systems can be improved by integrating the social network information into the model.
– **HSPF-off** achieves better results than **HSPF-on**, which means that the influence of offline social networks is greater than the online social network. This explains that face to face communication helps to become a true friend.

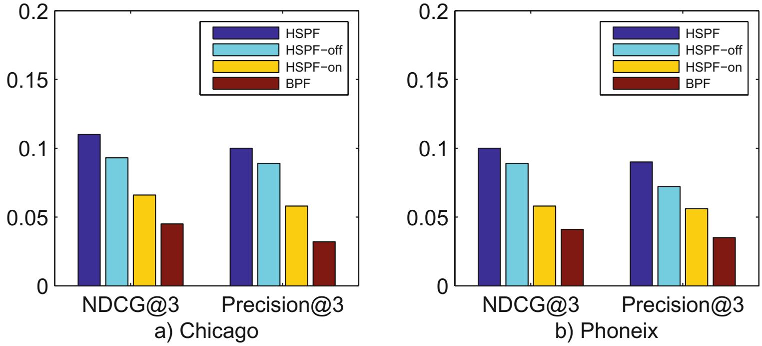

Fig. 4. Factor contribution of HSPF model

7 Conclusion

In this work, we propose a Heterogeneous Social Poisson Factorization(HSPF) model which combines online and offline social networks into one framework, and integrates the tie strength of online and offline friend relationships to the model. We test HSPF on Meetup dataset. Experimental results demonstrate that HSPF outperforms state-of-the-art recommendation methods. In the future work, we plan to exploit the influence of time, location, and content.

References

1. Cemgil, A.T.: Bayesian inference for nonnegative matrix factorisation models. Comput. Intell. Neurosci. **2009** (2009)
2. Chaney, A.J., Blei, D.M., Eliassi-Rad, T.: A probabilistic model for using social networks in personalized item recommendation. In: Proceedings of the 9th ACM Conference on Recommender Systems, pp. 43–50. ACM (2015)
3. Du, R., Yu, Z., Mei, T., Wang, Z., Wang, Z., Guo, B.: Predicting activity attendance in event-based social networks: content, context and social influence. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 425–434. ACM (2014)
4. Duan, R., Goh, R.S.M., Yang, F., Tan, Y.K., Valenzuela, J.F.: Towards building and evaluating a personalized location-based recommender system. In: 2014 IEEE International Conference on Big Data (Big Data), pp. 43–48. IEEE (2014)
5. Gopalan, P., Hofman, J.M., Blei, D.M.: Scalable recommendation with poisson factorization (2013). arXiv preprint arXiv:1311.1704
6. Gopalan, P.K., Charlin, L., Blei, D.: Content-based recommendations with poisson factorization. In: Advances in Neural Information Processing Systems, pp. 3176–3184 (2014)
7. Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J.: Stochastic variational inference. J. Mach. Learn. Res. **14**(1), 1303–1347 (2013)
8. Jamali, M., Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks. In: Proceedings of the Fourth ACM Conference on Recommender systems, pp. 135–142. ACM (2010)
9. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Mach. Learn. **37**(2), 183–233 (1999)
10. Liu, X., He, Q., Tian, Y., Lee, W.C., McPherson, J., Han, J.: Event-based social networks: linking the online and offline social worlds. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1032–1040. ACM (2012)
11. Ma, H., Yang, H., Lyu, M.R., King, I.: Sorec: social recommendation using probabilistic matrix factorization. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, pp. 931–940. ACM (2008)
12. Ma, H., Zhou, D., Liu, C., Lyu, M.R., King, I.: Recommender systems with social regularization. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 287–296. ACM (2011)
13. Macedo, A.Q., Marinho, L.B., Santos, R.L.: Context-aware event recommendation in event-based social networks. In: Proceedings of the 9th ACM Conference on Recommender Systems, pp. 123–130. ACM (2015)
14. Pham, T.A.N., Li, X., Cong, G., Zhang, Z.: A general graph-based model for recommendation in event-based social networks. In: 2015 IEEE 31st International Conference on Data Engineering (ICDE), pp. 567–578. IEEE (2015)
15. Qiao, Z., Zhang, P., Cao, Y., Zhou, C., Guo, L., Fang, B.: Combining heterogenous social and geographical information for event recommendation. In: Twenty-Eighth AAAI Conference on Artificial Intelligence (2014)
16. Rendle, S., Gantner, Z., Freudenthaler, C., Schmidt-Thieme, L.: Fast context-aware recommendations with factorization machines. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 635–644. ACM (2011)

17. Schein, A., Paisley, J., Blei, D.M., Wallach, H.: Bayesian poisson tensor factorization for inferring multilateral relations from sparse dyadic event counts. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1045–1054. ACM (2015)
18. Singh, A.P., Gordon, G.J.: Relational learning via collective matrix factorization. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 650–658. ACM (2008)
19. Yang, X., Guo, Y., Liu, Y., Steck, H.: A survey of collaborative filtering based social recommender systems. Comput. Commun. **41**, 1–10 (2014)
20. Zhang, W., Wang, J.: A collective bayesian poisson factorization model for cold-start local event recommendation. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1455–1464. ACM (2015)
21. Zhou, T., Shan, H., Banerjee, A., Sapiro, G.: Kernelized probabilistic matrix factorization: exploiting graphs and side information. In: SDM, vol. 12, pp. 403–414. SIAM (2012)