Context-Aware Event Recommendation in Event-based Social Networks

Augusto Q. Macedo
Dept. of Syst. and Computing
Fed. Univ. of Campina Grande
Campina Grande, Brazil
augusto@copin.ufcg.edu.br

Leandro B. Marinho
Dept. of Syst. and Computing
Fed. Univ. of Campina Grande
Campina Grande, Brazil
lbmarinho@dsc.ufcg.edu.br

Rodrygo L. T. Santos
Dept. of Computer Science
Fed. Univ. of Minas Gerais
Belo Horizonte, Brazil
rodrygo@dcc.ufmg.br

ABSTRACT

The Web has grown into one of the most important channels to communicate social events nowadays. However, the sheer volume of events available in event-based social networks (EBSNs) often undermines the users' ability to choose the events that best fit their interests. Recommender systems appear as a natural solution for this problem, but differently from classic recommendation scenarios (e.g. movies, books), the event recommendation problem is intrinsically cold-start. Indeed, events published in EBSNs are typically short-lived and, by definition, are always in the future, having little or no trace of historical attendance. To overcome this limitation, we propose to exploit several contextual signals available from EBSNs. In particular, besides content-based signals based on the events' description and collaborative signals derived from users' RSVPs, we exploit social signals based on group memberships, location signals based on the users' geographical preferences, and temporal signals derived from the users' time preferences. Moreover, we combine the proposed signals for learning to rank events for personalized recommendation. Thorough experiments using a large crawl of Meetup.com demonstrate the effectiveness of our proposed contextual learning approach in contrast to state-of-the-art event recommenders from the literature.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; H.2.8 [**Database Applications**]: Data mining

General Terms

Algorithms; Experimentation

Keywords

Event-based social networks; recommender systems

RecSys'15, September 16–20, 2015, Vienna, Austria.
© 2015 ACM. ISBN 978-1-4503-3692-5/15/09 ...$15.00.
DOI: http://dx.doi.org/10.1145/2792838.2800187.

1. INTRODUCTION

Event-based social networks (EBSNs) are online social networks where users can create, promote and share upcoming events of any kind with other users. For instance, Meetup.com,[1] one of the largest EBSNs available today, registered 3,000 new groups and 330,000 users participating in events in the first week of January, 2015 alone.[2] In Meetup.com, users can create and join groups of like-minded people, promoting events (aka Meetups) that they are interested in. However, the sheer volume of available events, especially in large and touristic cities, often undermines the users' ability to find the ones that best match their interests.

Recommender systems appear as a natural solution to overcome such an information overload, as they help users discover relevant information in large data sets. Nonetheless, the event recommendation problem is arguably more challenging than classic recommendation scenarios (e.g. movies and books), since event recommenders need to deal with the new item cold-start problem that arises naturally in this setting [14]. Indeed, events published in EBSNs are typically short-lived and, by definition, are always in the future, having little or no trace of historical attendance. In this scenario, classic recommendation approaches may underperform, leaving the EBSN users unsatisfied.

To overcome this limitation, we propose a context-aware approach to event recommendation, by exploiting several contextual signals generally available in EBSNs. Besides content-based signals based on the events' description and collaborative signals derived from users' RSVPs,[3] we exploit social signals based on group memberships, location signals based on the users' home distance to each event, and temporal signals derived from the users' time preferences. In particular, our assumption is that each of these signals has a positive influence on the users' decision about attending an event. For example, a user may decide to attend an event not only because the event matches his or her interests and other like-minded users have RSVP'ed to the event, but also because the event is in the user's neighborhood.

To exploit the aforementioned contextual signals, we devise and meticulously analyse the behavior of a specialized recommendation model tailored to each signal. To combine the strengths of the various devised models, we further propose a hybrid contextual recommender for learning to rank upcoming events in an EBSN. Thorough experiments using

[1] `http://www.meetup.com`
[2] `http://blog.meetup.com/chart`
[3] RSVP stands for the French expression "répondez s'il vous plaît", meaning "please respond".

a large crawl of Meetup.com demonstrate the effectiveness of our proposed contextual learning approach in contrast to state-of-the-art event recommenders from the literature. To the best of our knowledge, this is the first attempt to conjointly exploit the aforementioned contextual evidence for the event recommendation problem. Our contributions can be summarized as follows:

- We compare and contrast several specialized event recommendation models based on multiple contextual signals available from EBSNs;
- We propose a hybrid recommendation approach that leverages multiple context-aware recommendation models as features for learning to rank events;
- We thoroughly evaluate our proposed approach on a large crawl of Meetup.com and demonstrate its effectiveness in contrast to state-of-the-art recommenders.

In the remainder of this paper, Section 2 formalizes our problem setting and the nomenclature used throughout the paper. Section 3 describes our proposed context-aware models for event recommendation. Section 4 presents the setup and the results of our experimental evaluation. Relevant related works are described in Section 5. Lastly, Section 6 provides our conclusions and directions for future work.

2. PROBLEM SETTING

Traditional recommender systems typically model an interaction between two types of entity, namely, users and items. Context-aware recommenders, on the other hand, exploit additional signals that help further define such an interaction, including the time, location, and social environment in which the interaction takes place [1]. As pointed out in Section 1, there is no interaction between users and candidate events in EBSNs, since the latter are always in the future.[4] Indeed, even when using positive RSVP data as a proxy for user-event attendance—as we do later in Section 3—this information is still missing for the majority of events (cf. Section 4). To overcome this problem, also known as the new item cold-start problem, we propose to use the rich contextual evidence available in EBSNs.

The problem that we address in this paper can be stated as follows: given a target user and a set of contextual signals, which of the available events are more likely to be attended (i.e. RSVP'ed "yes") by this user? We simulate this scenario by splitting the available RSVP data into two partitions, before and after a given time-stamp θ. The (user,events) pairs occurring before θ form the training partition, while those that occur after θ form the test partition. Formally, in addition to the set of users U and the set of events E, we consider as contextual signals users' temporal preferences T, the set of groups G that users can join, users' geographic distance preferences D, and the textual content of events C. In this paper, we only consider implicit feedback data, i.e., the set $S \subseteq U \times E \times T \times G \times D \times C \times D$ of relations between users, time, events, groups, distances and content. The task is then to find a scoring function of the form:

$$\hat{s} : U \times E \times T \times G \times C \times D \to \mathbb{R}, \tag{1}$$

that assigns a preference score for candidate events. Thus, for a given target user $u \in U$ and the contextual signals $t \in T$, $g \in G$, $c \in C$ and $d \in D$, the top-n recommendations can be computed by:

$$\text{top-}n(u, t, g, c, d) := \underset{e \in E \setminus E_u}{\arg\max^{n}}\, \hat{s}(u, e, t, g, c, d), \tag{2}$$

where n denotes the number of events to be recommended and E_u the set of events that user u has attended in the past. In top-n recommendation settings, such as this, the goal is to minimize a ranking loss function of the form:

$$\ell : \mathcal{P}(E) \times \mathbb{R}^E \to \mathbb{R},$$

which quantifies the misfit between the recommended list (generated by $\hat{s} \in \mathbb{R}^E$) and the actual list (a subset of $\mathcal{P}(E)$) on test users whose actual events are unknown during training. In the sequel we will use $\hat{s}_X(u, e)$ to denote a context-aware recommender that computes the relevance score of $e \in E$ to user $u \in U$ based on context X.

3. CONTEXTUAL MODELS

Data sparsity can severely hamper the performance of traditional event recommenders based on a single strategy. To overcome this limitation, we propose a hybrid recommendation approach that leverages multiple context-aware recommenders as features for learning to rank events. In the following, we describe each of the proposed features as well as our learning approach for event recommendation.

3.1 Social-Aware

In some EBSNs, such as Meetup.com, users may join groups and interact with other users both online, through the communication tools available in the platform, and face-to-face, by means of the events promoted by the groups. Here we propose two models based on the social interactions in these two modes of communication.

3.1.1 Group Frequency Model

As we will see in Section 4, group membership has a strong influence on the users' decision on attending an event, specially when the user is a frequent attendant of the group events. The intuition here is that the likelihood of the target user $u \in U$ attending event $e \in E$ depends on the number of events this user attended in the group that e belongs to. In other words, the more events a user attends in a group, the higher is the probability that this user will continue to attend events of this group. More formally, assuming that g_e represents the group associated with the candidate event e, the relevance score of this event for the target user $u \in U$ is calculated as follows:

$$\hat{s}_{S_1}(u, e) := \frac{|E_{u,g_e}|}{|E_u|}, \tag{3}$$

where E_u denotes the set of all events that the user $u \in U$ attended, E_{u,g_e} denotes the set of events attended by u that were created by group $g_e \in G$, and S_1 is the social context represented by this recommender.

3.1.2 Multi-Relational Model

Although simple and effective (cf. Section 4), the method described above does not consider two important interactions, namely, between users and all the groups they are affiliated to and between the groups and the events created by them. By using these two sources of interactions conjointly one might discover, for example, that users affiliated

[4]Except for the events that occur periodically.

to the same or similar groups are prone to attend the same events created by these groups. The user-group relation is especially important because it enables the recommendation of cold-start events, i.e., events with zero RSVP.

To model and leverage these sources of interactions conjointly we propose to use Multi-Relational Factorization with Bayesian Personalized Ranking (MRBPR) [8], a state-of-art recommender that solves the top-n recommendation task by optimizing a personalized ranking function (aka BPR [16]) considering multiple binary relations. In our case, the idea is to reconstruct the target relation $R_{UE} \subseteq U \times E$ by considering a joint factorization approach that includes the auxiliary relations $R_{UG} \subseteq U \times G$ and $R_{GE} \subseteq G \times E$.

The parameters of the model $\Theta := \{\mathbf{U}, \mathbf{E}, \mathbf{G}\}$ are the latent matrices associated with each considered entity. For example, the set of users U is associated with the matrix $\mathbf{U} \in \mathbb{R}^{|U| \times k}$, where k is the number of latent factors considered. The objective function is then expressed as follows:

$$\begin{aligned} \arg\min_{\Theta}\ & \alpha L(R_{UE}, \mathbf{U}\mathbf{E}^T) \\ & + \beta L(R_{UG}, \mathbf{U}\mathbf{G}^T) \\ & + \gamma L(R_{GE}, \mathbf{G}\mathbf{E}^T) \\ & + \lambda_U \|\mathbf{U}\|^2 + \lambda_E \|\mathbf{E}\|^2 + \lambda_G \|\mathbf{G}\|^2, \end{aligned} \tag{4}$$

where L is a loss function (in our case BPR [16]) measuring the reconstruction error of the relation under consideration, α, β, γ are application specific weights for the losses of the relations considered and $\lambda_U, \lambda_G, \lambda_E$ are regularization parameters. The intuition is that latent factors are shared between all relations that involve the same entities, i.e., information is propagated between relations sharing the same entities. The parameter learning is performed via stochastic gradient descent as proposed by [8].

Once the model is learned, predicting the relevance score of an event $e \in E$ for the target user $u \in U$ is done as follows:

$$\hat{s}_{S_2}(u, e) = \sum_{f=1}^{k} \vec{u}_f \vec{e}_f, \tag{5}$$

where $\vec{u}$ and $\vec{e}$ are the latent vectors of user u and event e respectively. The key difference from a standard matrix factorization (i.e. based on a single relation) is that now the latent factors of u and e also bear group aspects that may influence the users' decision on attending a future event.

3.2 Content-aware

Content-based filtering appears as a promising approach in EBSNs. In particular, besides its usual capacity of retrieving events with similar descriptions, it also captures recurrent events since these events tend to have similar or near-identical content. Thus, as content-based filtering approach, we use a classic bag-of-words model based on the textual description of events. Here, each user is represented as a TF-IDF vector of the words extracted from the past events the user attended. Additionally, each event is weighted by its time distance to the recommendation instant by means of a time value of money decay function [17]. More formally, the profile of the target user $u \in U$ is defined as follows:

$$\vec{u} := \sum_{e \in E_u} \frac{1}{(1+\alpha)^{\tau(e)}} \times \vec{e}, \tag{6}$$

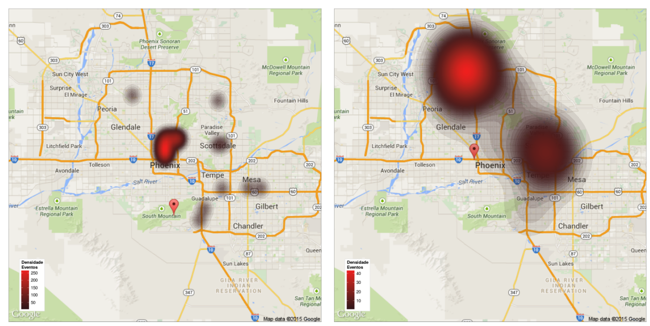
where $\vec{e}$ is the TF-IDF representation of event e considering the terms that appear in the event's description, α is a time decay factor and $\tau(e)$ returns the number of days from the RSVP to e until the moment of recommendation. In the end, the candidate events are ranked based on their cosine similarity with respect to the target user, such that:

$$\hat{s}_C(u, e) = \cos(\vec{u}, \vec{e}). \tag{7}$$

Other variations of this approach were also evaluated, such as Latent Dirichlent Allocation (LDA) [3], however, none of these variations outperformed the bag-of-words approach with time decay.

3.3 Location-aware

The mobility patterns of users within a city may vary from user to user. While someone may prefer to attend events in his neighborhood, others might like to attend events far away from their homes. Figure 3.3 depicts the estimated densities of two inhabitants of Phoenix, US, where the colored regions represent the density of attended events around their homes (depicted as map markers). Notice that while User 1 tends to concentrate his events in a single region with sporadic events in other regions, User 2 concentrates his events in two main regions.



(a) User 1 (b) User 2

Figure 1: Geographical densities of two users.

Here we propose a kernel-based density estimation approach to model the mobility patterns of individual users as distributions of geographic distances between the attended events. The relevance of a new event for a user is based on the aggregated likelihood of this event being located in any of the regions the user attended events in the past.

More formally, let L_u be the sample of lat-long coordinates of the events the target user has attended in the past, i.e.,

$$L_u := \bigcup_{e \in E_u} l_e, \tag{8}$$

where l_e is the lat-long coordinate of event $e \in E_u$.

Assuming that L_u comes from an unknown distribution f, a kernel density function $\hat{f}$ is defined over L_u as follows:

$$\hat{f}(l) := \frac{1}{|L_u|} \sum_{l' \in L_u} K_{\mathbf{H}}(l - l'), \tag{9}$$

where l is a lat-long coordinate, $K_H(.)$ is the bivariate Gaussian kernel,

$$K_{\mathbf{H}}(\mathbf{x}) := \frac{1}{\sqrt{2\pi|\mathbf{H}|}} \epsilon^{-\frac{\mathbf{x}\mathbf{x}^\top}{2\sqrt{\mathbf{H}}}}, \tag{10}$$

and $\mathbf{H}$ is a 2×2 symmetric and positive definite matrix that represents the bandwidth defined as $\mathbf{H} = (h_1, h_2) \times \mathbb{I}$.

The geographical preferences of a given user are then represented by the sum of all Gaussian distributions centered at l_e, for all $e \in E_u$. The candidate events are now scored based on their distances to the events attended by the target user in the past as follows:

$$\hat{s}_D(u, e) := \hat{f}(l_e). \quad (11)$$

3.4 Time-aware

Another important factor that might affect users' decision on attending an event is when the event occurs. Some users might prefer to attend events during the morning on weekends, while others prefer events occurring on Friday nights. We capture this intuition by assuming that users that attended events in the past at certain days of the week and at certain hours of the day will likely attend events with a similar temporal profile in the future. Figure 2 exposes two interesting cases. While User 1 tends to go to events every day, but mainly at night, User 2 attends events during the week and in the afternoon only.

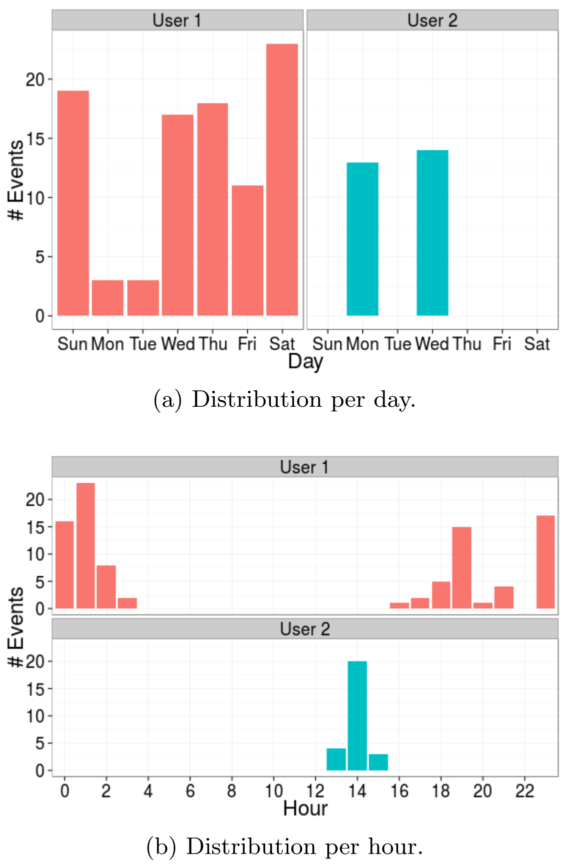


Figure 2: Temporal distribution of attended events.

To formalize this intuition, we represent each event e as a 24×7-dimensional vector $\vec{e}$ in the space of all possible days of the week and hours of the day, with a vector component set to 1 whenever the event happened at that particular day and hour. Accordingly, each user is represented as the centroid $\vec{u}$ of the events he attended in the past:

$$\vec{u} := \frac{1}{|E_u|} \sum_{e \in E_u} \vec{e}. \quad (12)$$

Similarly to Equation (6), the temporal score for user u and event e can now be expressed as the cosine between their vector representations, such that:

$$\hat{s}_T(u, e) := \cos(\vec{u}, \vec{e}). \quad (13)$$

3.5 Learning to Rank Events with Contextual Features

To further improve the effectiveness of event recommendation, particularly for cold-start events, we use the aforementioned recommenders as features for learning to rank events. In particular, let $\mathcal{D} := \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be the training set where $x_i := (\hat{s}_1(u, e), \ldots, \hat{s}_m(u, e), |U_e|)$ is a feature vector containing the normalized (i.e. z-score) scores generated for a given $(u \in U, e \in E)$ pair by each of the m context-aware recommenders plus the number of RSVPs $|U_e|$ the event e has received in the past. Notice that this last feature refers to the sparsity of this event. Also let $y_i = \{0, 1\}$ denote whether user u attended event e (class 1) or not (class 0) respectively. In a learning to rank setting, the classes are not nominal but ordinal (i.e. $0 < 1$). The goal is to learn a function $h(x)$ such that for any pair of training instances (x_i, y_i) and (x_j, y_j) the following implication holds:

$$h(x_i) > h(x_j) \Leftrightarrow y_i > y_j.$$

For learning $h(x)$, we have chosen Coordinate Ascent [12], a state-of-the-art listwise learning to rank approach. Coordinate Ascent directly optimizes a target ranking evaluation metric in an iterative fashion until convergence. As detailed in Section 4, in our experiments, we use NDCG@10 as the target optimization function. Once learned, we use function $h(x)$ to generate the final recommendation list. We will refer to this method as Multi-Contextual Learning to Rank Events (MCLRE for short).

4. EVALUATION

In this section, we thoroughly assess the effectiveness of our proposed context-aware learning approach for event recommendation. In Section 4.1, we describe the setup that supports our evaluation, including the EBSN dataset, the evaluation protocol and metrics, and the recommendation baselines used in our experiments. In Section 4.2, we discuss and analyse our experimental results.

4.1 Experimental Setup

4.1.1 Event Dataset

Meetup.com is an EBSN that promotes offline and face-to-face meetings, the so called Meetups. The network is formed by a Web-based structure of groups where like-minded people can plan, create, comment, share and advertise events. After the creation of a public event, any user can RSVP to it with "yes" or "no". We used the Meetup REST API[5] to crawl all public activity in the platform from January, 2010 to April, 2014, which comprises the dataset we used to evaluate the recommenders proposed in this paper.

Three cities located in the USA were selected for this investigation, namely Phoenix, Chicago and San Jose. We selected these cities for two main reasons: (i) they are among the most populous cities in the USA, which might indicate large event activity and (ii) the cities are located in different

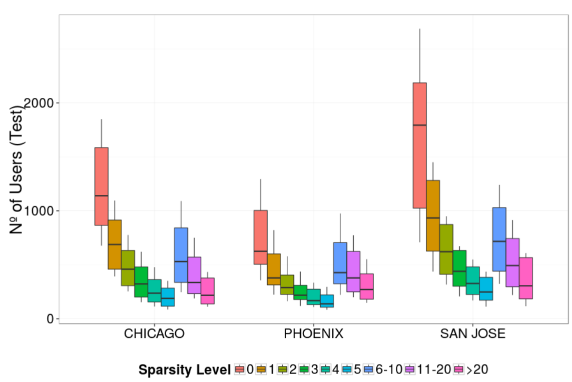
[5] `http://www.meetup.com/meetup_api/`

states, thus representing some degree of cultural diversity. The data collection procedure consisted of first collecting all the groups of a given city. Next, for each group, its members and events were collected. Finally, from each group member, we collected his or her event's RSVPs. Table 4.1.1 summarizes salient statistics of this dataset, in terms of the number of groups ($|G|$), users ($|U|$), events ($|E|$), and RSVPs. We also report the sparsity of the dataset, as the percentage of missing links (i.e., RSVPs) between users and events.

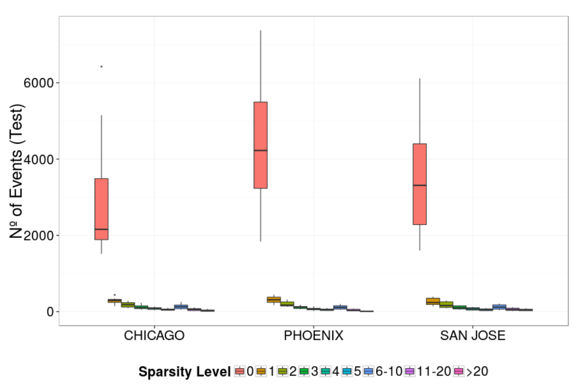
City	$\|G\|$	$\|U\|$	$\|E\|$	**RSVPs**	**Sparsity**
Chicago	2,321	207,649	190,927	1,375,154	99.996%
Phoenix	1,661	117,458	222,632	1,209,324	99.995%
San Jose	2,589	242,143	206,682	1,607,985	99.996%

Table 1: Statistics of the Meetup.com dataset.

The statistics in Table 4.1.1 emphasize the extreme sparsity of RSVPs in our dataset. In addition, Figure 4.1.1 depicts the user and event RSVP distribution per sparsity level. We show all the users (as well as events) in the test set that are associated with a specific number of "yes" RSVPs. In both cases, we see that the most frequent level of sparsity is zero, which means that most of the users and events in the test partition have no history of "yes" RSVPs. It is worth noticing that collaborative-filtering algorithms based solely on RSVP data will not work on these cases.



(a) Positive RSVPs per user.



(b) Positive RSVPs per event.

Figure 3: Distribution of positive RSVPs per sparsity level for (a) users and (b) events.

4.1.2 Evaluation Protocol

In order to simulate a realistic event recommendation scenario in our evaluation, we select 12 timestamps equally spaced in time across the 52 months covered by our dataset. By having multiple such timestamps, we can assess the effectiveness of each recommender through a sliding training window. In particular, each timestamp simulates a moment in time when an event recommendation must be generated. As illustrated in Figure 4, we consider as candidate events those created within the 6 months immediately *before* the target timestamp and that are scheduled to take place anytime *after* the timestamp. Accordingly, the RSVPs issued for a candidate event during the 6 months window preceding the target timestamp are used for training, whereas those issued after the timestamp form test instances to be recommended to the corresponding users. For each city, of the 12 train-test partitions induced by the chosen timestamps, the four initial partitions are used as validation sets for tuning the possible (hyper)parameters of the recommender models. The selected models are then tested by averaging their performance across the remaining eight partitions.

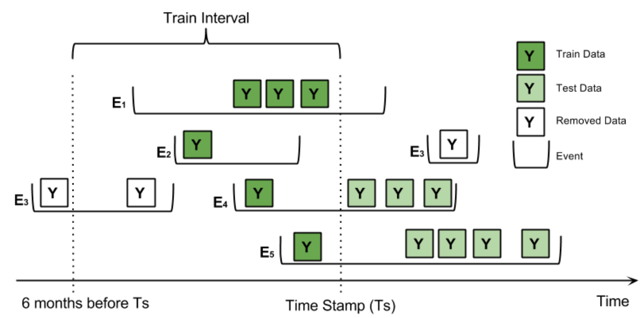


Figure 4: Example train-test partition.

Since the top-n item recommendation problem is usually related to personalized ranking (in our case, a ranked list of events to which a user is likely to RSVP "yes"), we use the well-known normalized discounted cumulative gain (NDCG) ranking evaluation metric truncated to the top 10 recommendations.

4.1.3 Recommendation Baselines

To assess the effectiveness of our MCLRE model, we compare it to the following state-of-the-art event recommenders:

- MP: A standard "most popular" baseline, which ranks events in descending order of the number of positive RSVPs that they received.
- BPR-MF: The Bayesian Personalized Ranking [16] is a state-of-the-art matrix factorization-based algorithm for top-n item recommendations.
- BPR-NET: This approach refers to the event recommender method proposed by [15]. The overall idea is to use two kinds of social networks as regularization terms of a BPR-MF model, namely, the social network based on the shared groups of users, and the social network inferred from the co-attended events.

The parameters used for all models were defined by a grid search on the validation sets. The following contextual model parameters were defined to Chicago, Phoenix and San Jose respectively: $\hat{s}_T$ (Equation (13)) considered 65, 50 and 100 neighbors; $\hat{s}_D$ (Equation (11)) used bandwidth parameters $h1$ and h_2 set to 0.001, 0.00075, 0.00075; and the time decay factor α of $\hat{s}_C$ (Equation (7)) was set to 0.005, 0.01 and 0.005. The following models had the same parameter values for all cities: the MRBPR (Equation (5)) and BPR-MF parameters were set to $k = 300$ (number of latent factors), learn rate 0.1 (step size of gradient descent), number of iterations 1500 (stop criterion); MRBPR relation weights were defined as 0.1 to R_{UE}, 0.22 to the R_{UG} and 0.68 to R_{GE}; BPR-NET was set with $k = 200$, learn rate 0.1 and number of iterations 600.

4.2 Results and Discussion

In this section, we assess the effectiveness of our proposed MCLRE model for event recommendation. In particular, we aim to answer the following research questions:

Q1. How effective is MCLRE for event recommendation?

Q2. How robust is MCLRE to sparsity in the RSVP data?

Q3. Which contextual features are effective recommenders?

In the remainder of this section, we address each of these research questions in turn.

4.2.1 Recommendation Effectiveness

To address Q1, we assess the effectiveness of MCLRE by comparing it to three state-of-the-art event recommenders from the literature, as discussed in Section 4.1.3: MP, BPR-MF, and BPR-NET. Figure 5 shows the result of this comparison in terms of NDCG@10 for events recommended in the three cities comprised in our dataset (Chicago, Phoenix, and San Jose). In particular, the distribution of NDCG@10 figures for the test users considered in each case is represented in a boxplot, highlighting the 25th, 50th (i.e., the median), and 75th percentiles inside the box, and possible outliers outside the box.

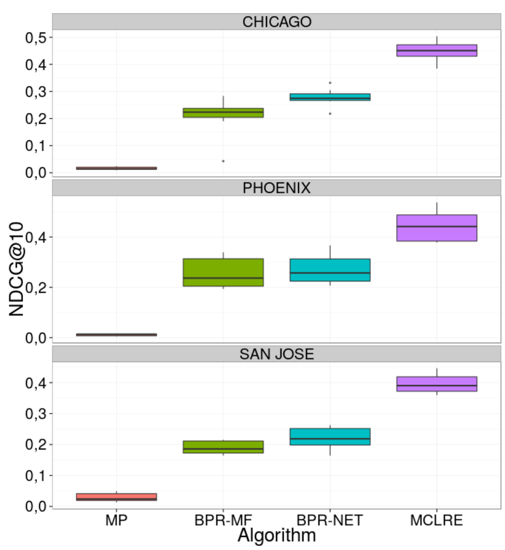


Figure 5: Event recommendation effectiveness.

From Figure 5, we first observe that MCLRE consistently outperforms all baselines in all three cities. Indeed, MCLRE improves upon the strongest baseline (BPR-NET) by 64% for Chicago, 72% for Phoenix, and 79% for San Jose in terms of the median NDCG@10. Recalling question Q1, this attests the effectiveness of our proposed model in contrast to state-of-the-art recommenders from the literature.

4.2.2 Robustness to Data Sparsity

One of the main reasons for the effective performance of our proposed model is that it is able to mitigate the cold-start problem for both users and events, which is intrinsic to EBSNs. To further assess the robustness of our model to data sparsity and hence address question Q2, we evaluate the effectiveness of MCLRE compared to the aforementioned baselines for various user and event sparsity levels. Figure 6 shows the result of this comparison in terms of NDCG@10 for event recommendations in Phoenix.[6] Each sparsity level on the x-axis denotes the number of RSVPs available for events (resp. users) at that sparsity level.

From Figure 6, we observe that MCLRE shows a higher resilience to data sparsity for both users and events compared to the other event recommenders. Even in the extreme case, when users and events have absolutely no past RSVP (i.e., the sparsity level 0), our model is able to provide recommendations with reasonable accuracy. BPR-NET is the only approach, besides ours, that is able to recommend for cold-start users, which is explained by its ability to exploit user-user interactions in terms of the shared groups these users are affiliated to. On the other hand, MCLRE is the only method able to recommend for cold-start events, mainly due to the content, geographic, and time-aware methods used as features. Notice that both BPR-NET and BPR-MF (and in fact any other recommendation algorithm that relies on user-event interaction) are not able to recommend cold-start events. Recalling question Q2, these results attest the robustness of our approach to sparse RSVP data.

4.2.3 Contextual Feature Analysis

To address Q3, we further analyse the effectiveness of our proposed approach in terms of its constituent contextual features. In particular, Figure 7 shows the NDCG@10 performance of our proposed context-aware recommenders compared to MCLRE, which combines them for learning to rank events. From the figure, we first note that MCLRE outperforms all individual recommenders. Recalling Q3, the social context appears as the most informative signal, with the multi-relational model (Equation (5)) achieving the best result followed by the simple but efficient group frequency heuristic (Equation (3)). The content-aware recommender (Equation (7)) also achieves good results, probably due to its ability to recommend cold-start events. The least informative contexts are the geographical and temporal ones (Equations (11) and (13), respectively), which is somewhat expected since they capture less specific aspects of the user preferences for events. Notice however that they are still better than the MP baseline.

5. RELATED WORKS

In this section, we summarize the most relevant related works on event recommendation. Minkov et al. [13] proposed a hybrid approach for recommending future events. Because no direct feedback exists for such events, users' pref-

[6]Results for Chicago and San Jose show similar trends and are omited due to space constraints.

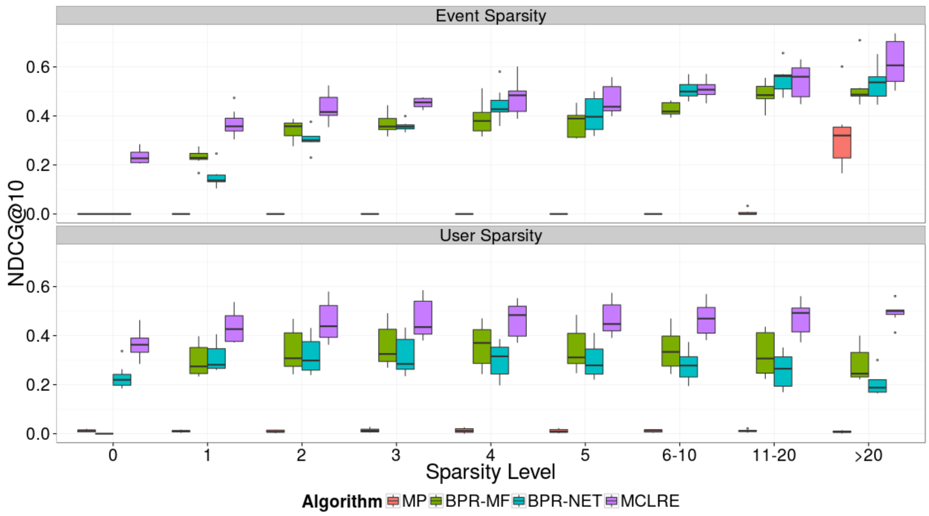


Figure 6: Recommendation effectiveness per sparsity level (Phoenix).

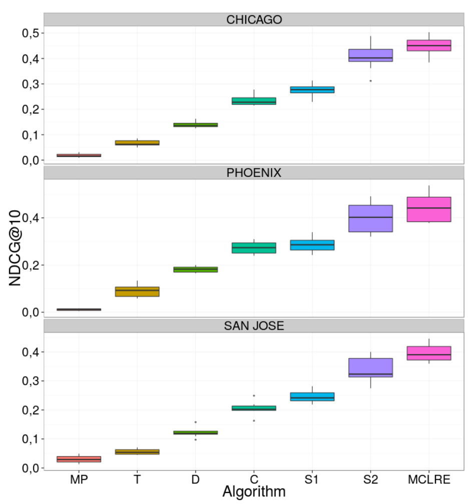


Figure 7: Per-feature recommendation effectiveness.

erences for an event were inferred based upon their preference for past events with similar content. In addition, they proposed to distinguish content dimensions shared collaboratively among users from dimensions unique to individual users. In a small user study in the domain of academic events, they demonstrated the effectiveness of the proposed hybrid approach in contrast to a pure content-based recommender. Additionally to using historical data of events attendance and the content of events, we exploit other contexts, such as the groups of users, location and temporal preferences. Moreover, we conduct a large scale experiment on data collected from a popular EBSN.

Another hybrid approach was proposed by Khrouf and Troncy [7] for recommending music-related events. In particular, their approach leveraged an enriched content-based representation of events by exploiting category information from DBPedia about the artists associated with each event. In addition, they modeled the cohesiveness of each user's content-based profile in order to avoid drifting from the user's core preferences. Lastly, this enriched content-based model was further combined with a collaborative model that exploited social interactions among users in an EBSN. Experiments on three music-oriented EBSNs demonstrated the effectiveness of the proposed extensions. Considering the multitude of domains found in general purpose EBSNs, such as Meetup.com, it may not be feasible to rely on DBPedia since many of these domains may not be covered. We do not make this assumption and leverage only the data directly available in the EBSN under consideration.

Liu et al. [9] proposed to exploit online and offline patterns of interaction in EBSNs for recommending new events. By exploiting only the topological structure of an EBSN, they proposed different information flow models to infer a user's interest for a given event based upon the manifested interest of other EBSN users for the same event. They demonstrated that community-based diffusion models, which favor the flow of information within (as opposed to across) communities, are particularly effective signals for event recommendation. This approach was later extended by Qiao et al. [15], who proposed a Bayesian matrix factorization approach to event recommendation [16], by employing social regularization factors inspired by users' interactions in an EBSN. Zhang et al. [19] showed that the proposed model improves upon ordinary and non-regularized matrix factorization models. Our work is complementary to these, by exploiting, additionally to RSVP and group information, other contextual data. Furthermore, Zhang et al. [19] compared their approach only against matrix factorization methods, while some recent works (see [11]) have shown that pure matrix factorization (based only on user-event interactions) performs poorly on EBSN data in comparison to simpler methods due to the high level of sparsity of these data sets.

There is a large body of research on venue recommendation in location-based social networks (LBSNs) [2, 4, 5, 6, 18, 10]. LBSNs share similar properties with EBSNs, e.g., location information of users and venues. However, LBSNs do not suffer from the new item cold-start problem as EBSNs do, since venues usually do not have an expiration date.

In a previous study [11], we performed a large-scale analysis of several factors that impact a user's propensity to reply positively to future events in an EBSN. In particular, based upon two years worth of user, event, and RSVP (i.e., user-event interactions) data from three representative US cities on Meetup.com, we found that users tend to provide RSVPs close to the occurrence of the events. We also found that state-of-the-art matrix factorization algorithms for top-n recommendation did not perform better than simpler collaborative filtering algorithms such as user-based k-NN. This motivated us to try other approaches that consider explicit features instead of latent ones.

6. CONCLUSIONS AND OUTLOOK

In this paper, we tackled the event recommendation problem through a learning to rank approach that takes several context-aware recommenders as input features. Besides content-based signals based on the events' description and collaborative signals derived from RSVPs, we exploited social, geographic, and temporal signals. We used Coordinate Ascent, a state-of-the-art learning to ranking algorithm that learns a personalized ranking from the output scores of the context-aware recommenders by optimizing NDCG@10 directly. To the best of our knowledge, this is the first attempt to leverage, collectively, the aforementioned contextual signals for the event recommendation problem.

In a thorough evaluation on a large crawl of Meetup.com, we showed that our approach improves upon a state-of-the-art context-aware event recommender based on matrix factorization with social regularization by up to 79%. From our results, we can draw many interesting conclusions, such as:

- The use of multiple contexts pays off and can both lead to highly accurate recommendations and mitigate the new user and new item cold-start problem on EBSNs;
- In Meetup.com, events created by groups of which a user is a member are far more relevant than the content of the events or collaborative RSVP data.

We have several ideas to extend this work in the future. First, we will repeat the experiments with cities in different countries. Second, we will exploit other contexts and approaches to leverage these contexts. We also intend to exploit other kinds of EBSNs such as, e.g., Facebook events.

7. ACKNOWLEDGMENTS

This work is partially supported by the National Institute of Science and Technology for Software Engineering (INES), funded by CNPq and FACEPE, grants 573964/2008-4 and APQ-1037-1.03/08; and Hewlett-Packard Brasil Ltda., through the FRH-Analytics 2013 project, and used incentives from the Brazilian Informatics Law (n. 8.2.48/1991). We also want to thank Lucas Drumond for generously sharing the implementation of his MRBPR method.

8. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Context-aware recommender systems. In *Proc. of RecSys*, pages 335–336, 2008.

[2] J. Bao, Y. Zheng, and M. F. Mokbel. Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proc. of SIGSPATIAL/GIS*, pages 199–208, 2012.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[4] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *Proc. of AAAI*, pages 17–23, 2012.

[5] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *Proc. of RecSys*, pages 93–100, 2013.

[6] B. Hu and M. Ester. Spatial topic modeling in online social media for location recommendation. In *Proc. of RecSys*, pages 25–32, 2013.

[7] H. Khrouf and R. Troncy. Hybrid event recommendation using linked data and user diversity. In *Proc. of RecSys*, pages 185–192, 2013.

[8] A. Krohn-Grimberghe, L. Drumond, C. Freudenthaler, and L. Schmidt-Thieme. Multi-relational matrix factorization using bayesian personalized ranking for social network data. In *Proc. of WSDM*, pages 173–182, 2012.

[9] X. Liu, Q. He, Y. Tian, W.-C. Lee, J. McPherson, and J. Han. Event-based social networks: linking the online and offline social worlds. In *Proc. of SIGKDD*, pages 1032–1040, 2012.

[10] X. Liu, Y. Liu, K. Aberer, and C. Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *Proc. of CIKM*, pages 733–738, 2013.

[11] A. Q. Macedo and L. B. Marinho. Event recommendation in event-based social networks. In *Proc. of Int. Work. on Social Personalization*, 2014.

[12] D. Metzler and W. Bruce Croft. Linear feature-based models for information retrieval. *Inf. Retr.*, 10(3):257–274, 2007.

[13] E. Minkov, B. Charrow, J. Ledlie, S. Teller, and T. Jaakkola. Collaborative future event recommendation. In *Proc. of CIKM*, pages 819–828, 2010.

[14] S.-T. Park and W. Chu. Pairwise preference regression for cold-start recommendation. In *Proc. of RecSys*, pages 21–28, 2009.

[15] Z. Qiao, P. Zhang, C. Zhou, Y. Cao, L. Guo, and Y. Zhang. Event recommendation in event-based social networks. In *Proc. of AAAI*, pages 3130–3131, 2014.

[16] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Proc. of UAI*, pages 452–461, 2009.

[17] T. Sandholm and H. Ung. Real-time, location-aware collaborative filtering of web content. In *Proc. of CaRR*, pages 14–18, 2011.

[18] M. Ye, P. Yin, W.-C. Lee, and D. L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proc. of SIGIR*, pages 325–334, 2011.

[19] W. Zhang, J. Wang, and W. Feng. Combining latent factor model with location features for event-based group recommendation. In *Proc. of SIGKDD*, pages 910–918, 2013.