Multimedia Tools and Applications (2021) 80:16599–16618
https://doi.org/10.1007/s11042-020-08884-9

An effective content-based event recommendation model

Thanh Trinh[1] · Dingming Wu[1] · Ruili Wang[2] · Joshua Zhexue Huang[1]

Received: 3 July 2019 / Revised: 8 January 2020 / Accepted: 27 March 2020 /
Published online: 21 April 2020

Abstract
Event-based social networks (EBSNs) facilitate people to interact with each other by sharing similar interests in online groups or taking part in offline events together. Event recommendation in EBSNs has been studied by many researchers. However, the problem of recommending the event to the top N active-friends of the key user has rarely been studied in EBSNs. In this paper, we propose a new method to solve this problem. In this method, we first construct an association matrix from the content of events and user features. Then, we define a new content-based event recommendation model, which combines the matrix, spatio-temporal relations and user interests to recommend an event to the active-friends of a key user. A series of experiments were conducted on real datasets collected from Meetup, and the comparison results have demonstrated the effectiveness of the new model.

Keywords EBSNs · Social networks · Topic model · Recommendation

1 Introduction

Event-based social networks (EBSNs) have become popular in the last couple of years, e.g., Meetup[1] and Douban[2], which provide platforms for those who interact with each other in

[1] www.meetup.com
[2] www.douban.com

✉ Dingming Wu
dingming@szu.edu.cn

Thanh Trinh
tthanh@szu.edu.cn

Ruili Wang
Ruili.wang@massey.ac.nz

Joshua Zhexue Huang
zx.huang@szu.edu.cn

[1] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China

[2] School of Natural and Computational Sciences, Massey University, Auckland, New Zealand

both online and offline activities, or who can create a user group with similar interests for professional networking. In March of 2019, Meetup, one of the largest EBSNs, had over 40 million users; more than 10 thousand events were created every day in more than 180 countries through the Meetup network.

Liu et al. [22] studied different types of recommendation problems in EBSNs, such as event recommendations and group recommendations. These problems have gained a significant increase in research interests [1, 4, 14–16, 21, 22, 24, 25, 29–32, 36–39]. Several techniques were proposed to solve the recommendation problems, such as matrix factorization [30, 31, 38] and exploiting content information [10, 14, 16, 23–25]. The similarities between users and between events were also studied in [9, 35], in which user interests and event contents were chosen to compute the similarities. Besides, the problem of friend recommendation in location-based social networks has been studied in [2, 6, 13, 33], but this problem has been ignored in EBSNs.

Understanding users' behavior and predicting activities of them are important to design effective recommendation systems. Whether a user takes part in an event or not depends mainly on the topic of the event, and his/her own interests or the interests of his/her active friends. We define that active-friends of a user are the people who are interested in the topic of an event that the user is interested in and the people who are likely to attend the event based on their previous attending histories. In this paper, we study the problem of event recommendation for active-friends in the context of EBSNs. We formulate this problem as follows: *Given an event e^f and a key user u^*, the goal is to recommend event e^f to the top N active-friends of user u^* based on a list of previous events.*

To tackle the problem, we crawled datasets of events in a time window and of users who attended these events from the Meetup network, and conducted data analysis to investigate major factors affecting the decisions of users to take part in the events. From our analysis, we propose a content-based model that combines user features and events contents to recommend an upcoming event to the friends of a key user.

Our proposed model is constructed with the following procedure: Firstly, we propose a new approach to calculating the similarity of events based on their contents and locations. In this approach, we model the contents of events as the Latent Dirichlet Allocation (LDA) topic model [3] and use Gibbs Sampling method [7] to generate topic distributions of the events. Then, we use the Jensen Shannon divergence [20] to compute the similarity between the topic distributions of two events. Secondly, we collect users who took part in the chosen events which are similar to the given event e^f, and propose a new content-based method to compute possibility scores of these users who are closely related to the key user u^*. In this method, we use multiple factors to compute the possibility scores, including the similarity between previously chosen events and the event e^f, locations and users' interests, and event times. Finally, we have a list of users with their possibility scores with respect to the key user and the event e^f. The event e^f will be recommended to the top N active-friends of the key user u^* who have the top possibility scores.

Experiments were conducted on real world datasets crawled from four cities. The experimental results have shown that the performance of our content-based event recommendation model outperformed three baseline methods.

The remainder of this paper is organized as follows. Section 2 reviews the related work. EBSNs data characteristics are analyzed, and the proposed model is presented in Section 3. Section 4 presents the Gibbs Sampling method to compute the topic vectors of events. Experiment results are presented in Section 5. Section 6 gives conclusions and new research directions of this work.

2 Related work

Event-based social networks (EBSNs) were initially studied by Liu et al. [22] and some unique and exciting features of EBSNs were discovered, such as locality structural networks and information flows. Different types of recommendation problems for EBSNs were also defined, such as event recommendation and group recommendation problems. These problems have been studied by many researchers [4, 9, 11, 14–16, 22, 24, 25, 29–31, 37–40].

Matrix factorization methods are famous for rating recommender systems [34]. Qiao et al [31] combined online and offline social relationships between users with locations to propose a standard matrix factorization. In another work, Qiao et al. [30] proposed a Bayesian matrix factorization model based on social interaction features and users rating data. However, these two methods ignored the content information of events.

To exploit the content information, several works studied the content-based event recommendation. Zhang and Wang [37] considered event contents, user relations, locations and organizers, and proposed a collective Bayesian Poisson factorization method to solve the cold-start local event recommendation problem. Based on analyzing users' interests and attended seminar events, Minkov et al. [25] proposed a hybrid content-based and collaborative method to recommend academic events to users. The problem of event detection was studied in [14, 17] on the Twitter network. The LDA topic model was adopted in [14, 39] to solve the event recommendation problem. Li et al. [19] analyzed the rich information of DoubanEvent network, and proposed a hybrid model with integration of the LDA topic model to predict social influence. In a series of recent works [26–28], Ogundele et al. studied the problem of personalized event recommendation in EBSNs, and proposed a framework, called SoCaST. They listed the four types of influences on users, i.e., temporal, geographical, categorical and social types, which were modeled in SoCaST. The multi-criteria decision making technique was adopted to make recommendations of the most suitable events to users.

Lu et al. [23] developed a cross-media event extraction system which effectively recommends events to users' ongoing search based on their interests. Jhamb and Fang [16] analyzed rich information of groups and presented two oriented items, user-oriented and event-oriented, to solve the event recommendation problem. They proposed a dual-perspective latent model that uses Logistic and Probit sigmoid functions to model the two oriented items. Another work in Macedo et al. [24] exploited a set of contextual information including description of events and interaction of users derived from their RSVPs. Then, they further proposed a hybrid context-aware model to recommend upcoming events to users in EBSNs. Chen and Su [4] also exploited the social interaction of users and collaborative friendships, and proposed a new social event recommendation model. Horowitz et al. [15] modeled the agenda of users and tags to form a mobile recommender system for events. To investigate how relevant human personality is to different types of places people visit, Chorley et al. [5] studied human behaviors on the Foursquare location-based social network. The conclusion was that the personality traits determine the sorts of places that users checked in.

Developing a unified model that can solve multiple recommendation tasks in one framework was also carried in [9] and [29]. Dong et al. [9] designed a probabilistic model that solves the group recommendation and the event recommendation problems in one framework. A time horizon was used in the model to recommend groups to users and offline events to users iteratively. A general graph-based model [29] was proposed to address three recommendation problems, namely, group-to-user, tag-to-user and event-to-user recommendations in one framework. The model treats the three problems as a query-dependent node proximity problem with rich information included. Extraction of features from users interests and events contents was studied in [1, 21, 36]. Zhang et al. [38] studied a new type of group recommendation problems, called the personalized event-based group recommendation. Location and social features were first modeled in matrix factorization, and the information of users and groups is taken into account. They proposed a pairwise tag-enhanced and feature-based matrix factorization method that is a combination of all the elements in order to solve the group recommendation problem. Du et al. [10] and Ding et al. [8] modeled the factors of spatial and temporal contexts of events to predict event attendance. Xu et al. [35] studied the problem of event recommendation through the Douban network, and proposed a semantic-enhanced and context-aware model that was constructed from a combination of the content information analysis and the social relations.

Hannon et al. [13] studied the tweets and relationships of users from Twitter and defined the friend recommendation problem, i.e., recommending those who should be followed by new users of the network. They proposed a model that combined collaborative filtering strategies with the contents to solve the problem. Several works [6, 18, 33] studied further extensions of friend recommendation. Chu et al. [6] proposed a new friend recommendation problem for location-based social networks. Based on the similar interests of the users, they employed the real-life location and dwell time in friend recommendation. After gathering these data, the proposed method analyzes the data using a weighted Voronoi diagram and interest similarity. Tu et al. [33] proposed an approach to solving the problem with a specific location and a friend in the location-based social networks. Li et al. [18] collected social and event-based features from DoubanEvent data to study the problem of followee recommendation in EBSNs, and they proposed an improved Bayesian model that uses these features to address this problem. However, in our work, we use data crawled from Meetup network and there is no direct links between users.

3 Methodology

In this section, we first describe the datasets of four cities we crawled from Meetup network. Then, we analyze the properties of the data, and investigate what features affect the decisions of users to take part in events. Finally, we propose a new content-based event recommendation model.

3.1 Data collection

We crawled datasets from Meetup.com that is considered as the largest event-based social network (EBSN). The structure of EBSNs is illustrated in Fig. 1, which contains five main entities: groups, events, users, venues and tags. One event is created from a specific group and hosted in a particular venue. Users confirm whether to attend the event or not by sending RSVPs (YES or NO). YES means that a user will attend an event. NO means that a user will not take part in an event. Each user can be a member of several groups and send RSVPs for any events. Users can express their personal interests by a set of tags. In our work, each event has content, location, time and a list of RSVPs (YES or NO). Each user has information of a set of tags and a location.

For example, in Fig. 1 User 1 and User 2 join Group 1, and User 2 and User 3 join Group 3. Group 1 held the Live Music and Reading Books events in a Coffee shop, which were attended by User 1 and User 2. Both User 3 and User 2 took part in the Cycling event in

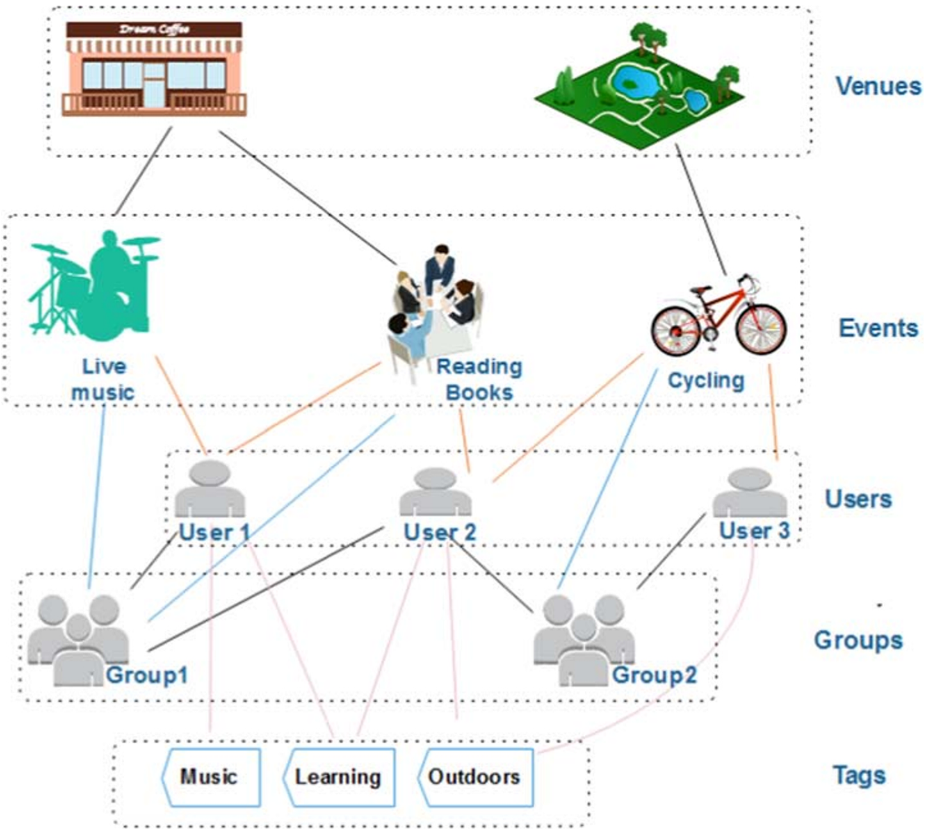


Fig. 1 An example of an event-based social network

a Park venue, which was held by Group 2. User 1 uses tags "Music and Learning". Tags "Learning and Outdoors" are used by User 2, and User 3 only uses tag "Outdoors".

3.2 User similarity

In EBSNs, there is no direct link to indicate that two users are friends. However, we model the similarities between users in two ways, i.e., interest-based and behavior-based similarities [9, 35]. We use these similarities to measure how users are related to each other.

3.2.1 Interest-based similarity

Users who are members of a particular group may have similar personal interests, which can be manifested by a set of tags (as shown in Fig. 1). To calculate the interest-based similarity between users, we utilize the tags. The interest-based similarity between users is defined:

$$S_i(u, u^*) = \frac{T^u \cap T^{u^*}}{T^u \cup T^{u^*}} \tag{1}$$

where T^u and T^{u^*} are two sets of tags of users u and u^* respectively.

3.2.2 Behavior-based similarity

The event participation is treated as a social behavior of users. The behavior-based similarity between two users can be defined as

$$S_b^c(u, u^*) = \frac{E^u \cdot E^{u^*}}{\| E^u \| \| E^{u^*} \|} \tag{2}$$

where E^u and E^{u*} are two sets of events that u and u^* are interested and have sent RSVPs to these events.

To be more specific to the upcoming event e^f, we define the behavior-based similarity between two users as

$$S_b(u, u^*) = \sum_{j=1}^{|E|} D_u^j \times D_{u^*}^j \tag{3}$$

where D^j is the similarity between the upcoming event e^f and e^j in E which the user attended. The similarity between events is defined in (5) in the following section.

3.3 Event similarity

Social interests of users are expressed by social events they attend in EBSNs. The social events are described in a set of documents shown on web pages as semi-structured text data, as illustrated in Fig. 2. We can observe that the event consists of the content (i.e., the title and description) and its location and time. These kinds of information have an impact on the decisions of users to confirm RSVPs (YES or NO). Hence, the content and location factors extracted from events are adopted to analyze the decisions of users.

3.3.1 Topics from events contents

The contents of events may include several themes or topics. Instead of using keywords, we represent an event as a vector of topics based on the Latent Dirichlet Allocation (LDA) topic model [3]. Given a set of event documents, we use the Gibbs Sampling method [7] to compute the topic vectors of these documents. Then, the set of event documents is converted to a matrix Θ where a row is a corresponding event document and a column is a topic. The process of computing Θ is described in details in Section 4.

3.3.2 Similarity between events

To measure how events are related to each other, the divergence between the topic distributions of the events is taken into account. We adopt Jensen-Shannon divergence [20] to compute the similarity between events.

Given two topic distributions θ_p and θ_q of events e_p and e_q respectively, Jensen Shannon divergence D_{JS} is defined as follows:

$$D_{JS}(\theta_p, \theta_q) = \frac{1}{2}\left[D_{KL}\left(\theta_p, \frac{\theta p + \theta q}{2}\right) + D_{KL}\left(\theta q, \frac{\theta p + \theta q}{2}\right)\right] \tag{4}$$

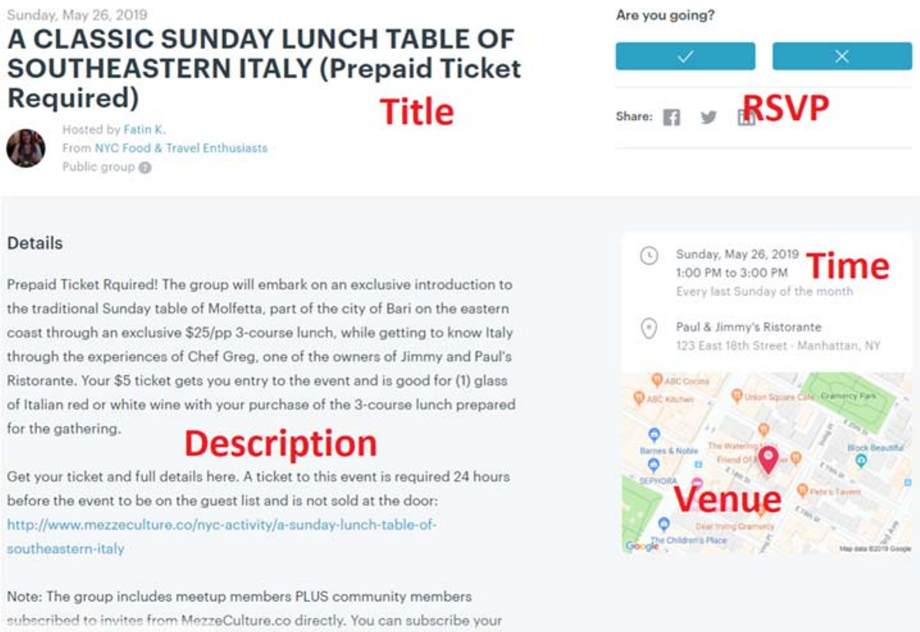


Fig. 2 Example of one Meetup event

where D_{JS} is the Jensen Shannon divergence, D_{KL} is the Kullback-Leibler divergence, defined as $D_{KL}(\theta p, \theta q) = \Sigma_{i=1}^{K} p_i ln \frac{p_i}{q_i}$ where p_i and q_i are the probabilities of topic i in event e_p and event e_q respectively.

The similarity of e_p and e_q is denoted by D, computed as

$$D = 1 - D_{JS}(\theta_p, \theta_q) \tag{5}$$

The value of D varies from 0 to 1. In particular, if the value is close to 1, the contents between e_p and e_q are very similar. Otherwise, if the contents are very different, the value is close to 0.

3.4 Location-based similarity between users and events

Obviously, users who are interested in previous events are willing to attend the upcoming event, if the new event is located closely to the previous event places and the users' location. The location-based similarity can be used to measure how users are related to events in terms of location. It is represented as

$$S_l(e, u) = e^{\frac{-d(e,u)^2}{2}} \tag{6}$$

where $d(.,.)$ is the Euclidean distance between location of event e and user u.

3.5 Event recommendation model

Given an upcoming event e^f and one key user u^*, we want to find a group of potential attendees who are closely related to u^*. We call them active-friends of u^* and recommend the upcoming event e^f to them. We name this recommendation problem as the problem of event recommendation for active-friends.

We propose an effective content-based event recommendation model (**ECA**) that is formed by event-based and user-based similarities to solve the problem. Our model consists of two phases. The first phase selects previous events in the same group or the same location with e^f in order to reduce the number of events and sparse data. Then, we compute the similarity between the selected previous events and e^f. In the second phase, we first collect users from the selected events, and then construct an association matrix of users and events. Consequently, we score all users and recommend e^f to top N active-friends of user u^* .

3.5.1 Phase 1: Find relevant events

This phase is used to select previous events relevant to the upcoming event e^f. The process is described as follows:

- Step 1 The set of events E^P is modeled by the Latent Dirichlet Allocation (LDA) topic model, and Gibbs sampling method is used to generate topic distributions Θ^P and topic-word distributions Φ^P from the whole set of events contents.
- Step 2 Given an upcoming event e^f and a key user u^*, the content of e^f and Φ^P are used to infer topic distribution θ^f of the event e^f.
- Step 3 An event e^i in the whole event set E^P is chosen and added in the set E^C, if it meets one of the following requirements: (1) e^i and e^f are created by the same group or (2) they have the same location.
- Step 4 For each event e^j in E^C, the similarity D^j between e^j and e^f is computed by (5).

3.5.2 Phase 2: Find active-friends

From Phase 1, we obtain a set of chosen events E^C. Then, we obtain a set of users U who attended these events in E^C. If a user u^i attended an event e^j, the similarity between event e^j and the upcoming event e^f is computed in (5), denoted by D^j. This similarity D^j represents the association between the user u^i and the event e^j. The results are represented in a user-event association matrix (UE). In UE, each row is a user, and each column is an event. The elements equal to 0 mean that the users did not attended the corresponding events. An example of UE is given below.

$$\mathrm{UE} = \begin{array}{c} \\ u^1 \\ u^2 \\ ... \\ u^* \\ u^i \\ u^n \end{array} \begin{array}{c} \begin{array}{ccccc} e^1 & e^2 & ... & e^j & e^m \end{array} \\ \begin{pmatrix} D^1 & 0 & 0 & 0 & D^m \\ 0 & D^2 & 0 & D^j & 0 \\ 0 & D^2 & D^{\cdots} & 0 & D^m \\ D^1 & 0 & D^{\cdots} & 0 & D^m \\ D^1 & 0 & D^{\cdots} & 0 & 0 \\ 0 & D^2 & 0 & D^j & 0 \end{pmatrix} \end{array}$$

For the key user u^*, we select one event organizer of the upcoming event as u^*, and this user has also attended previous events. Given a UE, we can compute the similarity between the key user u^* and all other users in UE, using (3).

To measure how the upcoming event e^f is associated with the last attended event e^l of a user u, we calculate the gap time T_u between them as

$$T_u = \frac{1}{T^{e^f} - T^{e^l}} \tag{7}$$

where the time difference of (T^{e^f} - T^{e^l}) is measured in a number of weeks. The bigger the gap time T_u is, the more associated the event e^f to the user u is.

Finally, each user in UE is scored against the key user u^* with respect to the upcoming event e^f as

$$S(u|(u^*, e^f) = S_b(u, u^*) + S_i(u, u^*) + T_u \tag{8}$$

where $S_i(u, u^*)$ and $S_b(u, u^*)$ are computed by (1) and (3) respectively.

Top N users with high scores are selected and the upcoming event e^f is recommended to them. If there are some users having the same scores, these users will be chosen by distance scores that are computed by (6). The user with high distance score is chosen.

4 Gibbs sampling method for computing topic vectors

We use the LDA topic model to model a set of event documents crawled from EBSNs. Given the number of latent topics K, the generative process of LDA for a set of N event documents with a set of unique words W is described in the following steps:

1. For each topic $z \in K$
   (a) Draw $\phi_z \sim Dirichlet(\beta)$, which is a multinomial distribution over words.
2. For each document $d \in N$
   (a) Draw a topic distribution $\theta_d \sim Dirichlet(\alpha)$.
   (b) For each w_i in d
      (i) draw a topic $z_i \sim Mult(\theta_d)$
      (ii) draw a word $w_i \sim Mult(\phi_{z_i})$

Suppose that parameters α and β are given, the generative process results in the following joint distribution for document d:

$$p(w_d, z_d, \theta_d, \Phi|\alpha, \beta) = \Pi_{i=1}^{N_d} p(w_{d,i}|\phi_{z_{d,i}}) p(z_{d,i}|\theta_d) \\ p(\theta_d|\alpha) p(\Phi|\beta) \tag{9}$$

where N_d is the number of words in document d.

Probability distributions for the set of N documents is

$$p(W, Z, \Theta, \Phi|\alpha, \beta) = \Pi_{d=1}^{N} p(w_d, z_d, \theta_d, \Phi|\alpha, \beta) \tag{10}$$

To represent the set of N documents as a set of N vectors, each vector with K topics, we need to calculate the topic distributions Θ and the topic-word distributions Φ. In this work, we use Gibbs Sampling method to calculate Θ and Φ. The pseudo code of this method is briefly described in Algorithm 1, which includes the following main three steps.

Step 1 **Initial step**: For each document d_j in the set of N documents, randomly assign each word w_i to one of K topics.

Step 2 **Gibbs Sampling**: For each iteration, we re-assign a topic to each w_i in d_j. First we need to reduce the value of W_{t^c}, N_{d_j,t^c} and W_{w_i,t^c} related to the current topic t^c by 1. We select topic k in K with the probability for word w_i denoted by:

$$P(z_t = k|z_{-t}, w_i, d_j) = \frac{W_{w_i,k} + \beta}{W_k + W\beta} \times \frac{N_{d_j,k} + \alpha}{N_{d_j} + K\alpha} \tag{11}$$

where $P(z_t = k|.)$ is the probability of word w_i assigned to topic k.

z_{-t} includes all topic allocations except for z_t.

$W_{w_i,k}$: Number of times word w_i assigned for topic k

W_k: Number of words in W assigned to topic k

N_{d_j}: Total number of words in document d_j

$N_{d_j,k}$: Number of words for topic k in document d_j

We have a set of $P(z|.)$. Then, a new topic t^n is sampled from $P(z|.)$ and reassigned to w_i. We increase the value of W_{t^n}, N_{d_j,t^n} and W_{w_i,t^n} by 1.

Step 3 **Generating Φ and Θ**: After a large number of iterations (e.g, 1000 loops), we obtain the stable topic-word distribution Φ and the stable topic distribution Θ as follows:

$$\phi_{w_i,k} = \frac{W_{w_i,k} + \beta}{W_k + W\beta} \tag{12}$$

where $\phi_{w_i,k}$ is the probability of word w_i for topic k, and

$$\theta_{d_j,k} = \frac{N_{d_j,k} + \alpha}{N_{d_j} + K\alpha} \tag{13}$$

where $\theta_{d_j,k}$ is the proportion of topic k in document d_j. Topic distribution of document θ_{d_j} is a set of $\theta_{d_j,k}$.

Algorithm 1 Gibbs sampling for LDA.

Input: a set of documents N and a set of words W
Output: Topic distribution Θ and Topic-word distribution Φ
Notations:
$T(d, w)$: Topic of word w in document d
$P(z_t = k|.)$: The probability of word w_i assigned to topic k.
z_{-t}: All topic allocations except for z_t.
$W_{w_i,k}$: Number of times word w_i assigned for topic k
W_k: Number of words in W assigned to topic k
N_{d_j}: Number of total words in document d_j
$N_{d_j,k}$: Number of words from topic k in document d_j
Iteration: number of times for sampling.
for d_j *in* N **do**
  for w_i *in* d_j **do**
    t ← sample from (K)
    $T(d_j, w_i) \leftarrow$ t
    $W_t += 1$; $N_{d_j,t} += 1$; $W_{w_i,t} += 1$
  end
end
for *iter in Iteration* **do**
  for d_j *in* N **do**
    for *word* w_i *in* d_j **do**
      $t^c \leftarrow T(d_j, w_i)$; the current topic of word w_i.
      W_{t^c} -=1; $N_{d_j,t^c} -= 1$; $W_{w_i,t^c} -= 1$
      for z_t *in* K **do**
        p($z_t = k|.) = \frac{W_{w_i,k}+\beta}{W_k+W\beta} \times \frac{N_{d_jk}+\alpha}{N_{d_j}+K\alpha}$
      end
      $t^n \leftarrow$ sample from $p(z|.)$; we have a new topic t^n
      $T(d_j, w_i) \leftarrow t^n$; reassigning a new topic t to word w_i
      $W_{t^n} += 1$; $N_{d_j,t^n} += 1$; $W_{w_i,t^n} += 1$
    end
  end
end
We obtain topic distribution Θ for the set of N documents and topic-word distribution Φ for the set of words W.
$\phi_{w_i,k} = \frac{W_{w_i,k}+\beta}{W_k+W\beta}$; $\theta_{d_j,k} = \frac{N_{d_j,k}+\alpha}{N_{d_j}+K\alpha}$;

5 Experiments

In this section, we present the experiment results to show the performance of the new recommendation model for active-friends. First, we describe the datasets. Then we evaluate the proposed model compared to other baseline methods.

5.1 Experimental setup

5.1.1 Datasets

The data used in our experiments were crawled from the Meetup website[3] in year 2016. Four cities in the world were selected, i.e., London(**LD**), Sydney(**SN**), and San Francisco (**SF**) and New York (**NY**). For each city, we collected information of events and attendees of the events. The information of each event includes title, description, time and location. Each event has a list of RSVPs (YES or NO), which present those who confirm YES or NO to take part in it. The number of users was obtained as the number of unique users. Moreover, each event must have at least 5 attendees (YESs). A list of attendees in the events is obtained. Each attendee contains the information of interests, location, and time to send RSVPs to the event.

We plot the distributions of events and users for four cities, illustrated in Fig. 3. We can see that 95% of users attended less than 15 events and 70% of events had less than 20 attendees. These datasets are summarized in Table 1. In our experiments, we consider the chronological order of events in selecting training and testing data. We first sorted the events on event time. Then, we took the early 80% events as the training data, the later 20% events as new events for the testing data for each city.

5.1.2 Evaluation metrics

To evaluate the performance of the active-friends recommendation we adopt two classic precision and recall metrics, denoted by $Precision@N$ and $Recall@N$ respectively.

$$Precision@N = \frac{\frac{|L^{rec} \bigcap L^{real}|}{N}}{M} \tag{14}$$

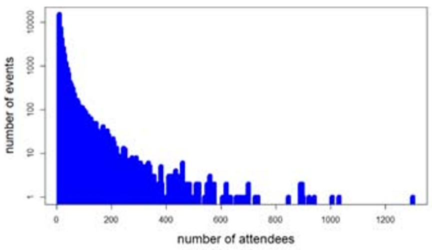
$$Recall@N = \frac{\frac{|L^{rec} \bigcap L^{real}|}{|L^{real}|}}{M} \tag{15}$$

where N is the number of friends of a user u, M is the number of upcoming events in the testing set. L^{rec} is the list of active-friends to whom the given event e will be recommended, and L^{real} is the list of real attendees who say YES to the event e.

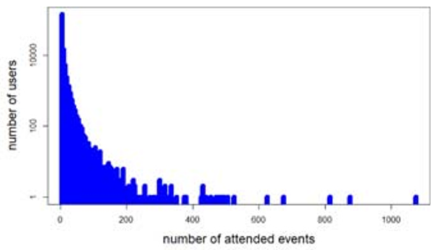
5.1.3 Parameters setting

For each dataset, we removed all punctuations and stop words from the contents of all events. Then, the cleaned contents in the training set were used as a collection of all documents P with a set of unique words W. In order to obtain the similarity between events [12], we set parameters $\beta = 0.01$, $\alpha = 50/K$, where K was set to 100 topics. The iteration of Gibbs Sampling method was set to 1000. These parameters were used to generate topic distributions Θ^P and topic-word distributions Φ^P.

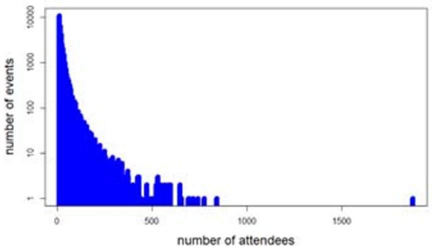
[3] www.meetup.com/meetup_api/



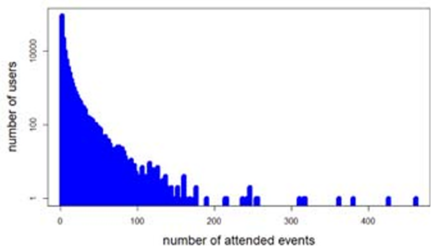
(a) Event distribution - New York



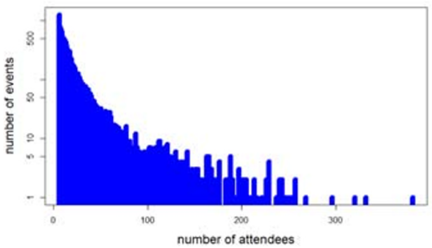
(b) User distribution - New York



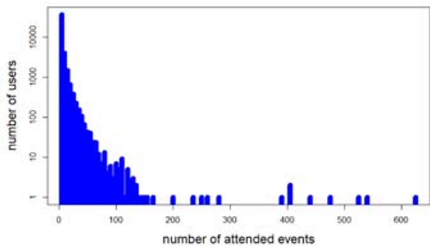
(c) Event distribution - London



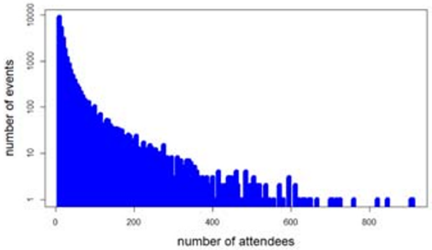
(d) User distribution - London



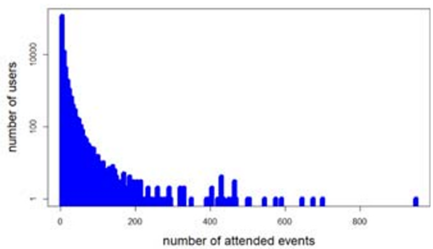
(e) Event distribution - Sydney



(f) User distribution - Sydney



(g) Event distribution - San Francisco



(h) User distribution - San Francisco

Fig. 3 Distributions of events and users in the four cities in year 2016

Table 1 Dataset statistics

City	#events	#users	#YES	#NO	Sparsity
New York	36858	167861	644423	206238	0.99986
San Francisco	24642	136182	519836	159118	0.99979
Sydney	10232	44462	175460	55233	0.99949
London	31980	142731	574051	244106	0.99982

5.1.4 Comparison methods

We compared our method, **ECA** for an effective content-based event recommendation model for active-friends, with three baseline methods listed below:

- Most-Popular (**MP**). This method is the traditional popularity-based approach that selects the users in a group who frequently attended the events in the training set for recommendation.
- User-based collaborative filtering method using Pearson correlation (**Pearson**)
- User-based collaborative filtering method using Cosine correlation (**Cosine**)

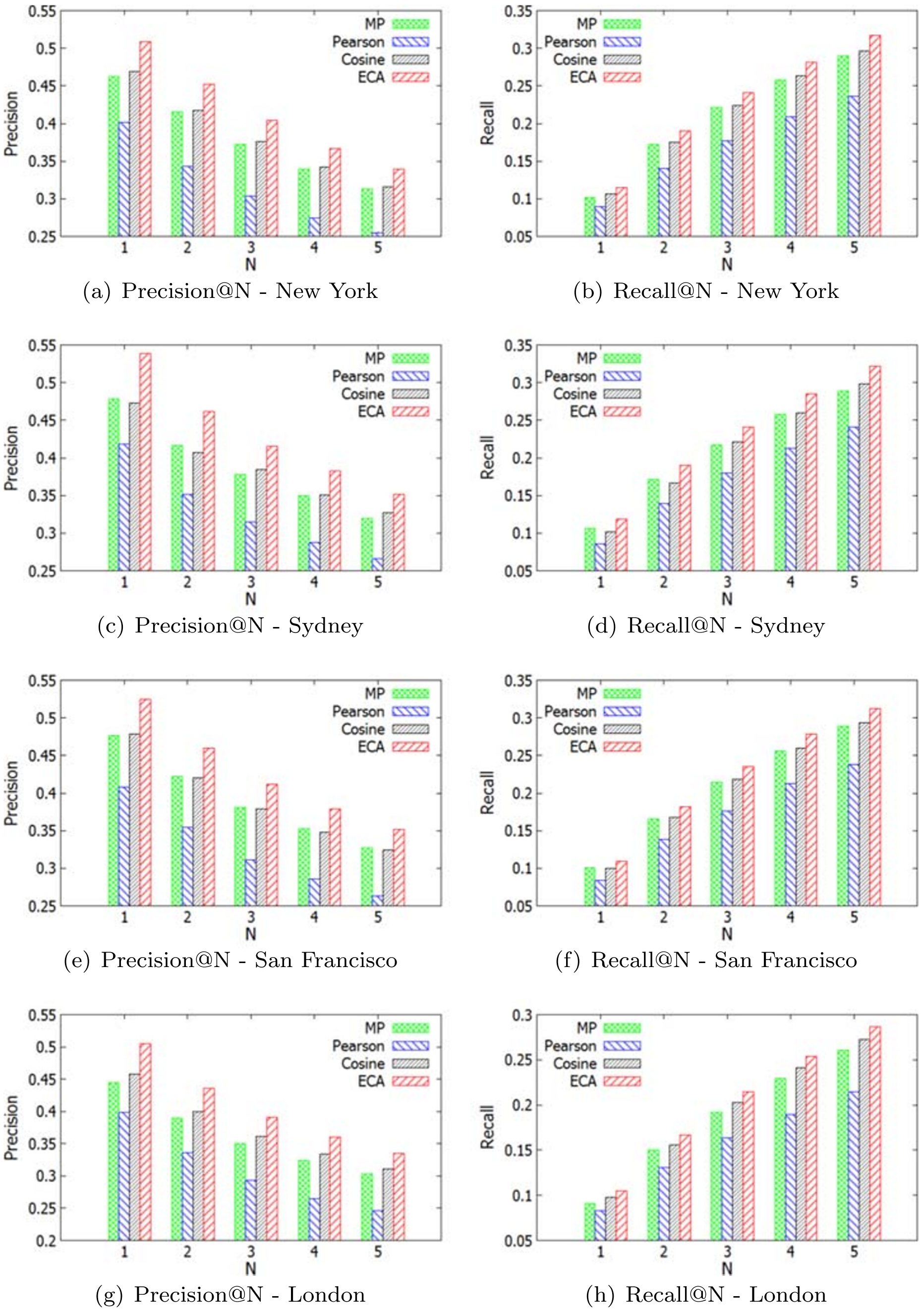
Platform The experiments were conducted on a machine with $12GB$ main memory and $3.4GHz$ dual-core CPU. All algorithms were written in Java. The operating system was Window 10.

5.2 Results and analysis

In this section, we compare the results of the proposed model with the results of baseline algorithms. The experiments were conducted on four datasets. For each event e in the testing set, we assigned one event organizer as the key user u^*. To evaluate the performance of all methods, we set the size of the recommended friend list of u^* from 1 to 5.

As described before, our model consists two phases. The first phase is to compute the similarity between the chosen events E^C and the upcoming event. The second phase is to compute the possibility scores of users. To be fair on the performance comparison of all methods, we used these chosen events E^C to compute the scores of users u to u^* when making comparisons for all methods. The UE matrix is constructed from E^C and a set of attendees obtained from E^C. Elements of the matrix represent the similarity between users and attended events. With this matrix, we can use $Cosine$ and $Pearson$ methods to compute similarities between users.

The results of the experiments are given in Fig. 4, where N is changed from 1 to 5. From these figures, it is clear that our proposed method outperformed all baseline methods in precision and recall metrics, and the $Pearson$ is the worst for the four cities with different Ns. We also observe that the precision of all methods decreases and the recall increases as N increases. It indicates that the recommendations closer to the top are more accurate, and these results also reflect the real life situations that many people do not have many close friends. The two user-based methods, $Pearson$ and $Cosine$, yield the worse results. Because the matrix constructed by the set events E^C and the users of E^C is sparse, these two methods result in low predictions. The other traditional method, MP, recommends those users who frequently attended the events. However, the users who were more active in the past may not be active in the present time, so it leads to worse results. In our method, this



(a) Precision@N - New York
(b) Recall@N - New York
(c) Precision@N - Sydney
(d) Recall@N - Sydney
(e) Precision@N - San Francisco
(f) Recall@N - San Francisco
(g) Precision@N - London
(h) Recall@N - London

Fig. 4 Performance of the four methods in event recommendation for active-friends weakness is overcome by (7). Moreover, we also consider topics, and interests together to compute the possibility scores of users. Therefore, our method produced better predictions in the four datasets.

6 Conclusions and future work

In this paper, we have studied the problem of event recommendation for active-friends in event-based social networks. The study was carried on the extensive datasets crawled from Meetup. We identified social topics and personal interests impacting on decisions of users to take part in events. Moreover, they are likely to attend events with friends. We developed the effective content-based model that is formed by chosen events and users to solve the problem. The proposed model produces a list of users with possibility scores, and recommends the future event to the top N users. Through evaluation using the datasets from four cities, the comparison results demonstrate the effectiveness of our model. In the future work, we will study the loyalty problem in social networks to address the question why users leave their group and do not take part in events, and what factors affect their decisions.

References

1. Athira U, Thampi SM (2018) Linguistic feature based filtering mechanism for recommending posts in a social networking group. IEEE Access 6:4470–4484. https://doi.org/10.1109/ACCESS.2017.2789200
2. Bagci H, Karagoz P (2016) Context-aware friend recommendation for location based social networks using random walk. In: Proceedings of the 25th international conference companion on World Wide Web - WWW '16 Companion. ACM Press, New York, pp 531–536. https://doi.org/10.1145/2872518.2890466. http://dl.acm.org/citation.cfm?doid=2872518.2890466
3. Blei DM, Edu BB, Ng AY, Edu AS, Jordan MI, Edu JB (2003) Latent Dirichlet allocation. J Mach Learning Res 3:993–1022. https://doi.org/10.1162/jmlr.2003.3.4-5.993. arXiv:1111.6189v1
4. Chen CC, Sun YC (2016) Exploring acquaintances of social network site users for effective social event recommendations. Inf Process Lett 116(3):227–236. https://doi.org/10.1016/j.ipl.2015.11.013
5. Chorley MJ, Whitaker RM, Allen SM (2015) Personality and location-based social networks. Comput Hum Behav 46:45–56. https://doi.org/10.1016/j.chb.2014.12.038
6. Chu CH, Wu WC, Wang CC, Chen TS, Chen JJ (2013) Friend recommendation for location-based mobile social networks. Proceedings - 7th international conference on innovative mobile and internet services in ubiquitous computing, IMIS 2013, pp 365–370. https://doi.org/10.1109/IMIS.2013.68
7. Darling W (2011) A theoretical and practical implementation tutorial on topic modeling and Gibbs sampling. Proceedings of the 49th Annual Meeting of the … pp 1–10, http://www.uoguelph.ca/~wdarling/research/papers/TM.pdf
8. Ding H, Yu C, Li G, Liu Y (2016) Event participation recommendation in event-based social networks. In: Spiro E, Ahn YY (eds) Social informatics: 8th international conference, SocInfo 2016, Bellevue, WA, USA, November 11-14, 2016, Proceedings, Part I. Springer International Publishing, Cham, pp 361–375. https://doi.org/10.1007/978-3-319-47880-7_22
9. Dong C, Shen Y, Zhou B, Jin H (2016) I2Rec: An iterative and interactive recommendation system for event-based social networks. Lecture Notes in Computer Science, vol 9708. Springer International Publishing, Cham, pp 250–261, https://doi.org/10.1007/978-3-319-39931-7_24. http://link.springer.com/10.1007/978-3-319-39931-7, http://link.springer.com/10.1007/978-3-319-39931-7_24
10. Du R, Yu Z, Mei T, Wang Z, Wang Z, Guo B (2014) Predicting activity attendance in event-based social networks. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing - UbiComp '14 Adjunct. ACM Press, New York, pp 425–434. https://doi.org/10.1145/2632048.2632063, http://dl.acm.org/citation.cfm?doid=2632048.2632063
11. Eirinaki M, Gao J, Varlamis I, Tserpes K (2018) Recommender systems for large-scale social networks : A review of challenges and solutions. Futur Gener Comput Syst 78:413–418. https://doi.org/10.1016/j.future.2017.09.015

12. Griffiths TL, Steyvers M, Blei B, Blei J (2004) Finding scientific topics. Pnas 101(SUPPL.1):5228–5235. https://doi.org/10.1073/pnas.0307752101. www.pnas.orgcgidoi10.1073pnas.0307752101
13. Hannon J, Bennett M, Smyth B (2010) Recommending Twitter users to follow using content and collaborative filtering approaches, pp 199–206
14. Hoang DT, Tran VC, Dosam Hwang B (2017) Social network-based event recommendation. Lecture Notes in Computer Science, vol 10448. Springer International Publishing, Cham, https://doi.org/10.1007/978-3-319-67074-4. http://link.springer.com/10.1007/978-3-319-67077-5, http://link.springer.com/10.1007/978-3-319-67074-4, arXiv:1011.1669v3
15. Horowitz D, Contreras D, Salamó M (2018) EventAware: A mobile recommender system for events. Pattern Recogn Lett 105:121–134. https://doi.org/10.1016/j.patrec.2017.07.003
16. Jhamb Y, Fang Y (2017) A dual-perspective latent factor model for group-aware social event recommendation. Information Processing & Management 53(3):559–576. https://doi.org/10.1016/j.ipm.2017.01.001. https://linkinghub.elsevier.com/retrieve/pii/S0306457316302357
17. Li R, Lei KH, Khadiwala R, Chang KCC (2012) TEDAS: A twitter-based event detection and analysis system. Proceedings - international conference on data engineering, pp 1273–1276. https://doi.org/10.1109/ICDE.2012.125
18. Li S, Xiang BC, Su S, Jiang L (2016) Followee recommendation in event-based social networks. In: Lecture Notes in Computer Science, vol 9645. Springer International Publishing, Cham, pp 27–42. https://doi.org/10.1007/978-3-319-32055-7, http://link.springer.com/10.1007/978-3-319-32055-7
19. Li X, Cheng X, Su S, Li S, Yang J (2017) A hybrid collaborative fi ltering model for social in fl uence prediction in event-based social networks. Neurocomputing 230(September 2016):197–209. https://doi.org/10.1016/j.neucom.2016.12.024
20. Lin J (1991) Divergence measures based on the Shannon entropy. IEEE Trans Inf Theory 37(1):145–151. https://doi.org/10.1109/18.61115
21. Liu S, Wang B, Xu M (2017) SERGE: Successive event recommendation based on graph entropy for event-based social networks. IEEE Access 6:3020–3030. https://doi.org/10.1109/ACCESS.2017.2786679
22. Liu X, He Q, Tian Y, Lee WC, McPherson J, Han J (2012) Event-based social networks. In: Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining - KDD ’12. ACM Press, New York, p 1032. https://doi.org/10.1145/2339530.2339693. http://dl.acm.org/citation.cfm?doid=2339530.2339693
23. Lu D, Voss C, Tao F, Ren X, Guan R, Korolov R, Zhang T, Wang D, Li H, Cassidy T, Ji H, Chang Sf, Han J, Wallace W, Hendler J, Si M, Kaplan L (2016) Cross-media event extraction and recommendation. In: Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: demonstrations, association for computational linguistics, Stroudsburg, PA, USA, vol 2016, pp 72–76. https://doi.org/10.18653/v1/N16-3015. http://aclweb.org/anthology/N16-3015
24. Macedo AQ, Marinho LB, Santos RLT (2015) Context-aware event recommendation in event-based social networks. In: Proceedings of the 9th ACM conference on recommender systems - RecSys ’15, vol 58. ACM Press, New York, pp 123–130. https://doi.org/10.1145/2792838.2800187. http://dl.acm.org/citation.cfm?doid=2792838.2800187
25. Minkov E, Charrow B, Ledlie J, Teller S, Jaakkola T (2010) Collaborative future event recommendation. In: Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM ’10. ACM Press, New York, p 819. https://doi.org/10.1145/1871437.1871542. http://portal.acm.org/citation.cfm?doid=1871437.1871542
26. Ogundele TJ (2017) EventRec : Personalized event recommendations for smart event-based social networks. 2017 IEEE International Conference on Smart Computing (SMARTCOMP) (1):1–8. https://doi.org/10.1109/SMARTCOMP.2017.7947006
27. Ogundele TJ (2017) SoCaST : exploiting social , categorical and spatio-temporal preferences for personalized event recommendations. https://doi.org/10.1109/ISPAN-FCST-ISCC.2017.68
28. Ogundele TJ, Member S (2018) SoCaST *: Personalized Event Recommendations for Event-Based Social Networks : A Multi-Criteria Decision Making Approach. IEEE Access 6(1):27579–27592. https://doi.org/10.1109/ACCESS.2018.2832543
29. Pham TAN, Li X, Cong G, Zhang Z (2015) A general graph-based model for recommendation in event-based social networks. In: 2015 IEEE 31st international conference on data engineering, pp 567–578. https://doi.org/10.1109/ICDE.2015.7113315
30. Qiao Z, Zhang P, Cao Y, Zhou C (2014) Combining heterogenous social and geographical information for event recommendation. Twenty-Eighth AAAI ... pp 145–151, http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8451

31. Qiao Z, Zhang P, Zhou C, Cao Y, Guo L, Zhang Y (2014) Event recommendation in event-based social networks (2):3130–3131
32. Trinh T, Nguyen NT, Wu D, Huang JZ, Emara TZ (2019) A new location-based topic model for event attendees recommendation. 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF) pp 1–6. https://doi.org/10.1109/rivf.2019.8713716
33. Tu W, Cheung DW, Mamoulis N, Yang M, Lu Z (2015) Activity-partner recommendation. In: Cao T, Lim EP, Zhou ZH, Ho TB, Cheung D, Motoda H (eds) Advances in knowledge discovery and data mining: 19th PacifAsia conference, PAKDD 2015, Ho Chi Minh City, Vietnam, May 19-22, 2015, Proceedings, Part I. Springer International Publishing, Cham, pp 591–604. https://doi.org/10.1007/978-3-319-18038-0_46
34. Xu C (2018) A novel recommendation method based on social network using matrix factorization technique. Inf Process Manag 54(3):463–474. https://doi.org/10.1016/j.ipm.2018.02.005
35. Xu M, Liu S (2019) Semantic-enhanced and context-aware hybrid collaborative filtering for event recommendation in event-based social networks. IEEE Access 7:17493–17502. https://doi.org/10.1109/ACCESS.2019.2895824
36. Zhang S, Lv Q (2018) Knowledge-based systems hybrid EGU-based group event participation prediction in event-based social networks. Knowledge-Based Systems 143:19–29. https://doi.org/10.1016/j.knosys.2017.12.002
37. Zhang W, Wang J (2015) A collective Bayesian poisson factorization model for cold-start local event recommendation categories and subject descriptors, pp 1455–1464
38. Zhang W, Wang J, Feng W (2013) Combining latent factor model with location features for event-based group recommendation. In: Proceedings of the 19th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, NY, USA, KDD '13, pp 910–918. https://doi.org/10.1145/2487575.2487646
39. Zhang Y, Wu H, Sorathia V, Prasanna VK (2013) Event recommendation in social networks with linked data enablement, pp 371–379. https://doi.org/10.5220/0004443903710379
40. Zhu Z, Shi L, Liu B, Ma Z (2018) Multi-feature based event recommendation 11:618–633

Thanh Trinh is from Vietnam. He received the MSc degree in Information Systems Design from University of Central Lancashire, UK. He is currently pursuing the Ph.D. degree with Shenzhen University, China. He has authored several papers on his research topics. His research includes topic model, database, social network, classification, forecasting disasters.

Dingming Wu received the PhD degree in computer science from Aalborg University, Denmark, in 2011. She is an assistant professor in the College of Computer Science & Software Engineering, Shenzhen University, China. Her general research area is data management and mining, including data modeling, database design, and query languages, efficient query and update processing, indexing, and mining algorithms.

Ruili Wang received the Ph.D. degree in computer science from Dublin City University, Dublin, Ireland., He is currently an Professor with the School of Engineering and Advanced Technology, Massey University, Auckland, New Zealand, and is also the Director of the Center of Language and Speech Processing, and the Chair of Research and International, Institute of Natural and Mathematical Sciences.. His research interests include speed processing, language processing, image processing, data mining, intelligent systems, and complex systems.,Dr. Wang serves as an Associate Editor and a member of the editorial boards of six international journals. He has received one of the most prestigious research grants in New Zealand, i.e., the Marsden Fund.

Joshua Zhexue Huang received his Ph.D. degree at The Royal Institute of Technology, Sweden in June, 1993. He is currently a Distinguished Professor of College of Computer Science & Software Engineering at Shenzhen University. He is also the director of Big Data Institute and the deputy director of National Engineering Laboratory for Big Data System Computing Technology. His main research interests include big data technology and applications. Prof. Huang has published over 200 research papers in conferences and journals. In 2006, he received the first PAKDD Most Influential Paper Award. Prof. Huang is known for his contributions to the development of a series of k-means type clustering algorithms in data mining, such as kmodes, fuzzy k-modes, k-prototypes and w-k-means that are widely cited and used, and some of which have been included in commercial software. He has extensive industry expertise in business intelligence and data mining and has been involved in numerous consulting projects in Australia, Hong Kong, Taiwan and mainland China.