Data & Knowledge Engineering 95 (2015) 66–86

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Editorial

A system for mining interesting tourist locations and travel sequences from public geo-tagged photos

Abdul Majid [a,b], Ling Chen [a,*], Hamid Turab Mirza [a,c], Ibrar Hussain [a,d], Gencai Chen [a]

[a] College of Computer Science, Zhejiang University, Hangzhou, P.R. China
[b] Department of Computer Science & IT, University of Azad Jammu & Kashmir, Pakistan
[c] COMSATS Institute of Information Technology (CIIT), Lahore, Pakistan
[d] COMSATS Institute of Information Technology (CIIT), Attock, Pakistan

ARTICLE INFO

Article history:
Received 18 September 2012
Received in revised form 28 October 2014
Accepted 8 November 2014
Available online 14 November 2014

Keywords:
Geo-referenced photographs
Trip planning
Context-aware query
Travel sequence
Spatio-temporal data mining

ABSTRACT

Geo-tagged photos of users on social media sites (e.g., Flickr) provide plentiful location-based data. This data provide a wealth of information about user behaviours and their potential is increasing, as it becomes ever-more common for images to be associated with location information in the form of geo-tags. Recently, there is an increasing tendency to adopt the information from these geo-tagged photos for learning to recommend tourist locations. In this paper, we aim to propose a system to recommend interesting tourist locations and interesting tourist travel sequences (i.e., sequence of tourist locations) from a collection of geo-tagged photos. Proposed system is capable of understanding context (i.e., time, date, and weather), as well as taking into account the collective wisdom of people, to make tourist recommendations. We illustrate our technique on a sample of public Flickr data set. Experimental results demonstrate that the proposed approach is able to generate better recommendations as compared to other state-of-the-art landmark based recommendation methods.

1. Introduction

Web 2.0 applications such as blogs, discussion forums, social and professional networks, and various other types of social media have changed users' online activities. These activities can no longer be characterized by just searching or browsing (read-only). The Internet usage is evolving to interacting, and quickly to creating and sharing contents. The Web users are no longer mere *consumers* of information, but the *producers* of information. They actively participate in social networks to share contents such as they upload their personal photos, share their bookmarks, write blogs, and annotate and comment on the information provided by others. They create information, build content and establish online communities. The facilitation of participatory collaboration has resulted in a paradigm shift, blurring the distinction between consumers and producers that had existed in the Web since its early days [1].

Geo-tagging is the application of geo-spatial metadata (i.e., latitude, longitude) to a range of Web based media (e.g., photos, videos, articles). The proliferation of digital photo and video-capture devices equipped with global positioning system (GPS) and the growing

* Corresponding author. Tel.: +86 13606527774.
E-mail addresses: majid@cs.zju.edu.cn (A. Majid), lingchen@cs.zju.edu.cn (L. Chen), hamid306@cs.zju.edu.cn (H.T. Mirza), ibrar@cs.zju.edu.cn (I. Hussain), chengc@cs.zju.edu.cn (G. Chen).

http://dx.doi.org/10.1016/j.datak.2014.11.001
0169-023X/

practices of sharing photos and videos online using social media sites, such as Flickr (*flicker. com*) and YouTube (*youtube. com*), have resulted in huge volumes of geo-tagged photos and videos available on the Web. Geo-tags of photos and videos provide a wealth of information about user behaviours and their potential is increasing, as it becomes ever-more common for images to be associated with location information in the form of geo-tags. This abundant location based data provided by geo-tagged photos and videos can be potentially used to provide a number of location specific information and services [2]. Recently, there is an increasing tendency to adopt the information from these geo-tagged photos for learning to recommend tourist locations.

For a tourist, before traveling to an unfamiliar city, the most important preparation is planning the trip. Without any prior knowledge, tourist must either rely on travel books, personal travel blogs, or a combination of online resources and services such as travel guides, map services, public transportation sites, and human intelligence to piece together an itinerary [3]. It is difficult, time consuming and painstaking to find out the locations worth to visit and to figure out the order in which they are to be visited.

A tourist needs two kinds of information to understand an unfamiliar city and plan a trip to visit it. That are, (1) the most interesting locations within the city, and (2) given these interesting locations in a city, what are the interesting travel sequences among them? Furthermore, context factors such as time, location, or weather might affect the preferences of users in terms of visiting a location or multiple locations in a certain sequence. It is recognized that the preferences of users can be affected by their context, and the goal of context-aware recommender systems is to deliver recommendation of better quality by incorporating available contextual information of the user [4,5].

Existing methods for tourist recommendations based on geo-tagged social media (e.g., [3,6–14]) addressed queries either with free of context constraints or with a few dimensions of context. In this paper, we focus on context-aware interesting locations and interesting travel sequences (a trajectory of tourist locations) recommendations based on geo-tagged photos. The method we propose is designed to be deployed in an application scenario that leverages the collective wisdom of people from collection of community contributed geo-tagged photos to provide a set of tourist locations and tourist travel sequences that are *interesting* and match the user's current *context* given a city that is new to that user. Note that, "interesting", "popular" or "significant" are subjective terms. Different people might have different ideas of defining them, and we are going to propose a reasonable one. To our best knowledge, this is the first research work that uses the context from the photo, such as spatial and temporal, in combination with weather context, retrieved from online weather Web services, to support the context-aware framework for the recommendations of interesting locations and travel sequences. Our contributions in this paper are summarized as follows.

1. We illustrate a system that recommends interesting tourist locations and tourist travel sequences using contextualized user-generated contents such as photos, from the social media repository Flickr.
2. Methods are proposed to:
   (a) Cluster Geo-tagged photos in order to find tourist locations within a city.
   (b) Aggregate clustered photos' textual information and enriches with supplementary information provided from Web services such as Google Places, to provide semantic meaning to aggregated locations.
   (c) Profiling locations. Temporal tags annotated to photos are exploited to infer users' visits for profiling locations. Profile of each location provides the information about the users who have visited that location, and the history of contexts (i.e., weather and temporal) in which the location has been visited.
   (d) Analyze temporal information to map and construct user travel sequences that define relationships between locations that the users have visited.
3. A probability-based approach is used to filter the tourist locations in the target city to meet the active user's current context. Moreover, a reasonable function is defined, based on user-expertise model, i.e., weighing of tourist locations' significance through the number of user visits to specific location categories, to score tourist location for ranking.
4. Evaluation of the performance achieved against an actual large scale geo-tagged dataset held by Flickr is provided.

In the rest of this paper, we first survey related work (Section 2). We then formally define the problem (Section 3), and propose our system for mining context-aware locations and travel sequences (Section 4). Next, we describe our evaluation framework (Section 5), present the results, and close with a summary and directions for future work (Section 6).

2. Related work

There are three areas of work related to this work. One deals with mapping GPS data to tourist locations, and locations' recommendations. Second is about mining travel itineraries and the other aims context-awareness in recommendations.

In [15,16], GPS trajectory data is utilized (1) to extract the interesting locations, and classical travel sequences by employing Hypertext Induced Topic Search (HITS) based inference method to users' location histories modelled by a tree-based hierarchical graph, and (2) to make location recommendations using the similarity of users in terms of their location histories. Yoon et al. [17], used user-generated GPS trajectories to extract social attributes in order to model and define social itineraries. In [18], an interactive process is adopted where the user provides feedback on Point Of Interests (POIs) suggested by their itinerary planning system and the system leverages those feedback to suggest the next batch of POIs, as well as to recommend the best itineraries so far. From historical trajectory data of users, [19] demonstrated a system to predict personal route of a user using probabilistic model. They proposed a novel mining algorithm for the extraction of route patterns. The main obstacle for GPS trajectories-based methods is that the data resources is not easy to obtain from a large number of people.

In order to provide location recommendations, location based services [16,20] provide location recommendations by first clustering the user-location matrix, which represents the locations visited by each user, and then making location recommendations based on the user and location relationships. However, a major problem with these methods is that these recommendation methods treat every user equally. But there exist reliable users that have much deeper and broader knowledge of a specific domain. These users' preference information is more credible and plausible. In this work, instead of merely considering users and locations represented in a user-location matrix, a new model is defined to capture the relationships among the three important entities, namely, users, locations and location-categories, in a tripartite data structure. Then a reasonable function, based on user-expertise model is defined to score tourist location for ranking. User expertise and other user related information are widely used for evaluating answer quality in community question and answering services [21,22] and in online forums [23].

With the development of the Web 2.0, many researchers focus on mining knowledge from social media. Geo-tagged photographs available on these collections provide a quick overview of the interesting places at travel destinations [24]. These photographs exhibit larger geographical coverage, and typically reflect the tourist trips of users sharing these images. Thus, it is intriguing to consider the possibility of utilizing the proverbial “wisdom of crowds” embedded in Flickr-like sources to mine sightseeing trips automatically [25]. Lately, some research efforts have been made to map geo-tags of photos to identify tourist locations based on assumptions that the tourist places are those geographical regions that are highly photographed [26–30]. Ahern et al. [26] created a World Explorer that used tags on Flickr geo-tagged photos to map well-liked tags to geographical locations, resulting in a scale dependent map overlayed with semantic information on the original data. This work is extended in [27] by applying content and context based analysis for ranking clusters and finding representative images in a cluster. [28] divided the map using a grid so each cell represents a location. They defined a language model to describe the relation of a tag and a place and estimated a cell as the place where a photo was taken by using tags annotated to the photo. A density based clustering is used to discover tourist locations from photos' geo-tags in [29]. Another work based on spectral clustering about identifying location as POIs is by [30]. They proposed a self-tuning approach based on the cut cost similarity to eliminate the effect of parameters from spectral clustering. Rattenbury et al. [31] investigated the place and event semantics of geotags, in addition to the representativeness. The proposed approach can automatically determine whether a tag corresponds to a “place” or an “event”. A “place” tag is defined as a one that exhibits significant spatial patterns, while an “event” tag refers to a one that exhibits significant temporal patterns. [32] presented a method to recommend tourist locations based on user's travel history in a collaborative filtering manner. Locations are ordered based on their popularity and then popularity score is linearly combined with personalized score weighted by the similarities between the active user and other users.

To infer the knowledge about tourist locations that are found using spatial proximity of photos, existing works used visual features of photos or photos metadata such as title, tags and description. Due to the unrestricted nature of photo sharing applications, one or more of the aforementioned metadata fields might be missing or incorrect. In our work, first we apply a density based clustering algorithm to the geo-tags of photos to extract tourist locations, then we aggregate photos' textual tags and enrich with supplementary information provided from a Web service such as Google Places to give semantic meaning to aggregated locations. Furthermore, to summarize the aggregated locations and to derive the dynamics of users' interests to these locations, temporal tags annotated to photos are exploited to infer users' visits for profiling locations. Profile of each location provides the information about the users who have visited that location, and the history of contexts that are weather and temporal, in which the location has been visited. Note that, we identify the temporal context of a visit by exploiting the time-stamps of photos that were taken during that visit and use this visit time to obtain the weather context of the visit from historical weather dataset retrieved from online weather resources.

Some studies focus on mining trip information based on the sequence of locations extracted from geo-tags of photos' collection. In [3], Choudhury et al. formulated trip planning as directed orienteering problem. [6] explored the photo sharing to assist two kinds of tourists. First, those who want important locations in recommended trajectory pattern, and second, those who want to explore the city in a diverse way. To deduce trip related information, [7] utilized the temporal information associated with photos. For recommendations, they focused the query with temporal constraints in terms of duration of the trip. An interactive tourist recommendation method is proposed by [8] that took into account a number of factors such as duration of the trip and travelling cost to help the tourist for trip planning. Large number of photos contributed by many people on the Web have been used for trip planning by extracting actual travel paths [9], Urban mobility analysis [10], and mining frequent trajectory patterns [13].

In the area of context-aware recommendations, Abowd et al. [33] proposed a context-aware tour guide named Cyberguide. They considered that a mobile application should take advantage of contextual information, such as position, to offer greater services to the user. In [34], it is discussed that how to abstract raw context information to contextual concepts in methods to improve the quality of recommendations. Dongjoo et al. [35] showed how to obtain customers' implicit preferences from event logs. They presented a strategy to abstract context information from event logs considering fuzziness in context. They acquired the customer's preference and context by exploiting the information implied in the customer previous event logs and adopted them into collaborative filtering recommendation method. In [36], it is shown how to incorporate several features into the ranking model. By decomposing a query, they proposed several types of ranking features that reflect various contextual effects. They also presented a retrieval model for using these features, and adopted a learning to rank framework for combining proposed features.

State of the art for tourist recommendation using geo-tagged social media [3,6–14]) addressed queries either with free of context constraints or with a few dimensions of context. In our work, the profiles of locations based on users' visits are built to better understand how locations are engaged by users in different contexts and to infer users' rating to these locations. Furthermore, for making recommendations we not only use the significance of locations or travel sequences but also consider the users' current context. Thus, we extend the notion of tourist locations and travel sequences recommendation while taking into account the contextual information in geo-tagged social media.

3. Problem definition

Before we formally define the problem, we give definitions of some basic concepts and terms.

Definition 1. (*Geo-tagged photo*) A geo-tagged photo p can be defined as $p = (id, t, g, X, u)$ containing a photo's unique identification, id; its geo-tags, g; photo's time-stamp, t; and the identification of the user who contributed the photo, u. Each photo p can be annotated with a set of textual tags, X. Geo-tags g of photo p is the coordinates of the geographical region where photo p was taken.

Definition 2. (*Photo collection*) Collection of all photos, contributed by all tourists can be represented as $P = P_1, P_2,.., P_n$ where $P_u(u = 1.. n)$ is the collection of photos contributed by user u.

Definition 3. (*Context-aware query*) A context-aware query Q is defined as $Q = (t, w)$, where t represents temporal context and w denotes weather context.

The problem of recommending interesting tourist locations and tourist sequences in geo-tagged social media is formulated as; given a collection of geo-tagged photos $P = \{P_1, P_2,.., P_n\}$, (1) how to find tourist locations within a city, infer the semantic of tourist locations, summarize and then rank the tourist locations based on their interestingness for recommendations, and (2) how to build users' travel histories, extract trips made by users, and mine travel sequence patterns for interesting travel sequence recommendations. To be specific, we aim to utilize the photos collections contributed by users publicly for interesting tourist locations and interesting tourist sequences recommendations with respect to user's current context.

4. Mining interesting tourist locations and travel sequences

This section discusses the proposed solution for mining interesting tourist locations and travel sequences.

4.1. System architecture

Fig. 1 depicts the different modules comprising our system. Tourist locations are identified using spatial proximity of photos (Section 4.2) and aggregated locations are enriched with semantic using textual tags annotated to photos in combination with

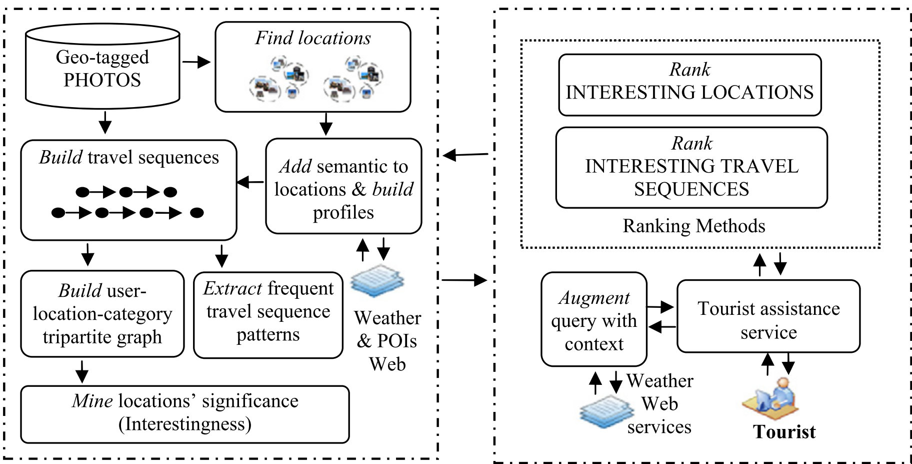


Fig. 1. Architecture for learning context-aware recommendation from geo-tagged media.

information provided by online Web services (Section 4.3). The profiles of locations are built to describe the contexts in which they have been visited (Section 4.4). For temporal context, temporal tags annotated to photos are exploited. Whereas, to derive weather context, third party weather Web services are queried to retrieve weather conditions. Temporal information is analyzed to map and construct the users' trips that define relationships between locations that the users have visited (Section 4.5). Next, a sequential pattern mining algorithm is employed to extract frequent travel sequence patterns (Section 4.6). It is proposed to exploit the relationships among users, locations, and location categories to define significance (interestingness) of each location using a user-expertise model (Section 4.7). For making recommendations, first, the tourist locations and travel sequence patterns are filtered based on user's current contextual constraints, and then locations and travel sequence patterns are ranked by using significance score of locations (Section 4.8).

4.2. Identification of tourist locations

Given a collection of photos that are tagged with location metadata (latitude and longitude), finding tourist locations can be viewed as a clustering problem of identifying highly photographed locations.

Definition 4. (*Tourist location*) A tourist location can be defined as a uniquely represented specific geographic area within the city; such as a sightseeing spot, a store, a restaurant, which is popular for tourists to visit and take photos. Formally, tourist locations identified by clustering geo-tags of photos' collection P can be represented as $L = \{l_1, l_2, \ldots, l_n\}$. Each element $l = \{P_l, g_l\}$, where P_l is a group of geographically clustered photos, and g_l are geographical coordinates to represent the centroid of photos' cluster P_l, and are computed from group of geo-tags associated with the photos in the cluster P_l.

Clustering algorithms such as k-mean and mean-shift have been used to cluster photos using associated geo-tags for the identification of tourist locations [27,6]. However, density based clustering algorithms such as DBSCAN [37] have several advantages over other types of clustering algorithms: they require minimum domain knowledge to determine the input parameters and can discover clusters with arbitrary shape. In addition, they can filter outliers and work effectively when applied to large databases. DBSCAN requires only two parameters: ε (epsilon) and the minimum number of points required to form a cluster (*minPts*). DBSCAN algorithm randomly selects an object and forms a range search with radius ε and iteratively discovers subsequent density reachable objects to make the cluster. DBSCAN clustering works with generic points having a unified density threshold for all clusters; however, the locations extracted by clustering the given collection of photos, can have varying sizes and densities. To address this problem, [29] proposed P-DBSCAN, a variant of DBSCAN. They extended the definition of directly density reachable by adding an adaptive density technique. In P-DBSCAN, an object O is directly density reachable from another object O' if it is not farther away than a given density radius ε and the ratio of surrounding objects between O and O' must be less than a density ratio.

In our work, it is proposed to use P-DBSCAN, in order to cluster photos to identify tourist locations based on the geo-tags of photos in a city.

4.3. Semantic annotation of locations

The geographical locations identified in the result of a clustering method are required to annotate with semantic to describe the locations, which is crucial prerequisite for locations search or recommendation services. The problem of *semantic annotation of locations* can be formulated as predicting appropriate name and category for a given tourist location.

Definition 5. (*Semantic location*) A semantic location l' can be defined as $l' = (l, a)$, where $a = (name, category)$ represents the semantic annotation used to describe the tourist location l.

To give semantic meanings to locations, we provide a method that uses textual tags annotated to photos in combination with the information provided by online Web services, to automatically generate textual descriptions (name) and category for each tourist location. Our method as summarized in Algorithm 1 contains three steps. In the first step (line 2), we use the method described in [27] to derive representative textual tag for each location $l = (P_l, g_l)$. Considering each location $l = (P_l, g_l)$ and set of tags X_l that appear with group of photos P_l, they used a method based on term frequency-inverse document frequency (TF-IDF) to score each tag $x \in X_l$. Note that, TF-IDF is a popular ranking method and is widely used in information retrieval. At the end of step one, for location $l = (P_l, g_l)$ we have a list of tags X_l and each tag $x' \in X'_l$ has a score $s(x)$. The higher the score, the more distinctive the tag is within a group X_l. In the second step (line 3), we use Web services available online, i.e., Google Places (*google.com/places*) to extract the information about the POIs in a certain geographical area. These services work in this way; we provide them a geographical coordinate g and a radius r in meters, in response they return the metadata (i.e., name and type) of places that are present within r of g. We use centroid g_l of location l to represent g. The output of step two is the set $PLACES = \{place\}$ for location l. Each entry in $PLACES$ represented by $place$ provides the information to describe a POI. In the last step, we aggregate the results of step one and step two to get the representative description of tourist location. The aggregation is performed as; we order the set of tags X'_l according to their score that is computed in step one (line 4). We iteratively compare each element of X'_l with all elements of $PLACES$ (line 5-12). In the result of comparison,

(a) If multiple matches are found, then the matched *place* that is closest to geographical coordinate g_l, in terms of spatial distance, we consider its name as the location "*name*" (line 13-14).
(b) If a single match is found, then the *place* that is matched, we use its name as the location "*name*" (line 15-16).
(c) If no match is found, we use the tag with highest score as the location "*name*" (line 17-18).

Algorithm 1. Semantic annotation

```
Require: L = {l} Set of locations
Ensure: L' = {l'} Set of semantic locations
 1: for all location l = (P_l, g_l) ∈ L do
 2:    COMPUTE score s for each tag x ∈ X'_l belongs to photos' group P_l
       using TF-IDF
 3:    RETRIEVE PLACES from POI Web services
 4:    SORT X'l based on score s
 5:    CREATE list MatchedList
 6:    for all x ∈ X'_l do
 7:       for all place ∈ PLACES do
 8:          if MATCH(x, place)=true then
 9:             ADD place to MachedList
10:          end if
11:       end for
12:    end for
13:    if LENGTH(MachtedList)> 1 then
14:       l.name ← CLOSEST(MatchedList).name
15:    else if LENGTH(MachtedList)= 1 then
16:       l.name ← place.name
17:    else
18:       l.name ← TOP(x)
19:    end if
20:    l.category ← TF(GENERALIZE(MatchedList))
21:    ADD l to L'
22: end for
```

Web services provide metadata of Point of Interests (POIs) in terms of *name* and *type*. Google Places supports 126 types to describe POIs for search queries. To infer the "*category*" of location *l*, we further generalize these types into 6 categories, i.e., education, shopping, religious, food, transportation, cultural, and entertainment. We select the "*category*" that is highest in frequency in the list of types associated with returned POIs as "*category*" of *l* (line 20). Popular locations in a city is shown in Fig. 2.

4.4. Profiling locations

Once the tourist locations have been identified by clustering the photos based on their spatial proximity and the aggregated locations have been annotated with semantic, we are interested in formulating the profiles of locations.

Definition 6. (*Location profile*) A profile of a tourist location l' can be represented as $Profile(l) = \{V_l\}$, where V_l represents the set of visits made by users to location *l* and describes that how the location *l* has been engaged and perceived by different users.

The method for locations' profiling is summarized in Algorithm 2. The first step is to identify visits made by different users from photos taken by them on these locations (line 1-12). For each location *l* in *L*, we sort photos of each user *u* according to photos taken time. We infer visit *v* from a photo *p* taken by a user *u* and at location *l* at time *t*. Note that, a user *u* can take more than one photo in same visit at same location. For this, if the difference between the time stamps of two photos ($p_2.t - p_1.t$) taken by same user taken at same location in less than visit duration threshold $visit_{thr}$, we consider both photos belong to same visit. We use the median of time-stamps associated with photos that belong to visit *v* as the visit time $v.t$.

Definition 7. (*Visit*) A visit *v* can be represented as $v = (l, u, t)$ where *u* is the user who made visit *v* at location *l* at time *t*.

The next step is to build the history of contexts in which tourist locations have been visited. To describe this, first we explain some notions formally. Let *L* be the set of all locations extracted and let *U* be the set of all users. For a user $u \in U$, let V_u be the set of visits to different locations made by *u* so we can derive $V = \cup_{u \in U} V_u$. The users who have visited the location *l* can be defined as $U_l = \{u \in U : u\ visited\ location l\}$. If we represent the visits made by user *u* at location *l* as $V_{ul} = \{v \in V_u : v.\ l \in L\}$ then all visits made by all user at

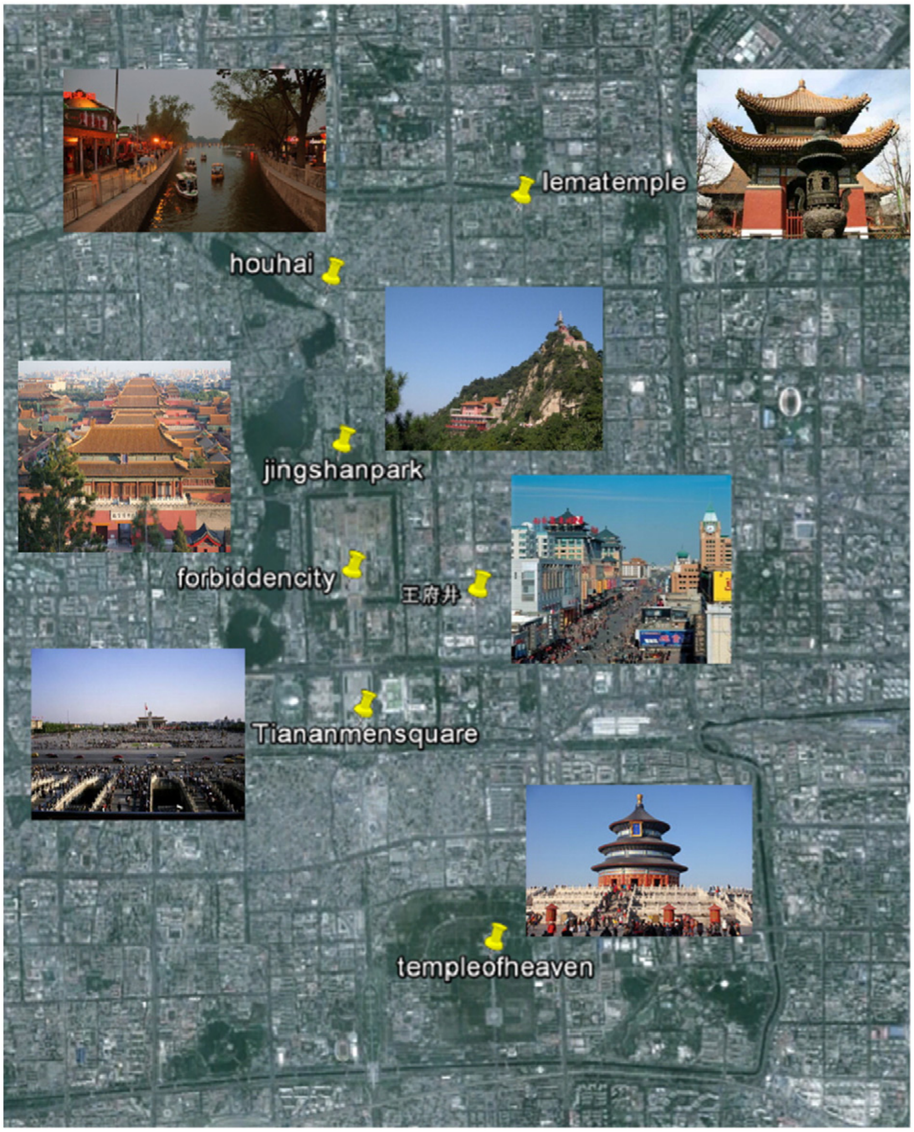


Fig. 2. Popular locations in a region in Beijing, aggregated from photos spatial proximity.

location l can be represented as $V_l = \{v \in V : v.\ l \in L\}$, where $v = (l, u, t)$. The example shown in Fig. 3 depicts how the visits are represented in terms of (l, u, t).

To build history of temporal and weather context, in which the location l has been visited, the available information for each visit is time. This time-stamp information enables us (1) to induce the temporal context t of each visit and (2) to retrieve weather context (condition) w, when visit $v = (l, u, t)$ was made by user u at location l at time t. Weather Services normally publish weather conditions at hourly/daily or monthly level and different variables such as temperature, precipitation, humidity, etc are used to represent these weather conditions. Context related data, such as the time-stamp, and weather variables cannot be directly used as contextual information, thus we need a context abstraction strategy to obtain abstract context concepts. Various context abstraction methods have been proposed for temporal and weather context abstraction [35,34]. For example, the raw context $(21:30, 25C^{\circ})$ can be abstracted to $(night, warm)$. We use context concepts given in Table 1 to represent the temporal and weather contexts of each visit.

For each $v \in V_l$, we transform $v.\ t$ to temporal context concepts given in Table 1. For weather context, we retrieve weather conditions of location l at $v.\ t$ from historical weather database and represent it using weather context concepts given in Table 1 (line 13). After the context retrieval and abstraction, each visit v belong to set of visits $V_l = \{v \in V : v.\ l \in L\}$ made to location l can be expressed as $v = (u, l, t, w)$. An example of visits' representation with abstract contextual concepts is depicted in Fig. 4.

Visits	Locations	Users	Time-stamp
v_1	l_1	u_1	18/04/2009 05 : 20 : 23
v_2	l_1	u_2	22/01/2011 07 : 40 : 05

Fig. 3. Representation of visits in terms of (u, l, t).

Algorithm 2. Profiling locations

```
Require: L = {l} Set of locations where l = (P_l, g_l)
Ensure: LDB = Database of locations with updated profiles
 1: for all location l = (P_l, g_l) ∈ L do
 2:   CREATE list V_l
 3:   CREATE list of users U_pl from P_l and SORT photos P_ul ∈ P_l taken
      by each user u ∈ U_Pl according to photo taken time p.t
 4:   for all user u ∈ U_Pl do
 5:     CREATE list T_v
 6:     for all p ∈ P_ul do
 7:       if p_i.t − p_{i−1}.t < visit_thr then
 8:         ADD p.t to T_v
 9:       else
10:         v ← NEW(visit)
11:         v.t ← MEDIAN(T_v)
12:         v.w ← RETRIEVE-FROM-WEATHER-DB(v.t)
13:         ABSTRACT(v.t, v.w)
14:         ADD v to V_l
15:         CLEAR P_v
16:         ADD p to P_v
17:       end if
18:     end for
19:   end for
20:   ADD l to LDB
21: end for
```

After identifying the visits made by different users to different locations and deriving the visited pattern of each location in terms of different contexts, we build a database of tourist locations (line 20).

Definition 8. (*Tourist locations database*) A tourist locations database $LDB = \{l_1, l_2,..., l_n\}$, where each location $l_i = \{V_{l_i}, g_l, a\}$, V_{l_i} are visits made to location l_i by different users, g_l is geographic coordinates (centroid), and a is the semantic to describe the location in terms of name and category.

4.5. Identification of users' trips

The photos, together with their time and geo-references, become the digital footprints of photo takers and implicitly document their spatio-temporal movements. The problem of identifying user's trips can be taken as the leveraging of wealth of these enriched online photos to retrieve the order in which the tourist locations are visited by users. The users' trips can be formally defined as following.

Definition 9. (*User trip*) A user trip is a sequence of locations visited by a user according to temporal order and the difference between the visiting time of two consecutive locations in the sequence is not greater than a $trip_{dur}$ threshold. More specifically, a user trip can be denoted as $TR = l_1, l_2,...., l_n$, where $l_i.\,t > l_{i-1}.\,t$ and $l_i.\,t - l_{i-1}.\,t > trip_{dur}$.

We propose to adopt the following procedure (as summarized in Algorithm 3) to extract the trips made by different users from their contributed photos.

1. Time-stamps annotated to a user's contributed photos are exploited to sort the photos in order to yield his travelling history. Each photo is replaced with its corresponding tourist location from *LDB* (line 1-3).
2. Travel history of each user is split into user trips if the difference in the time stamps of two consecutive photos ($p_{i+1}.\,t - p_i.\,t$) is greater than a given threshold $trip_{dur}$ (line 3-7).
3. A user can have more than one photos in one location in one visit. In this case, if two consecutive photos represent the same location then only one photo is taken into consideration (line 7-16).

After the extraction of trips for all users, a trips database is maintained (line 17-22). Fig. 5 shows the sample trips taken by two users in Beijing.

Table 1
Temporal and weather context concepts.

Temporal context concepts	Weather context concepts
Day of week: working day, weekend	Temperature: hot, warm, cold
Time of day: morning, afternoon, night	Condition: sunny, cloudy, rainy

Visits	Locations	Users	Temporal Context	Weather Context
v_1	l_1	u_1	Weekday, Morning	Warm, Sunny
v_2	l_1	u_2	Weekday, Evening	Cold, Raining

Fig. 4. Representation of visits in terms of (u, l, t, w).

Definition 10. (*Users' trip database*) A users' trip database can be denoted as $TDB = \{T_1, T_2, \ldots, T_n\}$, where T_i represents the set of trips made by user i.

4.6. Mining travel sequence patterns

Mining tourist travel sequence patterns involves the extraction of frequent sequences of locations (travel sequence patterns) from users' trips. Frequent means that a sequence occurs in no less than a given threshold, called minimum support min_{sup}.

Definition 11. (*Travel sequence pattern*) A travel sequence pattern s is a pair (sp, f), where $sp = \{l_1, l_2, \ldots, l_n\}$ is the sequence of locations, and $f \geq min_{sup}$ is the sequence frequency. The sequence frequency is the number of users visiting the locations according to the order in the sequence.

Prefix-projected Sequential Pattern mining (Prefix-Span) [38] is a popular algorithm for the extractions of frequent item-set patterns from the database. Prefix-Span enumerates frequent sequential patterns from a set of sequences. In our work, it is proposed to use re-fixSpan algorithm (as summarized in Algorithm 4) in order to extract the frequent sequential patterns from user trips.

Definition 12. (*Travel sequence pattern database*) Database of travel sequence patterns extracted from users trips using sequential pattern mining algorithm can be denoted as $SDB = \{s\}$, where $s = (sp, f)$ sp is the set of locations in the travel sequence pattern s and f is the sequence frequency.

Algorithm 3. Extract user trips

```
Require: LDB = {l} Database of locations where l = (P_l, g_l), Collection of
    photos P = PU_1, PU_2, PU_3, ....PU_n where P_i(i = 1..n) are set of photos
    taken by user i.
Ensure: TDB = {T} Database of sets of users' trips
for all PU ∈ P do
    SORT PU based on temporal tag p_i.t to generate travelling history
    of each user
    CREATE list T {collection of each user's travel sequences}
    CREATE list TR {a trip}
    for all p ∈ PU do
        if p_i.t − p_{i−1}.t < trip_dur then
            ADD p to TR
        else
            for all p ∈ TR do
                REPLACE p with corresponding location l from LDB
            end for
            for all l ∈ TR do
                if l_i = l_{i−1} then
                    DELETE l
                end if
            end for
            ADD TR to T
            CLEAR TR
        end if
    end for
    ADD T to TDB
end for
```

4.7. Mining significance of locations

In this section, we elaborate how to mine the significance of locations by giving weight to tourists according to their expertise in different categories of places using travel sequence database *SDB*. To explain this, let us consider the travel sequence database *SDB* contains travel sequences of four users; $S_{u1} = \{(l_1, l_3, l_7), (l_2, l_6)\}$, $S_{u2} = \{(l_2, l_5, l_6), (l_7, l_8)\}$, $S_{u3} = \{(l_3, l_5, l_4)\}$, $S_{u4} = \{(l_4, l_8)\}$,. The category of each location is considered as give in Table 2.

Algorithm 4. Mining travel sequences

Require: TDB Users trips database, minimum support threshold, min_{sup}
Ensure: $SDB = \{s\}$ Database of travel sequences, where $s = (sp, f)$ sp is the set of locations in the travel sequence pattern and f is the sequence frequency

CREATE list $TSDB$
for all $T \in TDB$ **do**
  for all $TR \in T$ **do**
    ADD TR to $TSDB$
  end for
end for
CREATE list SDB
SDB =PrefixSpan($TSDB, min_{sup}$)

We propose a model (called *ULC* Model) to organize user travel histories into a meaningful data structure to facilitate location significance mining. It represents three entities users, locations, and location categories and relationship between these three entities.

Definition 13. (*User-location-category tripartite graph*) A user-location-category tripartite graph can formally denoted as $G_{ULC} = (U; L; C; E_{UL}; W_{UL}; E_{UC}; W_{UC}; E_{LC})$, where U, L and C are nodes to represent users, locations and location categories respectively. E_{UL} and W_{UL}, are sets of edges and edge weights between nodes U and L to represent visits and number of visits. E_{UC} and W_{UC} are sets of edges and edge weights between nodes U and C to represent users' visits and number of visits to particular location categories. E_{LC} are edges between L and C to describe the categories of locations.

The significance of locations is mined using a user-expertise model, i.e., generate a weighing of location significance through the number of user visits to specific location categories.

1. Given m users and n categories, we build an $m \times n$ adjacency matrix M_{UC}. Each entry in $M_{UC}(p, q)$, depicts the experience of user u_p in location category c_q.
2. To capture the relationship between users and locations, given m users and k locations, we build a $m \times k$ adjacency matrix M_{UL}. So in matrix M_{UL}, $I_j = \{i\}$ where $M_{UL}(i, j) \neq 0$, is the set of indices of users who have visited the location l_j.
3. We compute significance score of each location $l_j (j = 1 \cdot\cdot k)$, which is of location category $c_q (q = 1 \cdot\cdot n)$ using Eq. (1).

$$g\left(l_j\right) = \sum_{i \in I_j} M_{UC}(i, q) \tag{1}$$

4.8. Recommendation methods

4.8.1. Tourist locations' recommendations

In this section, different methods to rank the tourist locations are discussed.

4.8.1.1. Popular tourist locations. We rank the locations based on general popularity score, determined in terms of number of unique visits made to those locations.

4.8.1.2. Personalized tourist location recommendations. The second method that is used to rank the tourist locations is user-based collaborative filtering that makes recommendations to users based on their interests (preferences). A user's interests can be formally defined as following.

Definition 14. (User interests) From matrix M_{UL}, the user u_p's travel interests can be derived as an array $R_p = < r_{p0}, r_{p1}, \ldots, r_{pn} >$, where r_{pi} is u_p's implicit rating (visits made by u_p) in a location i. $S(R_p)$ is the subset of R_p, $\forall r_{pi} \in S(R_p)$, $r_{pi} \neq 0$, i.e., the set of locations that has been preferred (visited) by u_p. The average rating in R_p is denoted as $\overline{R}_p$. For example, $R_1 = < 5, 2, 0, 4, 3, 0 >$, $R_2 = < 1, 0, 3, 2, 0, 1 >$ then $S(R_1) = < 5, 2, 4, 3 >$ and $S(R_2) = < 1, 3, 2, 1 >$.

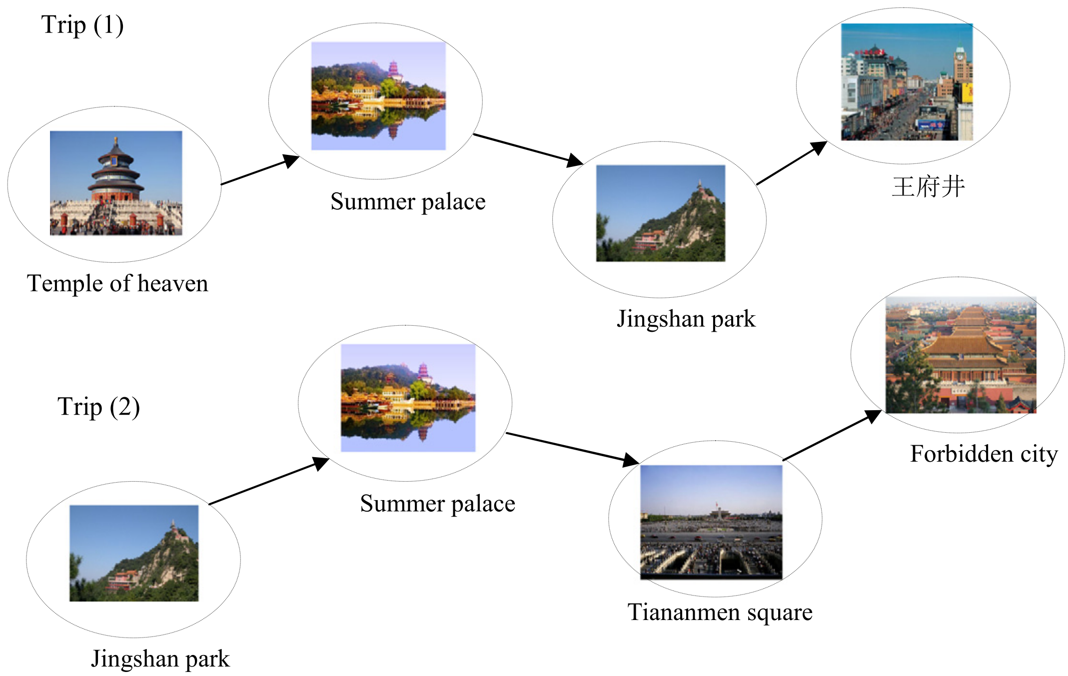


Fig. 5. Sample trips taken by two users in Beijing to visit tourist locations. (Best seen in color).

The similarities among users are calculated based on their travelling preferences using the Pearson correlation metric as given in Eq. (2) and a users' similarity matrix M_{UU} is built. Each entry in M_{UU} represents the similarity between u_p and u_q. A larger value means that both users are more similar in terms of travelling preferences.

$$sim\left(u_p, u_q\right) = \frac{\sum_{j\in S(R_p)\cap S(R_q)} \left(r_{pj}-\overline{R}_p\right).\left(r_{qj}-\overline{R}_q\right)}{\sqrt{\sum_{j\in S(R_p)\cap S(R_q)} \left(r_{p_j}-\overline{R}_p\right)^2} \cdot \sqrt{\sum_{j\in S(R_p)\cap S(R_q)} \left(r_{q_j}-\overline{R}_q\right)^2}} \quad (2)$$

The user-location matrix M_{UL} that represents the users' preference (users' rating) and M_{UU} that represents the similarities among users are used to personalize the recommendations for active user u_p in the target city. From M_{UU}, we retrieve similarities between active user u_p and top N most similar users $U' \in U$, who have visited the target city, and use Eq. (3) to predict preferences of user u_p for each location l_i from L', that is based on collaborative filtering [39]. In collaborative filtering, the user is recommended items that people with similar tastes and preferences liked in the past.

$$Score(l_i) = \overline{R}_p + k\sum_{u_q\in U'} sim\left(u_p, u_q\right).\left(r_{qi}-\overline{R}_q\right) \quad (3)$$

$$k = \frac{1}{|U'|}\sum_{u_q\in U'} sim\left(u_p, u_q\right) \quad (4)$$

Table 2
Location categories.

Location categories	Locations
c_1	l_1, l_2
c_2	l_3, l_5, l_6
c_3	l_4
c_4	l_7, l_8

Time (CST)	Temp.	Wind chill	Dew Point	Humidity	Pressure	Wind Speed	Events
12:00 AM	$3.0C^o$	$1.0C^o$	$-4.0C^o$	60%	1031hPa	7.2km/h/2.0m/s	Clear
12:30 AM	$3.0C^o$	$0.1C^o$	$-3.0C^o$	65%	1030hPa	10.8km/h/3.0m/s	Clear
1:00 AM	$3.0C^o$	$0.1C^o$	$-4.0C^o$	60%	1031hPa	10.8 km/h/3.0m/s	Clear
1:30 AM	$2.0C^o$	$-0.1C^o$	$-4.0C^o$	65%	1030hPa	7.2 km/h/2.0m/s	Clear

Fig. 6. Sample records from historical weather data.

$$\overline{R}_p = \frac{1}{\left|S\left(R_p\right)\right|}\sum_{j \in S\left(R_p\right)} r_{pj} \tag{5}$$

The similarity between user u_p and u_q, $sim(u_p, u_q)$ is used as a weight to calculate the rank score for each location l_i. That is, the more similar u_p and u_q are, the more weight r_{qi} will carry in the prediction of l_i. Instead of using the absolute values of ratings, the deviations from the average rating of the corresponding user is used. One problem with using the weighted sum is that it does not take into account the fact that different users may use the rating scale differently. Therefore, we use an adjusted weighted sum here. In Eq. (3), multiplier k serves as normalizing factor and usually selected as in Eq. (4) and average rating of user u_p, $\overline{R}_p$, from locations in his travelling history, is defined according to Eq. (5).

4.8.1.3. Context-aware significant tourist locations recommendations (CSR). The proposed approach to process the context-aware query $Q(t, w)$ made by user u, to rank the tourist locations based on user's current context and significance of tourist locations consists of two main steps; an initial filtering step retrieves t number of tourist locations of target city from the tourist locations database LDB that have high context-dependent popularity in the current context, thus producing a filtered set of tourist locations L'. The probability that the location l_j can be included in the filtered list L' given the current context C_u of a specific user u can be found by using Eq. (6):

$$P\left(l_j|C_u\right) = \frac{P\left(l_j, C_u\right)}{P(C_u)} = \frac{P\left(l_j\right)P\left(C_u|l_j\right)}{P(C_u)} \tag{6}$$

We assume the independence between different types of contexts (i.e., temporal and weather), such that the joint probability $P(C_u|l_j)$ can be further expressed as the product of the marginals (Eq. (7)),

$$P\left(l_j|C_u\right) = \prod_{c_u \in C_u} \frac{P\left(l_j\right)P\left(c_u|l_j\right)}{P(c_u)} \tag{7}$$

The c_u is one of the tuple of C_u (e.g., weather). $P(c_u|l_j)$ in the product above are easy to be estimated from location's profile (Eq. (8)).

$$P\left(c_u|l_j\right) = \frac{count\left(l_j \wedge C_u = c_u\right)}{count\left(l_j\right)} \tag{8}$$

The $count(l_j)$ is the total number of visits made to location l_j. Similarly, the $count(l_j \wedge C_u = c_u)$ is the total number of visits made to location l_j in context c_u. Note that, to avoid floating-point underflow when computing products of probabilities, all of the computations are in log space.

In the second step, significance score $g(l_j)$ of each location l_j is exploited for ranking locations in L' and top k tourist locations are returned as query result.

Photo Id	User Id	Title	Description	Taken Time	Upload Time	Lat	Long	Tags
303251010	74434506@N00	Shanghai	Shanghai Pearl Tower skyline	9/5/2010 09:22:32	9/20/2010 10:05:12	31.23827	121.487331	chin, river, Pearl Tower, Shanghai, bund
3465811048	32267947@N06	Museum	Shanghai Museum	3/24/2007 14:52:09	4/05/2007 18:05:02	31.23042	121.470871	china, colour, art, archaeology, museum
3057417546	32267947@N06	Bund	Shanghai Bund	3/23/2007 18:25:10	4/05/2007 18:10:32	31.24417	121.486985	river, asia, shanghai, chinese, Pudong, bund, Canada, good

Fig. 7. Sample records from photos metadata.

Photo Id	User Id	Title	Description	Taken Time	Upload Time	Lat	Long	Tags
1	1	Shanghai	Shanghai Pearl Tower skyline	9/5/2010 09:22:32	9/20/2010 10:05:12	31.23827	121.487331	chin, river, Pearl Tower, Shanghai, bund
2	2	Museum	Shanghai Museum	3/24/2007 14:52:09	4/05/2007 18:05:02	31.23042	121.470871	china, colour, art, archaeology, museum
3	2	Bund	Shanghai Bund	3/23/2007 18:25:10	4/05/2007 18:10:32	31.24417	121.486985	river, asia, shanghai, chinese, Pudong, bund, Canada, good

Fig. 8. Sample records from anonymized photos metadata.

4.8.2. Travel sequences recommendations

Here, we discuss different methods to rank travel sequence patterns.

4.8.2.1. Frequent travel sequence patterns (FR). We simply rank all extracted travel sequence patterns in *SDB* by their frequencies, where frequency refers to the number of users visiting the sequence.

4.8.2.2. Context-aware significant travel sequence patterns recommendations (CSTR). The proposed method for the processing of context aware query $Q(t, w)$ for context-aware travel sequences proceeds as a two-step approach: an initial filtering step retrieves travel sequence patterns from *SDB* that belong to target city, and have higher probability to meet the contextual constraints given in the query (Eq. (9)), thus producing a filtered set of travel sequence patterns S'.

$$p(s) = \frac{1}{n}\sum_{i=1}^{n} p(l_i). \tag{9}$$

In the second step significance score for each travel sequence pattern $s \in S'$ is computed by aggregating the significance score of locations it contains as in Eq. (10).

$$r(s) = \frac{1}{n}\sum_{i=1}^{n} g(l_i) \tag{10}$$

Next, the travel sequence patterns in S' are ranked using score r and top k travel sequences are returned as query result.

5. Experimental evaluation

5.1. Data

5.1.1. Data acquisition

The public application program interface (API) of Flickr is used to collect metadata of 736,383 geotagged photos that were shared publicly and were taken in six cities of China between January 01, 2001 and July 1, 2011. Historical weather data of these cities is collected using public API of Wunderground (*wunderground.com*). Sample records from historical weather data and photos' metadata are given in Figs. 6 and 7.

5.1.2. Data pre-processing

We removed metadata of (1) Photos that were collected in the result of search based on text containing name of a city in their metadata, i.e., tags, title, description but their spatial context (latitude, longitude) did not match the geographical context of that city, and (2) Photos with incorrect temporal context. For example, a photo is removed whose upload time is identical to its taken

Table 3
Dataset summary.

Cities	Photos		Users	Tags
	Raw	Filtered		
Shanghai	252,768	230,566	80,530	244,221
Beijing	241,216	220,631	46,635	232,164
Hangzhou	37,267	28,312	1090	29,715
Chengdu	20,876	18,514	524	19,388
Guangzhou	18,796	17,141	507	18,474
Hong Kong	196,194	185,008	25,590	192,421

Table 4
Summary of extracted tourist locations in different cities, notations *V* and *Vt* are used to represent visits and visitors respectively.

Cities	Total locations	Locations distribution across visits			Locations distribution across unique visitor		
		$V <= 10$	$10 < V < 20$	$V >= 20$	$Vt <= 10$	$10 < Vt < 20$	$Vt >= 20$
Shanghai	492	160	138	194	250	110	132
Beijing	411	139	92	180	212	70	129
Hangzhou	128	43	32	53	63	35	30
Chengdu	52	17	13	22	32	9	11
Guangzhou	39	12	14	13	9	14	16
Hong Kong	413	140	122	151	227	60	126

time because Flickr assigns a default value to photo without taken time recorded by camera. As metadata of photos contain users' locations information, it can be anonymized e.g., assigning new IDs to users and photos replacing real ones, in order to avoid privacy issues. An anonymized sample of metadata is given in Fig. 8. Statistics about photos' metadata is given in Table 3.

5.2. Finding tourist locations

Table 4 summarizes the information regarding the popularity of locations based on unique number of visits and visitors. To detect locations from photos, for P-DBSCAN we set the value of $minPts = 50$ photos, $\varepsilon(epsilon) = 50$ meters and density ratio $\omega = 0.5$. To detect visits made by users to different locations from their contributed photos, we use value of visit duration threshold $visit_{thr} = 6$ hours.

5.3. User trips and travel sequence patterns generation

We use $trip_{dur} = 6$ hours to identify users' trips and set the minimum support threshold $min_{sup} = 2$ to mine as many trajectory patterns as possible for ranking. Fig. 9 shows the spatial distribution of photos, identified tourist locations and sample user trip in Beijing, China. Fig. 10 shows the sample frequent travel sequence patterns with different length extracted from the trips made by different users in Beijing.

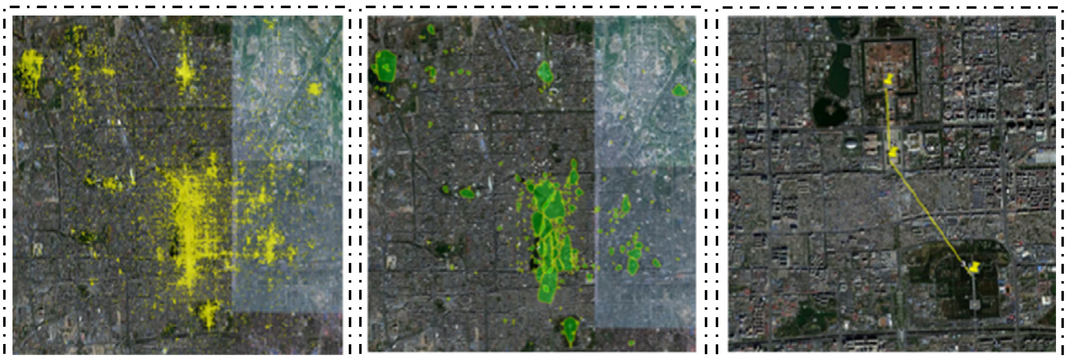
5.4. Tourist locations' recommendation

In this section, we describe the effectiveness of our proposed recommendation method in terms of prediction accuracy and ranking effectiveness. We explain our evaluation methodology and compare the results of different approaches.

5.4.1. Prediction accuracy

5.4.1.1. Metric. To measure the prediction accuracy, precision is used as a measure. Precision can be defined as the fraction of correct predictions in total number of predictions made.

5.4.1.2. Methodology. For evaluation, those users are selected that have visited at least 5 locations in test city C_t. We predict the locations actually visited by test user $u_p \in U$ in C_t. We use visits made by him to tourist locations in C_t to obtain; (1) number of relevant items (locations) denoted as k from total number of visits and (2) temporal and weather contexts associated with visits to build list of contextual constraints. We use these contextual constraints to filter the tourist locations by our context-



(a) (b) (c)

Fig. 9. In Beijing: (a) Spatial distribution of photos (b) Convex hull polygon on drawn to show the boundaries of aggregated tourist locations (c) A sample trip.

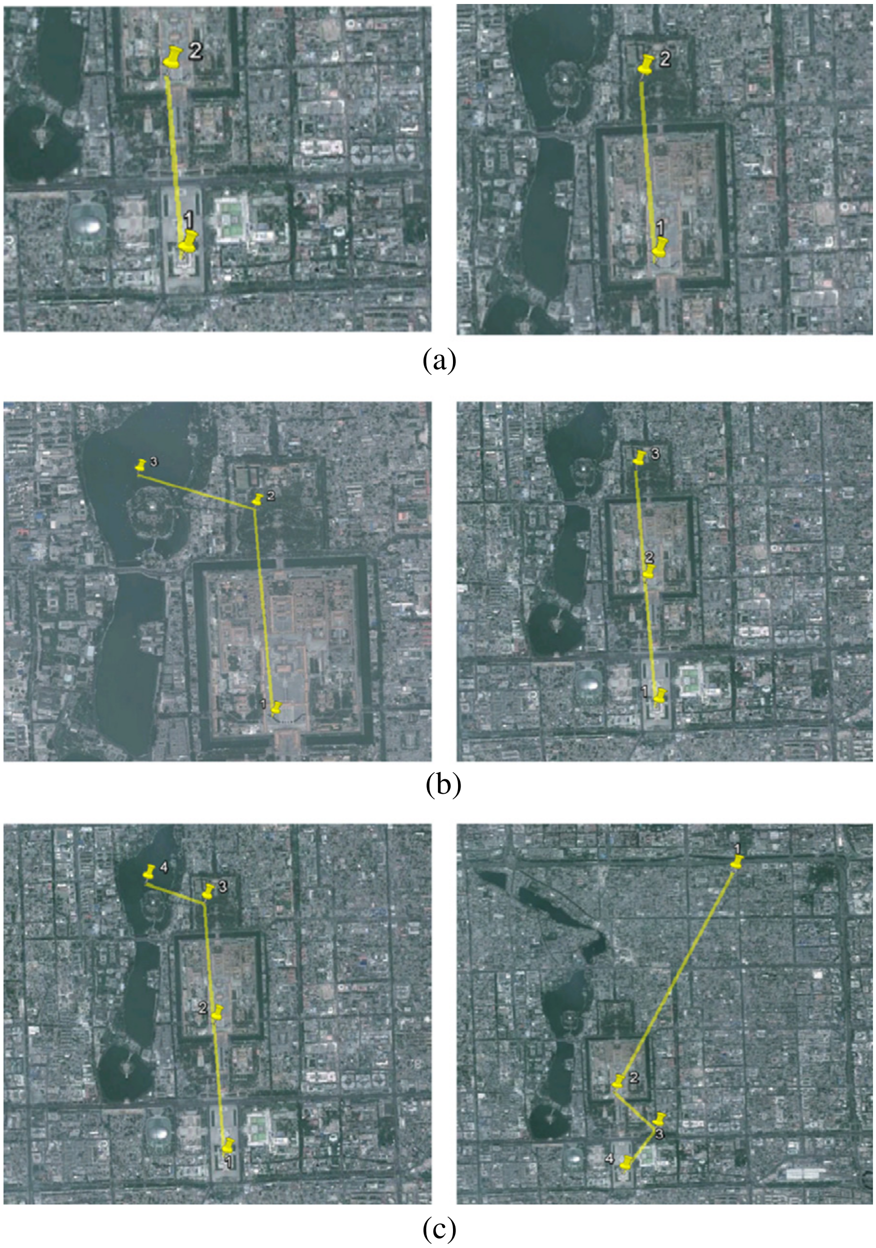


Fig. 10. Frequent travel sequence patterns in Beijing, (a) 2-length patterns, (b) 3-length patterns, and (c) 4-length patterns. (Best seen in colour.)

aware significant locations recommendation method. We recommend k number of ranked locations using our and baseline methods. To evaluate the performance of recommendation methods for user u_p, we match the recommended list with the actual list of locations visited by him in C_t.

5.4.1.3. Results. Fig. 11 reports the performance of prediction recommendation of our proposed context-aware significant tourist location ranking (CSR) and other methods in terms of precision (P). Popularity based ranking and CSR ranking give better prediction results as compared to collaborative filtering method, this might be because many users do not have single preferences but visit locations of many types and those locations which are popular and significant when they come to visit a new city. For this reason, evaluation method actually expects us to recommend these. We find that tourists comply more with the general travel preference and are therefore easier to predict by the popularity or significance based methods. Moreover, when we exploit context to filter the tourist locations for recommendations, it outperforms all other recommendations methods. It shows that we can increase the recommendation accuracy or relevancy if we consider the context in which the

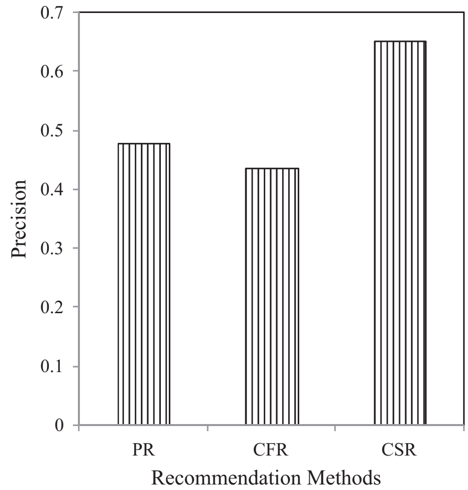


Fig. 11. Performance comparison in terms of P.

user is requesting and the context in which the locations have been frequently visited while addressing tourist locations recommendation query.

5.4.2. Ranking effectiveness

5.4.2.1. Metric. To measure the ranking effectiveness of different recommendation methods, we use Mean Average Precision (MAP@n). MAP@n is a widely used evaluation metric in information retrieval to measure the ranking effectiveness that is mean over the precision values after each correct recommendation in the top-n.

5.4.2.2. Methodology. To calculate MAP@50, we recommend 50 locations considering each visit made by each user test user in test city as a query and the location visited as one relevant item. We get average precision (AP) for each query $AP = 1/r$, where r is the position of relevant item in ranked list. We obtain the Mean Average Precision (MAP) using Eq. (11).

$$MAP = \frac{\sum_{i=1}^{N_q} AP_i}{N_q} \tag{11}$$

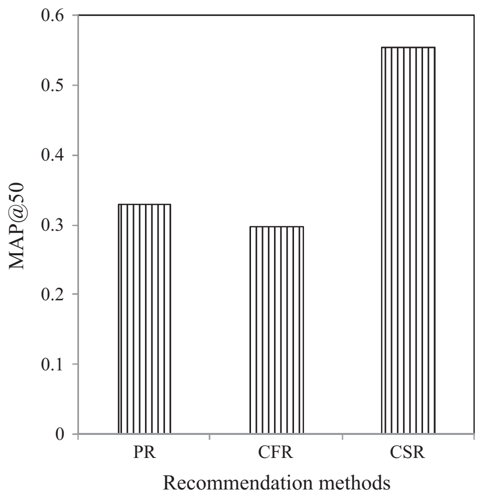


Fig. 12. Performance comparison in terms of MAP.

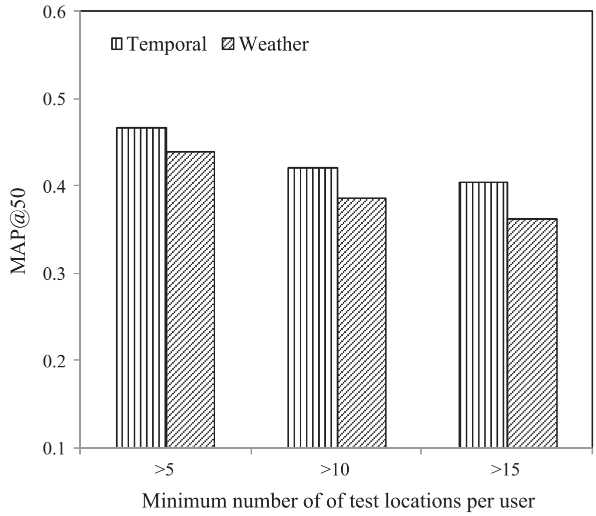


Fig. 13. Performance comparison in terms of mean average precision (MAP) with individual contexts.

where N_q is the total number of queries and AP_i is average precision for query i.

5.4.2.3. Results. Fig. 12 gives the performance of ranking ability of different ranking methods. Results obtained from experiments using metric MAP@50 shows that there is decent improvement in ranking ability of CSR method over other recommendation methods. The results also show that based on the criteria specified in terms of the minimum number of locations visited by each user in the test city, for number of 5 locations per user, the effectiveness of CSR ranking over popular ranking is 33%, and over personalized ranking is 29%.

The results of experiments, in terms of Precision, and Mean Average Precision, show considerable improvement of CSR over all other baseline methods and this improvement is statistically significant (based on paired t-test with $p < 0.05$).

5.4.3. Performance evaluation with individual contexts

Figs. 13 and 14 illustrate the performance of significant tourist location recommendation method with individual context factors, i.e., temporal context or weather context. We use precision (P) for prediction and MAP@50 to evaluate ranking ability.

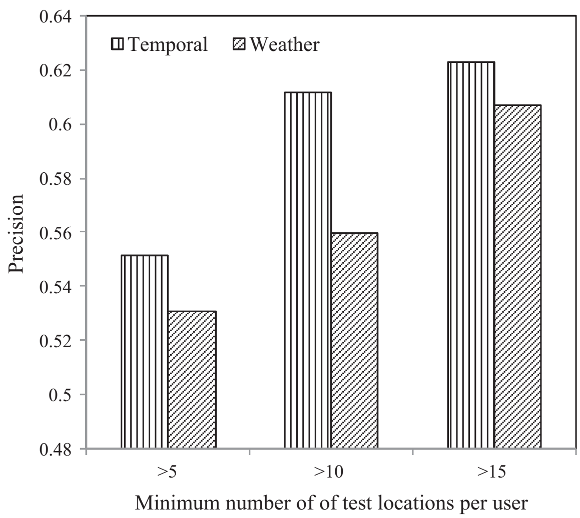


Fig. 14. Performance comparison in terms of Precision with individual contexts.

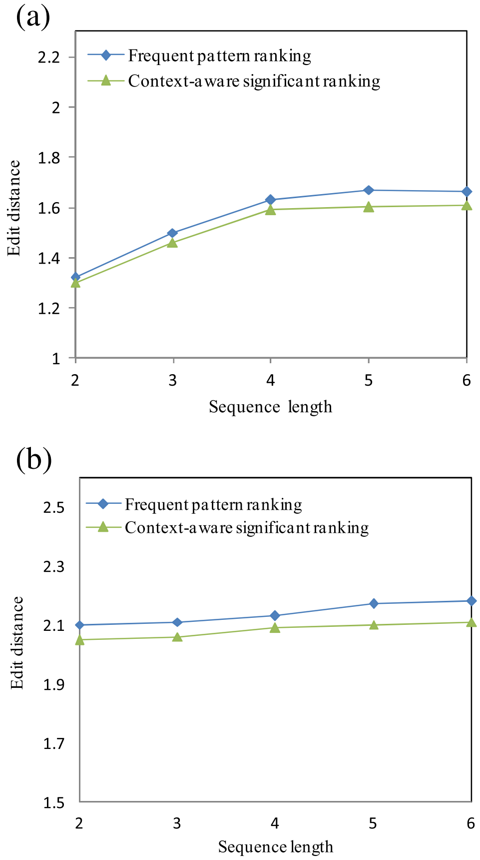


Fig. 15. Average edit distances of recommended tourist travel sequence pattern with different sequence length. Tourist locations identified using (a) ε (epsilon) = 100 meters, and (b) ε (epsilon) = 50 meters.

Results show that, exploitation of temporal context concepts produces better recommendation results as compared to weather context concepts.

5.5. Ranking travel sequence patterns

This experiment evaluates the appropriateness and relevance of the recommended travel sequence patterns.

5.5.1. Appropriateness

5.5.1.1. Metric. we used edit Distance [40] to measure the accuracy of tourist locations sequence recommendation. Edit distance measures the distance between two sequences in terms of the minimum number of edit operations required to transform one sequence into the other. The allowable edit operations are insert into a sequence, delete from a sequence, and replace one tourist locations with another.

5.5.1.2. Methodology. We applied the edit distance as the evaluation metric to measure the difference between the recommended travel sequence pattern and each test trip of users taken from users trips database (*TDB*). For context-aware travel sequence recommendations, we get the current context of a user from visits he made to the first location of his trip.

5.5.1.3. Results. We compared the results of context-aware significant travel sequence patterns and frequent travel sequence patterns recommendation methods. Fig. 15 shows the performance of each method. The results show that context-aware significant travel sequence pattern recommendation method offers better accuracy (i.e., lower edit distance) as compared to frequent travel sequence patterns recommendation method ($p < 0.05$ paired t-test).

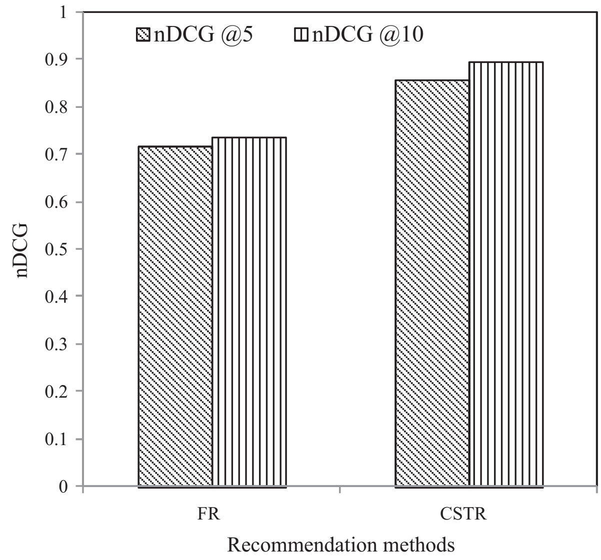


Fig. 16. Ranking relevance comparison.

5.5.2. Ranking relevance

5.5.2.1. Metric. We used nDCG [41] to measure the ranking effectiveness of recommendations. A higher nDCG value to a list of search results means that, the highly relevant items appearing earlier (with higher rank) in the result list. In particular, *nDCG @ p* measures the relevance of top *p* results.

5.5.2.2. Methodology. A large quantity of user queries were simulated with different contextual constraint settings. Four experts (tour guides) who are well familiar with these cities were requested to evaluate the output of recommendation methods and provide the feedback. They were asked to evaluate top 5 and top 10 travel sequence patterns using three scores, i.e., Very much interesting (3), interesting (2), neutral (1), not interesting (0).

The ratings obtained from an expert are used to generate the user's ideal ranking, which are compared against the top 5, and 10 location recommendations generated by CSTR and FR to obtain the nDCG value of the CSTR and FR recommendations.

5.5.2.3. Results. Fig. 16 reports the effectiveness of sequence recommendation methods in terms of normalized discounted cumulative gain (nDCG@5 and nDCG@10). Results show that, as compared to frequent travel sequence recommendations, context-aware significant travel sequence recommendations method made recommendations that are more relevant (paired t-test with $p < 0.05$).

6. Conclusions and future work

In this paper, we investigated the problem of mining significant tourist locations and tourist travel sequence patterns based on geo-tagged photos of users on social media site. Proposed method is capable of understanding context (i.e., time, date, and weather), as well as taking in to account the collective wisdom of people, to make tourist recommendations (i.e., tourist locations or travel sequence pattern). We illustrated our technique on a sample of public Flickr data set. Experimental results demonstrated that the proposed approach is able to generates better recommendations as compared to other state-of-the-art landmark recommendation methods.

This work demonstrated the exploration of the use of geo-referenced user generated content (UGC) to provide location-aware tourist information. UGC and other found data may not be "scientific" (in a narrow sense), but there is a wealth of these data available. A conceptual foundation is laid down for the analysis of spatio-temporal data of places obtained from community contributed geo-tagged photos' collection. It can also be utilized by local authorities, service providers, and tourist agencies for building user centric applications and to provide location based services.

This work motivates a number of important directions for further research. One modification which could be in future versions for better recommendation is to introduce some space-time constraints to the recommendation results, i.e., how long it would take to reach the sites, how much time is necessary to visit, and how much time the user has? We also anticipate that the investigation of personalization in combination with context awareness in trips (sequence of locations) for tourist recommendations is valuable. Here, personalization may refer to the types of locations visited, to the trips length or to the visit rhythm.

Acknowledgments

This work was funded by the Ministry of Industry and Information Technology of China (No. 2010ZX01042-002-003-001), China Knowledge Centre for Engineering Sciences and Technology (No. CKCEST-2014-1-5), the Natural Science Foundation of China (Nos. 60703040 and 61332017), the Science and Technology Department of Zhejiang Province (Nos. 2007C13019, 2011C13042, and 2013C01046).

References

[1] J. Hagenauer, M. Helbich, Mining urban land-use patterns from volunteered geographic information by means of genetic algorithms and artificial neural networks, Int. J. Geogr. Inf. Sci. 26 (6) (2012) 963–982.
[2] J. Luo, D. Joshi, J. Yu, A. Gallagher, Geotagging in multimedia and computer vision – a survey, Multimed. Tools Appl. 51 (2011) 187–211.
[3] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, C. Yu, Automatic construction of travel itineraries using social breadcrumbs, Proceedings of the 21st ACM Conference on Hypertext and Hypermedia, HT '10, ACM, New York, NY, USA, 2010, pp. 35–44.
[4] G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys '08, ACM, New York, NY, USA, 2008, pp. 335–336.
[5] A. Karatzoglou, X. Amatriain, L. Baltrunas, N. Oliver, Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering, Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, ACM, New York, NY, USA, 2010, pp. 79–86.
[6] Z. Yin, L. Cao, J. Han, J. Luo, T.S. Huang, Diversified trajectory pattern ranking in geo-tagged social media, Proceeding of SDM'11, 2011, pp. 980–991.
[7] A. Popescu, G. Grefenstette, Deducing trip related information from flickr, Proceedings of the 18th international conference on World wide web, WWW '09, ACM, New York, NY, USA, 2009, pp. 1183–1184.
[8] X. Lu, C. Wang, J.-M. Yang, Y. Pang, L. Zhang, Photo2trip: generating travel routes from geo-tagged photos for trip planning, Proceedings of the international conference on Multimedia, MM '10, ACM, New York, NY, USA, 2010, pp. 143–152.
[9] K. Okuyama, K. Yanai, A travel planning system based on travel trajectories extracted from a large number of geotagged photos on the web, The era of interactive media, Springer, New York, 2013, pp. 657–670.
[10] A. El Ali, S.N. van Sas, F. Nack, Photographer paths: Sequence alignment of geotagged photos for exploration-based route planning, Proceedings of the 2013 Conference on Computer Supported Cooperative Work, CSCW '13, ACM, New York, NY, USA, 2013, pp. 985–994.
[11] Y. Sun, H. Fan, M. Bakillah, A. Zipf, Road-based travel recommendation using geo-tagged images, Computers, Environment and Urban Systems, 2013.
[12] L. Bermingham, I. Lee, Spatio-temporal sequential pattern mining for tourism sciences, Procedia Comput. Sci. 29 (2014) 379–389.
[13] G. Cai, C. Hio, L. Bermingham, K. Lee, I. Lee, Mining frequent trajectory patterns and regions-of-interest from flickr photos, System Sciences (HICSS), 2014 47th Hawaii International Conference on, 2014, pp. 1454–1463.
[14] I. Lee, G. Cai, K. Lee, Exploration of geo-tagged photos through data mining approaches, Expert Syst. Appl. 41 (2) (2014) 397–405.
[15] Y. Zheng, L. Zhang, X. Xie, W.-Y. Ma, Mining interesting locations and travel sequences from gps trajectories, Proceedings of the 18th international conference on World wide web, WWW '09, ACM, New York, NY, USA, 2009, pp. 791–800.
[16] V.W. Zheng, Y. Zheng, X. Xie, Q. Yang, Collaborative location and activity recommendations with gps history data, Proceedings of the 19th International Conference on World Wide Web, WWW '10, ACM, New York, NY, USA, 2010, pp. 1029–1038.
[17] H. Yoon, Y. Zheng, X. Xie, W. Woo, Social itinerary recommendation from user-generated digital trails, Pers. Ubiquit. Comput. (2011) 1–16.
[18] S. Basu Roy, G. Das, S. Amer-Yahia, C. Yu, Interactive itinerary planning, Proceedings of the 2011 IEEE 27th International Conference on Data Engineering, ICDE '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 15–26.
[19] L. Chen, M. Lv, Q. Ye, G. Chen, J. Woodward, A personal route prediction system based on trajectory data mining, Inf. Sci. 181 (7) (2011) 1264–1284.
[20] X. Cao, G. Cong, C.S. Jensen, Mining significant semantic locations from gps data, Proc. VLDB Endowment 3 (1–2) (2010) 1009–1020.
[21] E. Agichtein, C. Castillo, D. Donato, A. Gionis, G. Mishne, Finding high-quality content in social media, Proceedings of the international conference on Web search and web data mining, WSDM '08, ACM, New York, NY, USA, 2008, pp. 183–194.
[22] J. Jeon, W.B. Croft, J.H. Lee, S. Park, A framework to predict the quality of answers with non-textual features, Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '06, ACM, New York, NY, USA, 2006, pp. 228–235.
[23] G. Cong, L. Wang, C.-Y. Lin, Y.-I. Song, Y. Sun, Finding question-answer pairs from online forums, Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '08, ACM, New York, NY, USA, 2008, pp. 467–474.
[24] D.J. Crandall, L. Backstrom, D. Huttenlocher, J. Kleinberg, Mapping the world's photos, Proceedings of the 18th international conference on World wide web, WWW '09, ACM, New York, NY, USA, 2009, pp. 761–770.
[25] S. Jain, S. Seufert, S. Bedathur, Antourage: mining distance-constrained trips from flickr, Proceedings of the 19th International Conference on World Wide Web, WWW '10, ACM, New York, NY, USA, 2010, pp. 1121–1122.
[26] S. Ahern, M. Naaman, R. Nair, J.H.-I. Yang, World explorer: visualizing aggregate data from unstructured text in geo-referenced collections, Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '07, ACM, New York, NY, USA, 2007, pp. 1–10.
[27] L. Kennedy, M. Naaman, S. Ahern, R. Nair, T. Rattenbury, How Flickr helps us make sense of the world: context and content in community-contributed media collections, Proceedings of the 15th international conference on Multimedia, MULTIMEDIA '07, ACM, New York, NY, USA, 2007, pp. 631–640.
[28] P. Serdyukov, V. Murdock, R. van Zwol, Placing flickr photos on a map, Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, SIGIR '09, ACM, New York, NY, USA, 2009, pp. 484–491.
[29] S. Kisilevich, F. Mansmann, D. Keim, P-dbscan: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos, Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research & Application, COM.Geo '10, ACM, New York, NY, USA, 2010, pp. 38:1–38:4.
[30] Y. Yang, Z. Gong, L.H. U, Identifying points of interest by self-tuning clustering, Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, SIGIR '11, ACM, New York, NY, USA, 2011, pp. 883–892.
[31] T. Rattenbury, N. Good, M. Naaman, Towards automatic extraction of event and place semantics from flickr tags, Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '07, ACM, New York, NY, USA, 2007, pp. 103–110.
[32] M. Clements, P. Serdyukov, A.P. de Vries, M.J. Reinders, Using flickr geotags to predict user travel behaviour, Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, SIGIR '10, ACM, New York, NY, USA, 2010, pp. 851–852.
[33] G. Abowd, C. Atkeson, J. Hong, S. Long, R. Kooper, M. Pinkerton, Cyberguide: A mobile context-aware tour guide, Wirel. Netw 3 (1997) 421–433.
[34] D. Shin, J.-W. Lee, J. Yeon, S.-G. Lee, Context-aware recommendation by aggregating user context, Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing, CEC '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 423–430.
[35] D. Lee, S. Park, M. Kahng, S. Lee, S.-G. Lee, Exploiting contextual information from event logs for personalized recommendation, in: R. Lee (Ed.), Computer and Information Science 2010, Vol. 317 of Studies in Computational Intelligence, Springer, Berlin Heidelberg, 2010, pp. 121–139.
[36] M. Kahng, S. Lee, S.-G. Lee, Ranking in context-aware recommender systems, Proceedings of the 20th International Conference Companion on World Wide Web, WWW '11, ACM, New York, NY, USA, 2011, pp. 65–66.
[37] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, Proceeding of Knowledge and Data Discovery (KDD '06), 1996, pp. 226–231.
[38] J. Pei, J. Han, B. Mortazavi-asl, H. Pinto, Q. Chen, U. Dayal, M. Chun Hsu, Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth, ICDE '01, IEEE Computer Society, Washington, DC, USA, 2001, pp. 215–224.

[39] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
[40] G. Navarro, A guided tour to approximate string matching, ACM Comput. Surv. 33 (1) (2001) 31–88.
[41] K. Järvelin, J. Kekäläinen, Ir evaluation methods for retrieving highly relevant documents, Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '00, ACM, New York, NY, USA, 2000, pp. 41–48.

Abdul Majid received the Ph.D degree in Computer Science from the College of Computer Science, Zhejiang University, Hangzhou, P. R. China in 2012. Currently he is working at Department of Computer Science & Technology, University of Azad Jammu & Kashmir, Muzaffarabad, Pakistan as an Assistant Professor. His research interests include Spatio-temporal data mining, pervasive computing and recommender systems.

Ling Chen received his B.S. and Ph.D degrees in computer science from Zhejiang University in Jun. 1999 and Mar. 2004, respectively. He joined Zhejiang University in 2004. Currently, he is an associate professor in the college of computer science. From Jan. 2006 to Jul. 2007, he was a research fellow in The University of Nottingham, UK. His research interests include ubiquitous computing, HCI, AI, pattern recognition, distributed systems, databases, and data mining. He has published over 50 works including recent papers in Information Sciences, Neurocomputing, GeoInformatica, IJGIS, PMC, IwC, IJHCI, UbiComp, CHI, AAAI, Multimedia, ICDE, CIKM, and WWW.

Hamid Turab Mirza received the B.S. degree in Computer Sciences from Hamdard University, Pakistan, in 2001, M.S. degree in Information Systems from The University of Sheffield, UK, in 2005, and Ph.D from Zhejiang University, Hangzhou, P.R.China in 2012. Currently he is associated with COMSAT Lahore Pakistan as an assistant Professor. Previously he has more than six years of research, teaching and information systems development experience in Aviation, Telecommunication and Education sectors of UK and Pakistan. His areas of research include CSCW, unstructured data management and information retrieval.

Ibrar Hussain received the M.S. degree in Information Management from The Queen Mary, University of London, UK, in 2006 and Ph.D degree from College of Computer Science at Zhejiang University, China in 2014. His areas of research include sptaio-temporal data mining, recommender systems, HCI, assistive technologies, and auditory interfaces for people with visual impairment.

Gencai Chen is a professor in the College of Computer Science, Zhejiang University, China. From 1987 to 1988, he studied in the Department of Computer Science, State University of New York at Buffalo, USA. He won the special allowance, conferred by the State Council of China in 1997. He is currently vice director of the Software Research Institute, Zhejiang University. His research interests mainly include database technologies, data mining, CSCW, speech emotion recognition. He has published over 80 papers on these areas in international conferences and journals.