11-2011

Towards Trajectory-Based Experience Sharing in a City

Byoungjip KIM

Youngki LEE
Singapore Management University, YOUNGKILEE@smu.edu.sg

SangJeong LEE

Yunseok RHEE

Junehwa SONG

Citation

KIM, Byoungjip; LEE, Youngki; LEE, SangJeong; RHEE, Yunseok; and SONG, Junehwa. Towards Trajectory-Based Experience Sharing in a City. (2011). *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks*. 85-88.

Available at: https://ink.library.smu.edu.sg/sis_research/2082

Towards Trajectory-Based Experience Sharing in a City

Byoungjip Kim[1], Youngki Lee[1], SangJeong Lee[1], Yunseok Rhee[2], Junehwa Song[1]

[1]Dept. of Computer Science
Korea Advanced Institute of Science and Technology (KAIST)
{bjkim, youngki, peterlee, junesong}@nclab.kaist.ac.kr

[2]Dept. of Digital Information Engineering
Hankuk University of Foreign Studies
rheeys@hufs.ac.kr

ABSTRACT

As location-aware mobile devices such as smartphones have now become prevalent, people are able to easily record their trajectories in daily lives. Such personal trajectories are a very promising means to share their daily life experiences, since important contextual information such as significant locations and activities can be extracted from the raw trajectories. In this paper, we propose MetroScope, a *trajectory-based real-time and on-the-go experience sharing system* in a metropolitan city. MetroScope allows people to share their daily life experiences through trajectories, and enables them to refer to other people's diverse and interesting experiences in a city. Eventually, MetroScope aims to satisfy users' ever-changing interest in their social environments and enrich their life experiences in a city. To achieve real-time, on-the-go, and personalized recommendation, we propose an approach of *monitoring activity patterns over people's location streams*.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – *Spatial databases and GIS*. H.3.5 [**Information Storage and Retrieval**]: On-line Information Services – *Data sharing*.

General Terms

Design, Experimentation, Human Factors.

Keywords

Location-based social networks, Experience sharing, Activity pattern monitoring.

1. INTRODUCTION

Social networking applications such as Facebook [2] and Twitter [5] have become one of the most important Web services, significantly changing the way people interact and consume news from each other. Recently, as location-aware mobile devices such as smartphones and tablet PCs have become prevalent, *location-based social networking applications* such as Foursqure [3], EveryTrail [1], and GeoLife [19] are emerging. In such location-based social networking applications, people want to share their location information and obtain reference information from other users' travel experiences. Many researchers expect that such location-based social networking applications will significantly enrich our daily lives by expanding the huge benefits of online social networks into our physical world.

Currently, the most popular types of location-based social networking applications are *location sharing applications* such as Foursquare [3] that allow people to simply share their current locations with their friends. However, in many cases, people want to share not only separate locations but also *whole experiences* in their daily lives. For example, while touring a city, people usually visit a sequence of places and want to share their experiences comprised of many activities happening at various places. More important, some people may want to recommend their significant traveling or dating experiences not only to friends but also to many other people. There are only a few *trajectory sharing applications* such as EveryTrail [1] and ShareMyRoutes [4]. However, these existing trajectory sharing applications are still insufficient to satisfy the abovementioned scenarios, mainly because they (1) share only raw GPS trajectories and (2) require people to search interesting trajectories through Web browsers by themselves. Even worse is that since the trajectory search function allows only coarse-grained search such as at the city level, it takes significant amount of time to find an interesting trajectory among many candidates, especially for on-the-go people.

In this paper, we propose MetroScope, a *trajectory-based real-time and on-the-go experience sharing system* in a metropolitan city. MetroScope allows people to share their daily life experiences such as hiking, touring, and dating by uploading their trajectories recorded by GPS- or Wi-Fi-enabled mobile devices to a sharing server. Basically, a trajectory is a series of location points with timestamps, but it can include additional contents such as photos or notes taken at some interesting locations. In MetroScope, trajectory-based experience sharing is performed through two main operations: *experience export* and *experience recommendation*. Experience export is a set of procedures that make a recorded personal trajectory public. Experience recommendation is a process to help people find other's relevant and interesting experiences.

To achieve real-time, on-the-go, and personalized recommendation, we propose the approach of *monitoring activity patterns over people's location streams*. The basic idea is that people will highly likely be interested in some experiences of other people who show similar activity patterns to them. For example, a young woman who is dating with her boyfriend (e.g., eating at a restaurant and watching a movie at a cinema) would be more interested in other nice and new dating courses rather than a city tour experience for foreigners (e.g., visiting imperial palace and memorial hall). More specifically, MetroScope automatically extracts semantic information such as significant locations and activities from the uploaded trajectories, and generates a recommendation rule based on the extracted information. Basically, a recommendation rule is a combination of people's activity pattern and profile. Then, MetroScope finds people who match the recommendation rule, and recommends the associated experience to the matched people.

There are two main challenges in comprehensively realizing MetroScope. First, we should accurately extract significant locations or activities from trajectories provided by users. However, it requires sophisticated processing, since a raw trajectory usually includes many errors and it is often hard to

ACM LBSN '11, November 1, 2011. Chicago, IL, USA

accurately determine the visited places or real activities. Fortunately, many research efforts have been devoted to mining locations and activities from trajectories [6][18][14]. Basically, we exploit existing solutions to implement MetroScope. Second, we should efficiently monitor activity patterns from location streams to recommend relevant and interesting experiences to people in a real-time manner. However, such continuous monitoring of location streams is very challenging since there are a huge number of people to monitor and a large number of monitoring rules to execute in the system. Therefore, we argue that it is important to develop scalable activity pattern monitoring techniques such as continuous query indexing and complex event processing.

The rest of the paper is organized as follows. Section 2 briefly discusses some related work. Section 3 presents a system overview of MetroScope. Finally, Section 4 concludes this paper.

2. RELATED WORK

Location Mining from Trajectories. Identifying significant locations such as visit places from a trajectory is one of the most required techniques, and has been studied by many researchers. The main approach to this problem is to apply clustering algorithms such as k-means and density-based clustering to trajectories. Ashbrook et al. [6] proposed a variant of k-means clustering in order to cluster GPS trajectories taken over an extended period of time into meaningful locations at multiple scales. Zheng et al. [18] proposed Tree-Based Hierarchical Graph (TBHG) for modeling multiple users' location histories. Using a density-based clustering algorithm over multiple users' trajectories, they hierarchically cluster location points into some geospatial regions. Thus, the similar stay points from various users would be assigned to the same clusters on different levels.

Activity Mining from Trajectories. Identifying activities from trajectories is another important problem. It is more challenging than identifying locations, since activities inherently include the high level of uncertainty. The basic idea to handle the problem is that people's daily activities can be inferred from contextual information such as location, time, and duration. Generally, approaches to the problem can be classified into two: statistics-based and POI (Point-Of-Interest)-based approach. As a statistics-based approach, Liao et al. [14] proposed a relational Markov network (RMN) which is an extension of conditional random fields (CRF) for inferring daily activities from GPS trajectories. However, statistics-based approaches require a large set of data enough to build a stochastic model. Unlike the statistics-based approach, POI-based approaches use the predefined POIs to identify activities. Xie et al. [17] proposed semantic join that identifies a sequence of activities by associating the closest POIs to a trajectory.

Continuous Query Indexing. The query indexing technique is essential for scalable continuous query processing [15][10][11]. Prabhakar et al. [15] proposed a query indexing technique for scalable evaluation of multiple continuous static range queries on moving objects. As opposed to traditional indexing of moving objects, such query indexing approach reduces the cost of maintaining the index by minimizing the number of index updates, as queries are not added/removed as frequently as object move. Lee et al. [10][11] proposed a BMQ-index for efficiently processing border-crossing monitoring queries. The BMQ-index consists of two linked lists for each dimension, and each list is a set of projected borders of monitoring regions to each dimension. By aggregating search results for each dimension, it can efficiently detect border crossing events of moving objects.

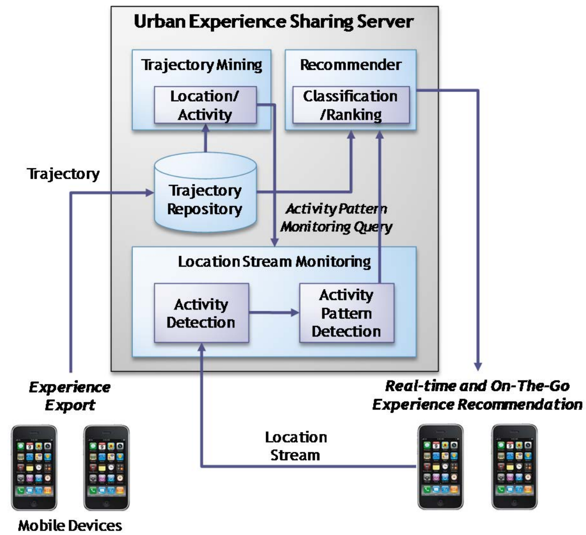


Figure 1. MetroScope system architecture.

Complex Event Processing. Complex event processing (CEP) systems [7][16][12] have been proposed for monitoring complex event patterns in diverse application domains such as RFID management, sensor networks, environment monitoring, and stock monitoring. The CEP systems provide event operators such as sequence, conjunction (AND), and disjunction (OR) that detect composite events by connecting primitive events. Wu et al. [16] proposed a NFA (Non-deterministic Finite Automata)-based algorithms named SASE for efficiently matching an event sequence over a large number of event sources. Unlike NFA-based approaches, recently, Lee et al. [12] proposed a network-based algorithm, i.e., Event-centric Composition Queue (ECQ). Using a shared processing technique that shares common event queues, the ECQ can efficiently process a large number of event monitoring queries.

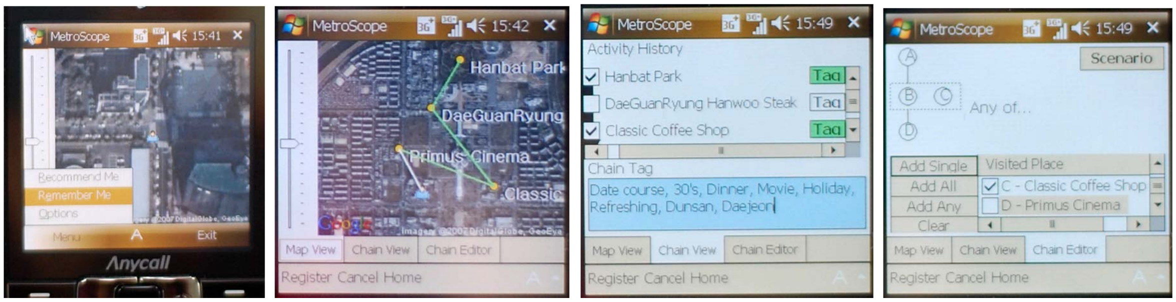
3. SYSTEM OVERVIEW

3.1 Key Concepts

MetroScope is a *trajectory-based experience sharing system*. Unlike the existing location sharing applications such as Foursquare [3], it aims to allow people to share their daily life experiences through trajectories, and take reference information on other people's diverse and interesting experiences in a city. Trajectory-based experience sharing is composed of two main operations: *experience export* and *experience recommendation*. There would be diverse requirements for such a trajectory-based experience sharing system. Among them, we emphasize that *real-time*, *on-the-go*, and *personalized* experience sharing is important.

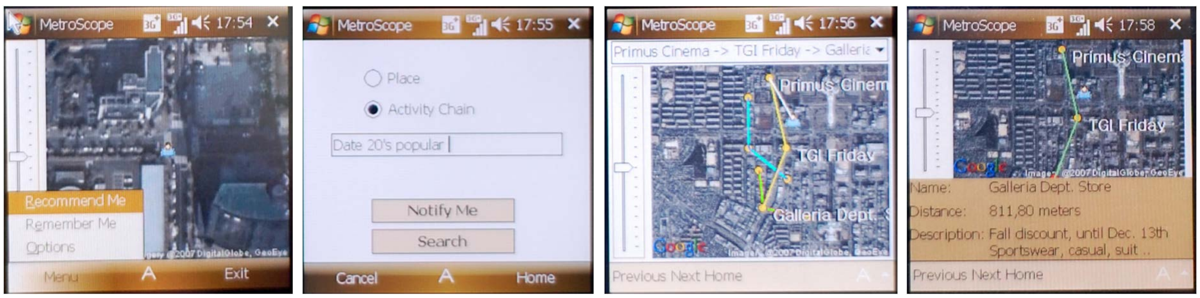
Experience Export. Experience sharing begins with experience export. Experience export is a set of procedures that make a recorded personal trajectory public. MetroScope allows people to upload their trajectories recorded by mobile devices to a central sharing server. However, uploading just raw GPS trajectories are insufficient. For intelligent experience sharing, semantic information such as significant locations and activities should be automatically extracted from the uploaded trajectories. Such extracted semantic information is used for finding people who will be highly likely interested in the trajectory.

Experience Recommendation. To help people find other's relevant and interesting other people's experiences, MetroScope supports experience recommendation. We argue that *real-time*,



(a) Home view (b) Location history (c) Experience description (d) Recommendation rule

Figure 2. Screenshots of MetroScope's experience export.



(a) Home view (b) Service mode (c) Recommended experiences (d) Experience details

Figure 3. Screenshots of MetroScope's experience recommendation.

on-the-go, and *personalized* recommendation is important in upcoming mobile computing era. Unlike PC era, in many cases, people using mobile devices want to take reference information on the surrounding area in a real-time manner, while they are on-the-go. For example, young dating couples often want to know some nice and new places to have dinner or to have fun, while hanging around a downtown. Also, many people often prefer to take personalized information such as gourmets' Thai restaurant recommendation than general information such as the most popular restaurant in a downtown. To achieve real-time, on-the-go, and personalized recommendation, we propose to monitor activity patterns over location streams reported from people who want to get recommended.

3.2 System Architecture

The system architecture of MetroScope is shown in Figure 1. As shown in the figure, MetroScope consists of a number of Mobile Clients and a Sharing Server. In the following subsections, we further provide the design and implementation of MetroScope.

3.2.1 MetroScope Mobile Client

The MetroScope Mobile Client is a location-aware mobile device such as smartphones, and plays as an agent of a user for the experience export and recommendation. For the experience export, the MetroScop Mobile Client is responsible for recording a user's trajectories and uploading them to a MetroScope Sharing Server when the user request. For the experience recommendation, the MetroScope Mobile Client is responsible for sending a user's location stream to the MetroScope Sharing Server and receiving relevant and interesting experiences for the user. To clearly deliver how people export their experience and get recommended, we provide prototype implementation as follows. For the prototype implementation, we use a smart phone, SAMSUNG blackjack 2 (SCH-M480) running Windows Mobile 6. It is equipped with a Marvell PXA310 (624MHz) processor, 3G/WiFi network interfaces, and a GPS receiver. The prototype is implemented in C# and runs on Microsoft .Net Compact Framework 2.0.

Figure 2 illustrates how the MetroScope Mobile Client helps a user to export her experience. First, the Mobile Client shows the current location of a user on a map in the home view (Figure 2-(a)). In the home view, a user starts to export her experience by selecting "Remember Me" in the menu (Figure 2-(a)). Then, the user's location history is automatically extracted and visualized on a map (Figure 2-(b)). Meanwhile, in the "Chain View" tab, the user can edit her experience by selecting which locations to export and making a note of additional information on each location. Also, the user can make a note that describes the overall experience to export (Figure 2-(c)).

Figure 3 describes how the MetroScope Mobile Client helps a user to get recommended with other people's experiences. A user can start experience recommendation by selecting "Recommend Me" in the menu of the home view (Figure 3-(a)). Then, the user can select one among two recommendation modes, i.e., "Notify Me" for proactive recommendation continuously updated by the server and "Search" for one time recommendation (Figure 3-(b)). In case of "Search", the user can input some keywords. In the "Notify Me" mode, the MetroScope Mobile Client continuously sends location data to the MetroScope Sharing Server. Then, the server can observe activity patterns of the user and continuously provide appropriate recommendations based on the patterns. Recommended experiences provided by the server are visualized on a map along with the city dynamics information (Figure 3-(c)). The sequence of circles presents a recommended experience, and each circle points a place and an activity. The size of a circle represents the crowdedness of a place. The bigger a circle is, the more crowded a place is. Also, the popularity of the experience is presented by the thickness and color of lines. If a user selects one of the experiences in the drop-down list, the details of the experience are shown on a map along with corresponding description (Figure 3-(d)).

3.2.2 MetroScope Sharing Server

The MetroScope Sharing Server plays as an orchestrator that enables the real-time and on-the-go experience sharing among a large number of people. As shown in Figure 1, the MetroScope Sharing Server is responsible for (1) mining the metadata such as significant locations and activities from the trajectories uploaded from users, (2) monitoring activity patterns over many people's location streams, and (3) recommending interesting experiences to relevant people according to their activity patterns detected by the monitoring unit. Among these functions, monitoring activity patterns over people's location streams is the key technique.

The MetroScope Sharing Server recommends relevant and interesting experiences as follows. First, the Sharing Server extracts semantic information such as significant locations and activities from trajectories uploaded from the Mobile Clients. The extracted information is maintained in a XML file named *Trajectory-based Experience Description Language (TEDL)*. The TEDL effectively describes a series of activities extracted from a trajectory. Here, an activity includes location coordinates, place name, place category, star time, and end time. For example, Bob's dating experience can be represented as follows (for convenience, we provide the example in a natural language form, rather than in a XML form): A1) drinking a cup of coffee at Starbucks from 2:00 p.m. to 2:30 p.m., A2) watching a movie at a multiplex from 3:00 p.m. to 5:00 p.m., A3) having dinner at T.G.I Friday from 6:00 p.m. to 8:00 p.m. The extracted semantic information is used for generating a monitoring query in the next step. Second, the Sharing Server associates the exported experiences with an activity monitoring query. And then, it starts to monitor activity patterns over people's location streams to find people who may be relevant and interested in the associated experience. If the Sharing Server finds people who match the associated monitoring query, it sends the experience to those people.

Query 1:

RECOMMEND Experience.title = "Bob's Dating" **TO** People
FROM Experience, People, PeopleProfile
PATTERN People **==** **SEQ**(**ACT**("Starbucks", 30min, 60min),
ACT("Cinema", 60min, 180min))
WHERE People.id = PeopleProfile.id
AND PeopleProfile.age >= 20
AND PeopleProfile.age < 30

Data schema:

- Experience<id, title, path(trajectory), path(TEDL), path(AQL)>
- People<id, location_x, location_y, timestamp>
- PeopleProfile<id, age, gender>

Activity Pattern Monitoring. For activity pattern monitoring, we propose a SQL-like query language, i.e., *Activity Pattern Monitoring Query Language (AQL)* [13]. Query 1 is an example monitoring query. It can be used to recommend the Bob's dating experience to people who have been drinking a cup of coffee at Starbucks and then watching movie at a multiplex, and is in their 20s. As shown in Query 1, the PATTERN clause specifies an activity pattern to monitor.

MetroScope first detects primitive activities from continuously incoming location streams, and then detects activity patterns by composing the primitive activities. In the AQL, we develop a new operator, **ACT**. The **ACT** operator is specified as **ACT**(*place_id*, *min_time*, *max_time*), where *place_id* is the identifier of a monitoring place, *min_time* and *max_time* are the minimum staying time and maximum staying time at the place specified by *place_id*, respectively. The operator detects whether a person stays for a specified amount of time (i.e., more than the *min_time* and less than the *max_time*) at a given place, while monitoring the person's location stream. The basic idea of using **ACT** operator is that people's daily activities can be effectively detected by location, time, and duration. The details can be found in our previous work [13]. It is similar to the POI-based activity mining from trajectories discussed in Section 2, since it exploits predefined POIs to detect activities. However, our approach is significantly different from those, since we apply the basic idea to detect activities over continuously incoming location streams. We adopt *event operators* (i.e., sequence (SEQ), conjunction (AND), disjunction (OR), etc) from the existing complex event processing (CEP) systems (i.e., Sentinel (a.k.a. Snoop) [7] and SASE [16]). Event operators connect primitive or composite events together to form new composite events. We developed *scalable activity pattern monitoring techniques* based on our previous work [8][9] [10][11][12][13].

4. CONCLUSION AND FUTURE WORK

In this paper, we proposed MetroScope, a *trajectory-based real-time and on-the-go experience sharing system* in a metropolitan city. We plan to further study other important issues of experience sharing systems. The issues may include activity detection accuracy, activity monitoring scalability, recommendation performance, location privacy, etc.

5. REFERENCES

[1] EveryTrail. http://www.everytrail.com.

[2] Facebook. http://www.facebook.com.

[3] Foursquare. http://foursquare.com.

[4] ShareMyRoutes. http://www.sharemyroutes.com.

[5] Twitter. http://www.twitter.com.

[6] D. Ashbrook, and T. Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing, vol. 7, no. 5*, 2003.

[7] S. Chakravarthy, V. Krishnaprasad, E. Anwar, and S.-K. Kim. Composite events for active databases: Semantics, Contexts and Detection. In *Proc. of VLDB*, 1994.

[8] K. Cho, I. Hwang, S. Kang, B. Kim, J. Lee, S. Lee, S. Park, Y. Rhee, and J. Song. HiCon: a hierarchical context monitoring and composition framework for next generation context-aware services. *IEEE Network, Vol. 22, No. 4*, 2008.

[9] B. Kim, S. Lee, Y. Lee, I. Hwang, Y. Rhee, and J. Song. Mobiiscape: Middleware support for scalable mobility pattern monitoring of moving objects in a large-scale city. *Journal of Systems and Software, Volume 84, Issue 11*, 2011.

[10] J. Lee, Y. Lee, S. Kang, S. Lee, H. Jin, B. Kim, and J. Song. BMQ-Index: shared and incremental processing of border monitoring queries over data streams. In *Proc. of MDM*, 2006.

[11] J. Lee, S. Kang, Y. Lee, S. Lee, and J. Song. BMQ-Processor: a high-performance border crossing event detection framework for large-scale monitoring applications. *IEEE Transactions on Knowledge and Data Engineering (TKDE), Vol. 22, No. 2*, 2009.

[12] S. Lee, Y. Lee, B. Kim, K. S. Candan, Y. Rhee, and J. Song. High-performance composite event monitoring system supporting large number of queries and sources. In *Proc. of ACM DEBS*, 2011.

[13] Y. Lee, S. Lee, B. Kim, J. Kim, Y. Rhee, and J. Song. Scalable activity-travel pattern monitoring framework for large-scale city environment. *To appear in IEEE Transactions on Mobile Computing (TMC)*.

[14] L. Liao, D. Fox, and H. Kautz. Location-based activity recognition using relational Markov networks. In *Proc. of IJCAI*, 2005.

[15] S. Prabhakar, Y. Xia, D. V. Kalashnikov, W. G. Aref, S. E. Hambrusch. Query indexing and velocity constrained indexing: scalable techniques for continuous queries on moving objects. *IEEE Transactions on Computer, vol. 51, no. 10*, 2002.

[16] E. Wu, Y. Diao, and S. Rizvi. High-performance complex event processing over streams. In Proc. of SIGMOD, 2006.

[17] K. Xie, K. Deng, and X. Zhou. From trajectories to activities: a spatio-temporal join approach. In *Proc. of ACM LBSN*, 2009.

[18] Y. Zheng, L. Zhang, X. Xie, and W. Y. Ma. Mining interesting locations and travel sequences from GPS trajectories. In *Proc. of WWW*, 2009.

[19] Y. Zheng, X. Xie, and W. Ma. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull., 33(2):32-40*, 2010.