Proceeding of the 4th International Workshop on Location-Based Social Networks (LBSN 2012)

Sept 8, 2012 - Pittsburgh, Pennsylvania, USA

Held in conjunction with Ubicomp 2012

General Chair:

Jason Hong, Carnegie Mellon University

Program Chair:

Yu Zheng, Microsoft Research Asia, China

ACM ISBN 978-1-4503-1224-0/12/09

Microsoft Research

Carnegie Mellon University

acm

UBICOMP 2012

Steering Committee

Jason Hong, Carnegie Mellon University

Christian S. Jensen, Aarhus University, Denmark

Wang-Chien Lee, Pennsylvania State University

Wen-Chih Peng, National Chiao Tung University

Xing Xie, Microsoft Research Asia (China)

Yu Zheng, Microsoft Research Asia, China

Xiaofang Zhou, University of Queensland (Australia)

Program Committee:

Licia Capra, University College of London

Tanzeem Choudhury, Cornell University, USA

Xin Chen, NAVTEQ, USA

Jing (David) Dai, IBM T.J. Watson, USA

Takahiro Hara, Osaka University, Japan

Pan Hui, Deutsche Telekom

Marek Kowalkiewicz, SAP Research, Singapore

WeiShinn Ku, Auburn University, USA

Nicholas Lane, Microsoft Research Asia, China

Wang-Chien Lee, Pennsylvania State University

Janne Lindqvist, WINLAB / Rutgers University

Cecilia Mascolo, University of Cambridge, UK

Wei Pan (Media Lab, MIT, USA)

Wen-Chih Peng, National Chiao Tung University, Taiwan

Daniel Gatica-Perez, IDIAP, Switzerland

Kazutoshi Sumiya, University of Hyogo, Japan

Xueyan Tang, Nanyang Technological University, Singapore

Vincent S. Tseng, National Cheng Kung University, Taiwan

Xing Xie, Microsoft Research Asia, China

Qiang Yang, Hong Kong University of Science and Technology, China

Joy Zhang, CMU Silicon Valley, USA

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.

ACM 978-1-4503-1224-0/12/09.

The Preface of the 4th International Workshop on Location-Based Social Networks

Yu Zheng
Microsoft Research Asia
Beijing, China
yuzheng@microsoft.com

Jason Hong
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
jasonh@cs.cmu.edu

ABSTRACT
We briefly introduce the 4th international workshop on location-based social networks (LBSN 2012), describing its objective, importance, and results.

Author Keywords
Location-based social networks, LBSN 2012, UbiComp 2012.

AIMS AND SCOPE
Social networks have been prevalent on the Internet and become a hot research topic attracting many professionals from a variety of fields. The advances in location-acquisition and mobile communication technologies empower people to use location data with existing online social networks in a variety of ways. For example, users can upload location-tagged photos to a social networking service such as Flickr, comment on an event at the exact place where the event is happening (for instance, in Twitter), share their present location on a website (such as Foursquare) for organizing a group activity in the real world, record travel routes with GPS trajectories to share travel experiences in an online community (for example GeoLife [1][2]), or log jogging and bicycle trails for sports analysis and experience sharing (as in Bikely).

The dimension of location helps bridge the gap between the physical world and online social networking services [3]. For example, a user with a mobile phone can leave her comments with respect to a restaurant in an online social site (after finishing dinner) so that the people from her social structure can reference her comments when they later visit the restaurant. In this example, users create their own location-related stories in the physical world and browse other people's information as well. An online social site becomes a platform for facilitating the sharing of people's experiences.

UbiComp' 12, Sep 5 – Sep 8, 2012, Pittsburgh, USA.

Furthermore, people in an existing social network can expand their social structure with the new interdependency derived from their locations [4][5][6]. As location is one of the most important components of user context, extensive knowledge about an individual's interests, behaviors, and relationships with others can be learned from her locations [7][8][9]. For instance, people who enjoy the same restaurant can connect with each other. Individuals constantly hiking the same mountain can be put in contact with each other to share their travel experiences [4]. Sometimes, two individuals who do not share the same absolute location can still be linked as long as their locations are indicative of a similar interest, such as beaches or lakes [6].

These kinds of location-embedded and location-driven social structures are known as location-based social networks, formally defined as follows [10][11]:

A location-based social network (LBSN) does not only mean adding a location to an existing social network so that people in the social structure can share location-embedded information, but also consists of the new social structure made up of individuals connected by the interdependency derived from their locations in the physical world as well as their location-tagged media content, such as photos, video, and texts. Here, the physical location consists of the instant location of an individual at a given timestamp and the location history that an individual has accumulated in a certain period. Further, the interdependency includes not only that two persons co-occur in the same physical location or share similar location histories but also the knowledge, e.g., common interests, behavior, and activities, inferred from an individual's location (history) and location-tagged data.

In a location-based social network, people can not only track and share the location-related information of an individual via either mobile devices or desktop computers [8], but also leverage collaborative social knowledge learned from user-generated and location-related content, such as GPS trajectories and geo-tagged photos. One example is determining this summer's most popular restaurant by mining people's geo-tagged comments. Another example could be identifying the most popular travel routes in a city based on a large number of users'

geo-tagged photos [19]. The city dynamics can also be modeled with the social media generated by a large number of users [20][21]. Consequently, LBSNs enable many novel applications that change the way we live, such as travel planning [12][13], location recommendations [5][13][14] [15], friend suggestion [5][9], activity suggestion [16][17] [18], event detection, and community discovery, while offering many new research opportunities, including link prediction, human mobility modeling, and user activity recognition , computer human interaction, and privacy [22].

TOPICS OF INTEREST

Topics of interest include but not limited to the following:

Understanding users in LBSNs

- User preference modeling
- User mobility modeling and analysis
- Real-world user activity sensing and recognition
- User similarity computing based on locations
- Link prediction and social tiers inference
- Friend recommendations and community discovery
- Expert discovery and influential person identification
- User intension understanding

Understanding locations in LBSNs

- Hot spots, significant places, and interesting locations detection
- Generic or personalized location recommendations
- Popular travel routes discovery from social media
- Trip planning and itinerary suggestion for users
- Location annotation and semantic meaning identification
- Location prediction and location privacy
- Anomaly detection and event discovery from social media
- Trajectory data mining in LBSNs

Information sharing in LBSNs

- Location and location-related data sharing
- Location and location-tagged media visualization
- Human-computer interaction in LBSNs
- Information retrieval in LBSNs.

Results

LBSN 2012 was held in Sept. 8 2012, in conjunction with UbiComp 2012 at Pittsburg, USA. Over 40 people participated in LBSN 2012. We received 19 submissions from 10 countries and regions. Each submission was assigned to three PCs for a peer review. As a result, we accepted 6 full oral papers and 9 short-presentation papers. The acceptance rate of full paper is about 31.6%. All the accepted papers will be included in ACM Digital Library, having the same length of up to 8 pages. A few quality full presentation papers will be invited to the special issue on urban computing in ACM Transaction on Intelligent Systems and Technology. The accepted papers were organized into four sessions: Privacy and Location Prediction, Topics and Events in LBSNs, Understanding user behavior in LBSNs, and Recommendations in LBSNs.

Organizers

Dr. **Yu Zheng** is a researcher from Microsoft Research Asia. He is an senior member of both IEEE and ACM. His research interests include trajectory data mining, location-based social networks, and urban computing. He has published over 70 referred papers at international conferences and journals, such as SIGMOD, KDD, AAAI, ICDE, WWW, Ubicomp, IEEE TKDE, and ACM TWEB. These papers have been featured by top-tier presses like MIT Technology Review multiple times. He has received 3 best paper awards respectively from UIC'10, ACM SIGSPATIAL GIS'11, and ADMA'11, as well as 1 best paper nominee from Ubicomp'11. He has written two book chapters and edited one book as an editor-in-chief. He has been invited to over 30 prestigious international conferences as a chair or program committee member, including ICDE, KDD, Ubicomp, IJCAI, ACM SIGSIAPTAL, ACM MM, PAKDD, and SSTD, etc. He is also an editorial board of 4 international journals and is a guest editor of ACM Transaction on Intelligent Systems and Technology. So far, he has supervised over 30 visiting Ph.D. students from around the world. He has received 3 technical transfer awards from Microsoft and 20 granted/filed patents. In 2008, he was recognized as the Microsoft Golden Star. He joined MSRA in July 2006 right after received his Ph.D. degree in communication & information systems from Southwest Jiaotong University.
Homepage: http://research.microsoft.com/en-us/people/yuzheng/

Jason Hong is an associate professor in the Human Computer Interaction Institute, part of the School of Computer Science at Carnegie Mellon University. He works in the areas of ubiquitous computing and usable privacy and security. He is also an author of the book The Design of Sites, a popular book on web design using web design patterns. Jason is also a co-founder of Wombat Security Technologies, which focuses on the human side of computer security. Jason received his PhD from Berkeley and his undergraduate degrees from Georgia Institute of Technology. Jason is also an Alfred P. Sloan Research Fellow.
Homepage: http://www.cs.cmu.edu/~jasonh/

REFERENCES

1. Yu Zheng, Yukun Chen, Xing Xie, Wei-Ying Ma. GeoLife2.0: A Location-Based Social Networking Service. In proceedings of International Conference on Mobile Data Management, IEEE press, 2009.

2. Yu Zheng, Xing Xie, and Wei-Ying Ma, GeoLife: A Collaborative Social Networking Service among User, location and trajectory, in IEEE Data(base) Engineering Bulletin, IEEE, June 2010

3. Justin Cranshaw, Eran Toch, Jason Hong, Niki Kittur, Norman Sadeh. Bridging the Gap Between Physical Location and Online Social Networks. In Proceedings of The Twelfth International Conference on Ubiquitous Computing (Ubicomp 2010).

4. Quannan Li, Yu Zheng, Xing Xie, and Wei-Ying Ma, Mining user similarity based on location history, in ACM SIGSPATIAL GIS 2008, ACM, 2008

5. Yu Zheng, Lizhu Zhang, Zhengxin Ma, Xing Xie, and Wei-Ying Ma, Recommending friends and locations based on individual location history, in ACM Transaction on the Web, ACM, 2011.

6. Xiangye Xiao, Yu Zheng, Qiong Luo, Xing Xie. Inferring Social Ties between Users with Human Location History. Journal of Ambient Intelligence and Humanized Computing.

7. Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, Wei-Ying Ma. Understanding Mobility Based on GPS Data. In Proceedings of UbiComp'08, ACM Press.

8. Yang Ye, Yu Zheng, Yukun Chen, Jianhua Feng, Xing Xie. Mining Individual Life Pattern Based on Location History. In proceedings of MDM'09. IEEE, 1-10.

9. Jason Wiese, Patrick Kelley, Lorrie Cranor, Laura Dabbish, Jason Hong, John Zimmerman. Are you close with me? Are you nearby? Investigating social groups, closeness, and willingness to share. In Proceedings of The Thirteenth International Conference on Ubiquitous Computing (Ubicomp 2011)

10. Yu Zheng. Location-based social networks: users. In Computing with Spatial Trajectories. Yu Zheng and Xiaofang Zhou, Eds. Springer 2011. ISBN: 978-1-4614-1628-9.

11. Yu Zheng. Tutorial on Location-Based Social Networks. WWW2012.

12. Hyoseok Yoon, Yu Zheng, Xing Xie, and Woontack Woo, Social Itinerary Recommendation from User-generated Digital Trails, in Personal and Ubiquitous Computing, Springer Verlag, 10 May 2011

13. Yu Zheng, Xing Xie, and Wei-Ying Ma, Mining Interesting Locations and Travel Sequences From GPS Trajectories, in WWW 2009, ACM, 2009

14. Yu Zheng, Xing Xie. Learning Location Correlation from GPS trajectories. In proceedings of the International Conference on Mobile Data Management 2010, IEEE press. USA.

15. Yu Zheng and Xing Xie, Learning travel recommendations from user-generated GPS traces, in ACM Transaction on Intelligent Systems and Technology, ACM, January 2011.

16. Vincent W. Zheng, Yu Zheng, Xing Xie, and Qiang Yang, Towards mobile intelligence: Learning from GPS history data for collaborative recommendation, Artificial Intelligence, Elsevier, April 2012.

17. Vincent Wenchen Zheng, Bin Cao, Yu Zheng, Xing Xie, and Qiang Yang, Collaborative Filtering Meets Mobile Recommendation: A User-centered Approach, in AAAI 2010, ACM. 2010

18. Vincent Wenchen Zheng, Yu Zheng, Xing Xie, and Qiang Yang, Collaborative Location and Activity Recommendations With GPS History Data, in WWW 2010, ACM, 2010

19. Ling-Yin Wei, Yu Zheng, Wen-Chih Peng, Constructing Popular Routes from Uncertain Trajectories. 18th SIGKDD conference on Knowledge Discovery and Data Mining (KDD 2012).

20. Jing Yuan, Yu Zheng, Xing Xie. Discovering regions of different functions in a city using human mobility and POIs. 18th SIGKDD conference on Knowledge Discovery and Data Mining (KDD 2012).

21. Justin Cranshaw, Raz Schwartz, Jason Hong, and Norman Sadeh. The Livehoods Project: Utilizing Social Media to Understand the Dynamics of a City. ICWSM 2012.

22. Yu Zheng, Xiaofang Zhou. Computing with Spatial Trajectories. Springer, 2011. ISBN: 978-1-4614-1628-9

Table of Contents

Session 1: Privacy and Location Prediction

Session 2: Topics and Events in LBSNs

Session 3: Understanding user behavior in LBSNs

Session 4: Recommendations in LBSNs

We Know Where You Live: Privacy Characterization of Foursquare Behavior

Tatiana Pontes*, Marisa Vasconcelos*, Jussara Almeida*, Ponnurangam Kumaraguru†, Virgilio Almeida*

*Universidade Federal de Minas Gerais, Brazil
†Indraprastha Institute of Information Technology, India
*{tpontes,marisav,jussara,virgilio}@dcc.ufmg.br
†pk@iiitd.ac.in

ABSTRACT

In the last few years, the increasing interest in location-based services (LBS) has favored the introduction of geo-referenced information in various Web 2.0 applications, as well as the rise of location-based social networks (LBSN). Foursquare, one of the most popular LBSNs, gives incentives to users who visit (check in) specific places (venues) by means of, for instance, mayorships to frequent visitors. Moreover, users may leave tips at specific venues as well as mark previous tips as done in sign of agreement. Unlike check ins, which are shared only with friends, the lists of mayorships, tips and dones of a user are publicly available to everyone, thus raising concerns about disclosure of the user's movement patterns and interests. We analyze how users explore these publicly available features, and their potential as sources of information leakage. Specifically, we characterize the use of mayorships, tips and dones in Foursquare based on a dataset with around 13 million users. We also analyze whether it is possible to easily infer the home city (state and country) of a user from these publicly available information. Our results indicate that one can easily infer the home city of around 78% of the analyzed users within 50 kilometers.

Author Keywords

Location Prediction, Privacy, Foursquare

ACM Classification Keywords

K.4.1 Computing Milieux: Computers and society—*Public policy issues*

General Terms

Experimentation, Measurement

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.

INTRODUCTION

Online social networks (OSN), such as Facebook, Twitter and the recent Google+ , are currently very popular. Some reasons for their great popularity include the easiness at which users can communicate and share content at large scale, the opportunity for self-promotion, commercial interests, as well as the simple intent of socialization [16]. Thus, users share a lot of information about themselves including age, address, relationship status, photos, and topics of interests on OSNs.

Due to the increasing use of smart devices equipped with Global Positioning System (GPS), LBSs have become very prevalent, thus attracting the interest of the research community. They have also motivated the creation of LBSNs [20], which emerge with an additional attraction in relation to OSNs, namely, the association of geographical information with the shared data. Out of the various existing LBSNs, such as Gowalla and Brightkite, Foursquare[1] is currently one of the most popular ones. Its overall goal revolves around the location sharing while users accumulate awards for visiting specific places in the system. It has been recently reported that Foursquare already achieved over 20 million members, with a history of around two billion visits notified by users in places all over the world.[2]

In Foursquare, users can inform their friends about their current location through *check ins* which may be converted into virtual rewards as badges or mayorships if the user is a frequent visitor of the same venue. Besides this gamification aspect, Foursquare has massively invested into the recommendation aspect allowing users to leave notes (tips) for friends and other users about their experiences at specific places (venues). Users can also keep track of tips marking them as done or saving them in a to-do list.

Easy availability of information about the location of a user raises several concerns about privacy violation [18]. For instance, the information about one's location may facilitate inferences about her behavioral patterns and habits. For instance, in Foursquare, although check ins are shared only with the user's friends, the use of other features of the system, such as mayorships, tips and dones are publicly avail-

[1] http://foursquare.com
[2] http://mashable.com/2012/04/16/foursquare-20-million/

able to everyone. In other words, the information about the venue(s) where the user is mayor (if any) as well as all venues where she left a tip or marked a tip as done or to-do is available to anyone. This information may reveal a lot about a user. For instance, a mayorship at a specific venue means that the user is a frequent visitor of that venue, whereas a tip (or a done/to-do) implies a prior visit or intention to checking out the place in the future. Tipping patterns may ultimately reveal user habits and personal interests. Indeed, if one considers that writing a tip requires more effort from the user than simply doing a check in, it could be argued that the locations at which a user left tips are even stronger indications of places she actually visited than check ins.[3] Users may do check ins when they are traveling, far from home, to show their friends that they are enjoying different places.

In this paper, we analyze how users explore these publicly available features, notably, mayorships, tips and dones in Foursquare, and their potential as source of information leakage and privacy violation. More specifically, we provide a characterization of mayorships, tips and dones in Foursquare based on a large dataset we crawled containing information on more than 13 million users and 15 million venues. As a first step towards investigating how much information about a user can be inferred from her tips, dones and mayorships, we analyze whether one can easily infer the home city (state and country) of a user from these publicly available information, by simply taking the location of the majority of the venues the user is connected to via mayorships, tips and dones. Note that the home city in Foursquare user profile is not a mandatory field and appear as an open text field. Thus, a user may choose not to reveal her home city by simply writing an invalid city name or even leaving it blank. Recent analyses of the location field in Twitter have pointed out that 34% of the users did not provide real locations, often including fake locations or even sarcastic comments. One of the reasons that justifies this user behavior may be to avoid unwanted messages that, for instance, may use the location information to provide a more efficient targeted advertisement mechanism. The question that we address here is: *despite being a private data that the user may choose not to reveal, can we still infer the home city of a user in Foursquare from her mayorships, tips and dones?*

We note that the literature contains several models for predicting user's home city mainly, exploiting the contents of user messages [1, 8, 11] or location of their friends [6]. Focused mainly on Twitter, these prior efforts aim at improving personalized services [1], performing targeted regional advertisements [19] or even detecting major events [15]. Instead, we here focus on a different application, Foursquare, exploiting different publicly accessible features, as our intention is to investigate their potential as source of inference of information about the user.

RELATED WORK

The increasing popularity of LBSNs have attracted researchers towards the awareness of location data. A number of recent studies have focused on geographically referenced information addressing aspects such as understanding why users share their location [16], human mobility patterns [2, 3, 14], user profile identification [10, 17], event detection [15] and analysis of a city urban development through check ins [5].

The information sharing in LBSNs and online social networks in general also raises concerns about exposure of user private data, touching privacy related issues. For instance, some studies have shown that it is possible to infer user implicit data through explicit information shared in such systems [7]. Mislove *et al.* have shown that users' personal interest can be inferred from friends [12], specially because, as argued in [4], people with common preferences are more likely to be friends. Other studies focused on assessing how users face privacy related issues and which strategies they often adopt to manage their exposure in the system [9].

There have also been studies that investigate whether it is possible to infer a user's location through other features which contain implicit location information. In [6], the home location of Twitter users are inferred from friends, with the simple assumption that users tend to have friends that live near them. In [1], Cheng *et al* estimated the user home city using the content of tweets with the assumption that people who live nearby do have a similar vocabulary. Other studies use machine learning approaches to infer user home location exploiting tweets' textual content [8] or users' tweeting behavior [11]. Unlike these previous studies, we here focus on inferring user's home city in a very popular LBSN (Foursquare), exploiting publicly available features such as mayorships, tips and dones that are associated with location information. To our knowledge, no previous work has addressed this problem yet.

FOURSQUARE DATASET

In this section, we briefly review the main elements of Foursquare as well as the crawled dataset used in our experimental evaluation.

Foursquare: Background

Foursquare is currently the largest and the most popular LBSN where members can share their locations with friends and followers through *check ins*. Check ins are performed via devices with GPS when a user is close to specific locations (*venues*). Venues are pages in the system that represent real locations of a great variety of categories such as airports, hotels, restaurants, monuments or squares.

Foursquare has a playful aspect that gives incentives to users who share more locations. Thus, check ins can be accumulated and exchanged by *badges* and *mayorships*. Badges are like medals given to users who check in at specific venues or achieve some predefined number of check ins. A mayorship, in turn, is given to the user who was the most frequent visitor (in number of check ins) of a venue in the last 60 days. Venue mayors are often granted rewards, promotions, discounts or even courtesies by business and marketing managers who own the venue. Once a user becomes a mayor of a given venue, that mayorship will be listed in her history,

[3]Note that Foursquare allows a user to check in at a venue even if she is not near the corresponding physical location.

even if some other user later ousts her from that position. That is, each user maintains a history of all mayorships she conquered. Multiple mayorships at the same venue are listed only once in this history. Moreover, mayorships are not temporal referenced.

Users can post *tips* at specific venues, commenting on their previous experiences when visiting the corresponding physical places. Tips can also serve as feedback, recommendation or review to help other users choose places to visit. Examples of tips include the best option of a menu in a restaurant, the best place to have lunch in an airport, or even a complaint about a service. With a limitation of 200 characters, tips nourish the relationship between users and real businesses and may be a key feature to attract future visitors [17]. Each user has a history of all tips she posted, with associated venue and timestamp. When visiting a venues' page, after reading a previously posted tip, a user may mark it as *done* or *to-do*, in sign of agreement with the tip's content or intention to visit that location in the future, respectively. The history of mayorships as well as the list of tips and dones, along with corresponding venue and timestamp information, of a user are publicly available at the user's profile page.

Crawled Dataset

Our study is based on a large dataset collected from Foursquare using the system API. We crawled user profile data consisting of user type, user home city, list of friends, mayorships, tips, dones, total number of check ins, Twitter screen names and Facebook identifiers. Our crawler ran from August to October 2011, collecting a total of 13,570,060 users, which is a good approximation of the entire Foursquare community at the time of the crawling since, reportedly, the number of users registered in Foursquare was 10 million in June 2011, reaching 15 million in December of the same year [13]. Our dataset contains 10,618,411 tips, 9,989,325 dones and 15,149,981 mayorships at 15,898,484 different venues.

FOURSQUARE CHARACTERIZATION

In this section, we discuss characterization of Foursquare users, focusing on user attributes that are publicly available in the system API and are associated with geo-referenced information, i.e., home cities, mayorships, tips and dones. Recall that the user home city and the venue location are open text fields, whose validity is not enforced by the system. Indeed, they may carry noise and invalid locations. Thus, we start our study by analyzing the amount of valid location information in our dataset. Next, we analyze the use of tips, dones and mayorships, focusing on the distribution of associated locations around the globe. Finally, we perform a temporal and spatial analysis of user activities in terms of tipping and marking previous tips as done.

Location Information in Foursquare

We here discuss the location information available in public attributes of Foursquare users, i.e., in home city, mayorships, tips and dones.[4] Since mayorships, tips and dones are associated with venues, we here analyze the user home city and the venue location attributes in our dataset.

User home city, in particular, is limited to 100 characters and is not required to be filled. It is expected that users provide the name of the city where they live, although the system provides neither rule to enforce it nor any automatic tool to help users filling the field (e.g., a predefined list of cities from which the user can choose one). Thus, users are free to provide this location information at various granularities, ranging from specific addresses, to city, state and country names, or even regions of the planet (e.g., "North Pole"). We also observed some home city fields filled with emails, phrases, or even numbers in our dataset. Similarly, the location associated with a venue, and thus, indirectly, with mayorships, tips and dones of that particular venue, is also an open text field. Unlike the user home city, the address and the city of a venue must be filled before the venue is created. Moreover, it is necessary to set a pin in a map to update the venue's location. Once again users may choose to provide invalid addresses and city names, and mark arbitrary locations in the map.

Table 1. Dictionary. GI = geographic information. UHC = User Home City. VL = Venue Location.

Statistics	UHC	VL
# in dataset	13,570,060	15,898,484
# valid GI	13,299,112	11,683,813
# valid but ambiguous GI	359,543	2,868,636
# non-GI	244,233	4,214,671
# empty entries	26,715	0

Table 2. Quality of Geographic Information.

Quality	# Users	# Venues
Continent	107	61
Country	602,932	294,596
State	390,224	93,513
County	251,383	276,097
City	10,354,058	6,937,523
Neighborhood	981,139	1,060,124
Area of Interest/Airport	27,307	47,896
Street	326,751	95,543
Point of Interest	5,607	9,792
Coordinate	61	32

Thus, in order to standardize the home city and venue location fields, we created a dictionary of city names using the *Yahoo! PlaceFinder*, the Yahoo's geo-coding API.[5] This tool was used to verify the validity of the data in both fields. For a given query (text), the tool either returns some geographic data, in case the query consists of a valid location, or an error, otherwise. For queries consisting of valid locations, the tool's response depends on the "quality" of the query, which, in turn, is related to the spatial granularity (e.g., street, city, state, country) of the location information provided in the query. For instance, for a query "New York", *Yahoo! PlaceFinder* returns that the query's quality is at the granularity of city, and provides the corresponding geographic coordinates, a standardized city name as well as the state and country names. *Yahoo! PlaceFinder* may also identify locations at the finer granularity of streets. Moreover, note that the use of standardized city name allows us to

[4]Check ins are private, it is not possible to access the geographic location associated with them.

[5]http://developer.yahoo.com/geo/placefinder/

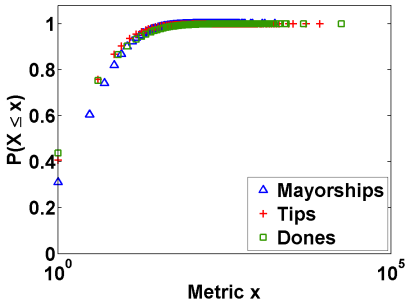


Figure 1. Cumulative Distribution of the Number of Mayorships, Tips and Dones per User.

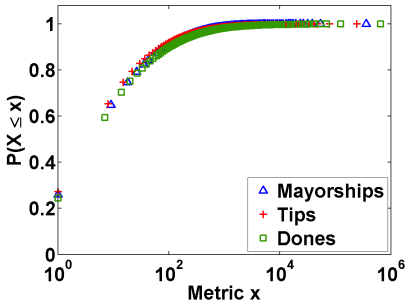


Figure 2. Cumulative Distribution of the Number of Mayorships, Tips and Dones per City.

uniquely identify the city, despite the existence of multiple name variations (e.g., NY, New York City, etc).

Table 1 provides some details about our dictionary, indicating the total number of users and venues with valid, invalid as well as empty location information. Note that, perhaps surprisingly, the vast majority (98%) of the users do provide valid locations, according to *Yahoo! PlaceFinder*, in their home city attributes, and only a tiny fraction of users leave this attribute empty (0.2%). The fraction of venues with valid locations is smaller (73.5%), but, also accounts for most venues in our dataset. We note that, for some queries, *Yahoo! PlaceFinder* returned multiple ambiguous answers reflecting alternative locations with the same name (e.g., there are ten cities named "Springfield" in the United States). We chose to disregard users and venues with ambiguous locations, which correspond to 2.7% and 24.6% of all users and venues with valid locations, respectively, in our dataset.

Next, we analyze the "quality", in terms of spatial granularity, of valid (unambiguous) locations associated with users and venues. In Table 2, we present the distributions of users and venues across 10 different quality levels, ranging from continent to specific coordinates. Note that, the majority of users and venues provide location information at the granularity of city or at finer granularities. Indeed, users and venues are associated with 100,629 different cities around the world. Note, however, that over 1.2 million users provide location information at a coarser granularity, often at the country level. Thus, the inference of the home city or even state of these users based on their mayorships, tips and dones will reveal private information.

Mayorships, Tips and Dones

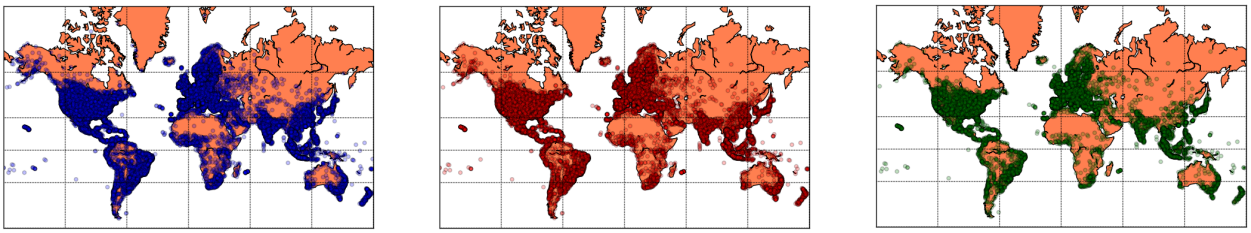
In this section, we analyze the mayorships, tips and dones of users in our dataset. Since our goal is to exploit the location of the venues associated with these attributes to infer the user home city, we start by showing an overview of the use of mayorships, tips and dones among users in our dataset. We observe that almost 4,2 million users, or around 30% of all users in our dataset, have at least one of these attributes. Out of these, around 1 million have only mayorships, 670 thousand have only tips and 367 thousand have only dones, whereas 890 thousand users have all three attributes. Thus, exploiting these attributes to infer a user home city is promising as the required information is available in a large fraction of all users. Moreover, as shown in Figure 1 and consistent with previous analyses of Foursquare [14, 17], the distributions of the numbers of mayorships, tips and dones per user are very skewed, with a heavy tail, implying that few users have many mayorships (tips or dones) while the vast majority have only one mayorship (tip or done). Indeed, for users that have one of these attributes, we find that 69% (59% and 56%) of the users have 2 or more mayorships (tips and dones).

Figure 2 shows the distributions of numbers of mayorships, tips and dones per city, considering only cities with at least one instance of the attribute. As shown, the distributions are also very skewed, with a few cities having as many as 100 mayorships, tips or dones.

Next, we analyzed the correlation between the number of mayorships, tips and dones per city. We found that there is a high correlation between the number of mayorships and the number of tips across cities, with a Spearman's correlation coefficient ρ [21] equal to 0.78. Similarly, the correlation is also high between the number of mayorships and the number of dones ($\rho = 0.72$). Moreover, we found that the cities with the largest numbers of mayorships tend also to have large numbers of tips and dones, although some interesting differences are worth noting. For instance, mayorships are more concentrated in Southeast Asia, in cities like Jakarta, Bandung and Singapore, which are the top three cities in number of mayorships, jointly having more than 500,000 mayorships. Tips, in turn, are concentrated in different locations around the Earth: the top three cities in number of tips are New York, Jakarta and São Paulo, with a total of 600,000 tips. Dones, on the other hand, tend to be concentrated in venues in the United States, in cities like New York, Chicago and San Francisco, which jointly received around 1 million dones.

We note that, although other studies [1, 8, 11] have exploited textual features to analyze user location, we here chose not to exploit the tip's content as they are often targeted towards more generic topics such as food and service quality. We observe that most words extracted from tips in our dataset are adjectives or are related to food, meal, services and generic places where one can eat or drink.

We now discuss the distribution of cities with venues where users have mayorships, tips and dones around the world.



(a) Mayorships. (b) Tips. (c) Dones.

Figure 3. Global Distribution of Mayorships, Tips and Dones.

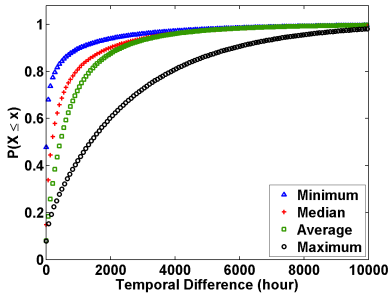


Figure 4. Cumulative Distribution of Time Interval Between Consecutive Tips/Dones Posted per User.

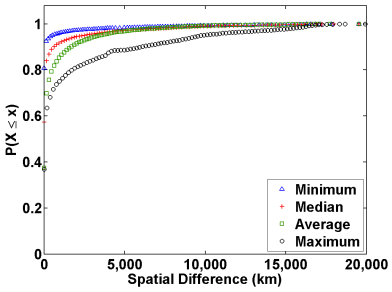


Figure 5. Cumulative Distribution of Displacements Between Consecutive Tips/Dones Posted per User.

We only consider attributes associated with venues that have valid cities (validated by *Yahoo! PlaceFinder*) as location. Figure 3 shows these distributions in maps of the globe, with each point representing a city with venues with at least one mayorship, tip or done.[6] As the maps show, Foursquare venues are spread all over the world, including remote places such as Svalbard, an archipelago in the Arctic Ocean, with coordinates (78.218590,15.648750). Moreover, all three maps are very similar, with most incidences of points in America, Europe and Southeast Asia. The distribution of mayorships, shown in Figure 3(a), is denser, with a total number of unique cities (79,194) much larger than in the distributions of tips and dones, which cover a total of 54,178 and 30,530 unique cities, respectively. The somewhat sparser tip map (Figure 3(b)) indicates that there are many cities, particularly in Canada, Australia, central Asia and Africa, where, despite the existence of venues and mayors, users do not post tips. The distribution of dones, shown in Figure 3(c), reveals an even sparser map, with most activity concentrated in touristic or developed areas, such as USA, western Europe and southeast Asia. We note that a similar map was produced for check ins in [2]. Besides both datasets were collected at different times, we can see that their main areas of concentration overlap.

Temporal and Spatial Analyses

We perform a temporal and spatial analysis of user activity in terms of tips and dones. Our goal is to analyze how often users leave tips / dones as well as how far users "go" between consecutive tips / dones. To that end, we make use of the timestamp associated with each tip and done as well as the location of the venue where the tip (or done) was left.

[6]The Antarctica continent was omitted because there was no point on it.

We start by investigating the frequency at which users leave tips and/or mark previous tips as done. We do so by analyzing the time interval between consecutive activities (be it a tip or a done) of the same user. Thus, we consider only users with at least two activities, covering a total of 1,959,647 users. We summarize user activity by the minimum, median, average and maximum inter-activity times. Figure 4 shows the cumulative distributions of these four measures computed for all considered users. We note that the distribution of minimum inter-activity times is very skewed towards short periods of time, with almost 50% of the users posting consecutive tips/dones 1 hour apart. However, on average, median and maximum, users do tend to experience very long periods of time between consecutive tips and dones. For instance, around 50% of the users have an average inter-activity time of at least 450 hours, whereas around 80% of the users have a maximum inter-activity time above 167 hours (roughly a week).

Next, we analyze the displacement between two venues visited in sequence by the user, as indicated by consecutive tips and/or dones of the user. For this analysis, we consider only users with at least two activities, provided that the venues associated with these activities have valid locations, with "quality" of city level or finer granularity. Our dataset contains almost 1.5 million users in this group. For these users, we computed the displacements between consecutive tips/dones by taking the difference between the coordinates of the associated venues. Once again, we summarize user activity computing the minimum, median, average and maximum displacement per user. Figure 5 shows the distributions of these measures for all analyzed users. Around 36% of the users have average and maximum displacements of 0 kilometer, indicating very short distances (within a few

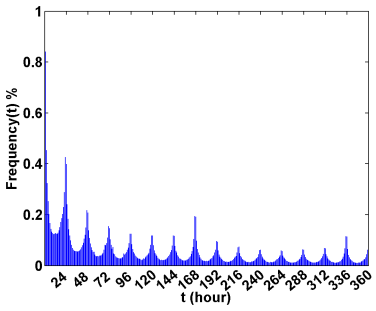


Figure 6. Distribution of Returning Times.

meters). Moreover, 70% of the users have an average displacement of at most 150 kilometers, which could be characterized as within the metropolitan area of a large city. Also 60% of the users have a maximum displacement of at most 100 kilometers, possibly the distance between neighboring cities. Thus, overall, consecutive tips/dones of a user are often posted at places near each other. However, there are exceptions. About 10% of the users have a maximum displacement of at least 6,000 kilometers.[7]

Finally, we analyze how often users return to the same venue for tipping or marking tips as done. That is, we analyze the returning times, defined as the time interval between consecutive tips/dones posted at the same venue by the same user. This analysis is focused on 813,607 users, who have at least two tips/dones in the same venue, and cover more than 3 million returns. We here choose to show the distribution of all measured returning times, as opposed to summarizing them per user first, so as to compare our results against previous findings of check in patterns [2]. Figure 6 shows the distribution, focusing on returning times under 360 hours, which account for 69.7% of all measured observations. The curve shows clear daily patterns with returning times often being multiples of 24 hours, which is very similar to the distribution of returning times computed based on check ins [2]. We note, however, that 50% of the measured returning times are within 1 hour, which cannot be seen in the Figure as its y-axis is truncated at 1% so that the rest of the curve could be distinguished. Moreover, out of these observations, 90% of them are at most 10 minutes. Thus, returning times, in general, tend to be very short. If we analyze the behavior per user (omitted more details, due to space constraints), we note that most users have very short minimum returning times, which is below 1 hour for 62% of the users. However, consistently with results in Figure 4, on average, median and maximum, users do tend to experience longer returning times. For instance, 52% of the users have average returning times of at least 168 hours.

INFERRING USER'S HOME LOCATION

In this section we investigate whether one can infer, with reasonable effectiveness, the location where a user lives based only on information that is publicly available on her Foursquare profile page, notably the lists of mayorships, tips and dones.

[7]Note that the maximum displacement between two points in the Earth is the distance between antipodes (two diametrically opposed points) that is about 20,000 kilometers.

We here discuss the inference approach and evaluation methodology adopted in Methodology section whereas our main results are discussed in Experimental Results section.

Methodology

The key assumption behind this work is that users tend to have mayorships, tips and dones in venues at the same location (e.g., city) where they live. At first, one might think that the mayorship locations are perhaps the strongest piece of evidence about a user's home location, as the former represent places the user possibly goes very often. Recall that a user only becomes mayor of a venue if she is the most frequent visitor in the last 60 days. However, tips may also reveal places where a user has been, since when posting tips users are often sharing experiences. [8] Finally, dones may also provide some evidence about a user's home location, although perhaps not as strong as tips and mayorships. Our conjecture is that users often mark as done tips about physical places where they have been to or intend to go soon. We note however that, despite intuitive, the aforementioned assumption is not guaranteed to hold for all users. As discussed in Temporal and Spatial Analyses section, 10% of the users in our dataset have a maximum displacement of at least 6,000 kilometers between consecutive tips and dones.

As a first step to address this question, we consider a simple approach that takes the most popular location among the attributes (mayorships, tips and/or dones) of a user as her home location, using a majority voting scheme. We note that more sophisticated methods could be applied such as classification algorithms (e.g., k-nearest neighbor) and other machine learning techniques [8, 11, 1]. Instead, we chose a simple majority voting approach as it allows us to assess the potential for effective inferences of this type in Foursquare.

We consider seven inference models which differ in terms of the attributes used for inference. The *Mayorship* model uses only the locations of the mayorships to infer the user's home location. Similarly, the *Tip* and *Done* models use only locations of tips and of dones, respectively. The *Mayorship+Tip*, *Mayorship+Done*, *Tip+Done* models use information from only two attributes, whereas the *All* model takes all three attributes jointly. By comparing alternative models, we are able to assess the potential of each attribute as source of inference. Moreover, recall that, as discussed in Mayorships, Tips and Dones section, there are non-negligible numbers of users that only have one or two of the attributes. Thus, the combination of multiple attributes may enable the inference for a larger user population. The models are here used mainly to infer the user's home city, although we also consider inferences about the user's home state and country.

To evaluate the effectiveness of each model, we take the information provided in the user's home city attribute as ground truth. Although users are free to enter whatever they want in this attribute, we found that the majority of Foursquare users do enter valid locations (see Table 1). To evaluate our in-

[8]Although users may post tips at unknown venues to, for instance, inquire about driving directions, operation time, or parking conditions, we believe that this does not occur very often.

Table 3. Home Location Inference.

Classes Distribution									
	Home City			**Home State**			**Home Country**		
Features	Class 0	Class 1	Class 2	Class 0	Class 1	Class 2	Class 0	Class 1	Class 2
Mayorship	727,179	847,876	239,129	707,953	913,166	110,110	727,179	1,053,703	33,302
Tip	725,073	671,576	192,781	702,583	727,219	99,672	725,073	835,532	28,825
Done	546,815	541,795	106,297	524,137	561,165	55,115	546,815	630,937	17,155
Mayorship+Tip	898,293	1,322,214	300,831	878,578	1,398,351	146,526	898,293	1,581,654	41,391
Mayorship+Done	825,009	1,213,917	270,974	805,029	1,278,784	130,439	825,009	1,447,581	37,310
Tip+Done	831,759	1,038,268	223,093	807,091	1,089,638	116,549	831,759	1,228,043	33,318
All	939,888	1,573,471	310,045	919,938	1,643,825	153,955	939,888	1,840,850	42,666
Accuracy									
	Home City			**Home State**			**Home Country**		
Features	Class 0	Class 1	Total	Class 0	Class 1	Total	Class 0	Class 1	Total
Mayorship	**51.61%**	**67.41%**	60.12%	**71.27%**	**80.92%**	**76.70%**	89.79%	92.92%	91.64%
Tip	51.52%	67.29%	59.11%	70.29%	80.59%	75.53%	**90.12%**	**93.67%**	**92.02%**
Done	50.09%	61.74%	55.89%	70.16%	78.38%	74.41%	89.12%	92.38%	90.87%
Mayorship+Tip	51.57%	66.24%	**60.31%**	70.21%	80.27%	76.39%	89.71%	93.13%	91.89%
Mayorship+Done	51.05%	65.27%	59.51%	70.01%	79.89%	76.07%	89.18%	92.78%	91.47%
Tip+Done	51.18%	64.16%	58.38%	69.76%	79.28%	75.23%	89.52%	93.04%	91.62%
All	51.46%	64.86%	59.85%	69.74%	79.53%	76.02%	89.29%	92.89%	91.67%

ferences, we consider only users whose home city attributes contain valid locations at the city level or at a finer granularity, as validated by *Yahoo! PlaceFinder*.

In our evaluation, we group users into three classes. *Class 0* consists of users who have a single activity, either a mayorship, a tip or a done. In this case, the unique choice is to set the user's home location equal to that of her activity. *Class 1* consists of users who have multiple activities with a predominant location across them. For these users, the inferred location matches the most often location of their activities. *Class 2*, in turn, consists of users with multiple activities in which there is no single location that stands out (i.e., there are ties). Our current inference approach cannot be applied to *Class 2* users.

Thus, we evaluate the proposed models by assessing their accuracy on users of both *Class 0* and *Class 1*. The accuracy corresponds to the percentage of correctly inferred locations out of all users of each class. Moreover, we also report the overall accuracy of each model, considering all users that are eligible for inference by the given model (i.e., users who have the required attributes).

Experimental Results

In this section, we present the experimental evaluation of our inference models. We start by discussing the results for inferring a user's home city, our main focus, discussing the inference of home state and country later in this section.

Table 3 shows, for each inference model, the number of users eligible for inference (i.e., users that have the required attribute) in each class (top of the table). It also shows the accuracy of the model for users in classes 0 and 1 as well as the overall accuracy considering all eligible users (bottom). We start by noting that the vast majority of the eligible users (87%- 91%) are in classes 0 and 1. Thus, for most users, either they have a single activity (33-45%) or they have multiple activities with a predominant location, and thus their home city can be inferred by our approach.

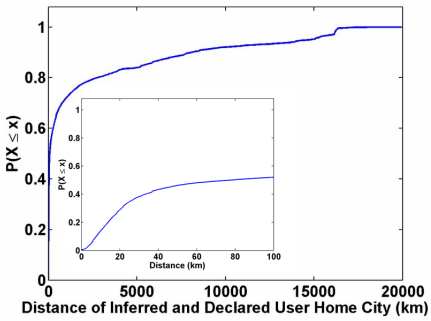


Figure 7. Cumulative Distribution of Distances Between Inferred and Declared User Home City.

We find that the models produce only marginally different results, in terms of accuracy, both per class and overall. As expected, mayorships are the best single attribute to infer home location, although, perhaps surprisingly, tips are only marginally worse. Dones, in turn, produce the worst results among the three attributes, when used in isolation. The combination of attributes does hurt the accuracy, in comparison with the *Mayorship* model, in most cases (*Mayorship+Tip* being the exception), possibly because tips and dones add some noise. However, note that, despite a somewhat lower accuracy, these combined models actually cover a much larger user population. For instance, the *Mayorship* model can only be applied to 1,814,184 users, whereas the *All* model is applicable to 2,823,404. Thus, considering the actual number of users for which each model was able to correctly predict the home city, we found that the best model was *All* (1,504,262 correct inferences) followed by *Mayorship+Tip* (1,339,152 correct inferences).

To better understand the models' errors, we computed for each incorrect inference the distance between the inferred city given by the *All* model and the declared user home city. Figure 7 shows the distribution of these distances. We found that around 46% of the distances are under 50 kilometers, which is a reasonable distance between neighboring (twin)

cities. Thus, combining these results with the correct inferences produced by our model, we find that we can correctly infer the city of around 78% of the users within 50 kilometers of distance.

We now turn our attention to the inference of a user's home state, whose results are also shown in Table 3. We note that, in comparison with the home city inference, all models improved for home state inference, reaching an overall accuracy around 75%. Once again, mayorships arise as the single attribute that produces the highest accuracy, for home state inference, followed by tips and dones. Nevertheless all models lead to very similar accuracies, both per class and overall. Thus, once again, due to the larger user coverage, the *All* model is able to correctly infer the home state of the largest number of users (1,948,851).

Finally, we also evaluate the models to infer a user's home country as a complementary analysis to validate our key assumption that users tend to have mayorships, tips and dones close to where he lives. As expected, Table 3 shows that all models achieve accuracies above 90% for home country inference. Unlike in the previous two cases, despite the great similarities in the results, the *Tip* model is the single attribute model that produces the best accuracy, followed by *Mayorship* and *Done*. The combined models produce very similar results, with *All* producing the largest number of correct inferences (2,549,177).

Our study presented satisfactory results in predicting user home location. Thus, an interesting implication of our work is that even the mispredictions may highlight some implicit user behavior in terms of mobility. At the city level, for instance, we observed some users that live nearby the inferred cities, which may indicate that they probably live in one place and move frequently to another. At the state level, the lower but non-negligible fraction of errors indicates that there are some users that have interstate mobility. Moreover, the inference of the home state may help disambiguate home cities, such as the case of Springfield. Finally, at the country level, we observed that there is a high concentration of the activities considered (mayorships, tips and dones) in the declared user home location. This can be verified by the higher accuracy that we obtained in our models. However, inference errors are still possible since some users may have his current home location outdated (e.g., a user who has just moved to another country) or may travel a lot around the world, or may even have a significant place-based identity with some city of another country (as discussed in [8]).

CONCLUSIONS AND FUTURE WORK

In this paper, we address the problem of privacy inference using publicly available features in Foursquare. Using a model that takes into account the majority of places where the user have interacted through mayorships, tips or dones, we are able to infer with high accuracy where the user current lives or his home location (city, state or country).

As future work, we plan to analyze the impact of differentiating features, e.g. giving weights, in the accuracy of our model. Also, we can explore more sophisticated machine learning approaches in attempt to increase our inference accuracy. Moreover, we plan to investigate other types of information that can be inferred using the same attributes.

Acknowledgements

This research is partially funded by the Brazilian National Institute of Science and Technology for Web Research (MCT/CNPq/INCT Web Grant Number 573871/2008-6), and the authors' individual grants from CNPq, CAPES and Fapemig.

REFERENCES

1. Z. Cheng, J. Caverlee, and K. Lee. You are Where You Tweet: a Content-based Approach to Geo-locating Twitter Users. In *Proc CIKM'10*.
2. Z. Cheng, J. Caverlee, K. Lee, and D. Sui. Exploring Millions of Footprints in Location Sharing Services. In *Proc AAAI ICWSM'11*.
3. E. Cho, S. Myers, and J. Leskovec. Friendship and Mobility: User Movement in Location-based Social Networks. In *Proc ACM SIGKDD'11*.
4. M. Choudhury, H. Sundaram, A. John, D. Seligmann, and A. Kelliher. "Birds of a Feather": Does User Homophily Impact Information Diffusion in Social Media? *CoRR'10*.
5. J. Cranshaw, R. Schwartz, J. Hong, and N. Sadeh. The Livehoods Project: Utilizing Social Media to Understand the Dynamics of a City. *ICWSM'12*.
6. C. Davis Jr., G. Pappa, D. Oliveira, and F. Arcanjo. Inferring the Location of Twitter Messages Based on User Relationships. *Transactions in GIS*, 15(6):735–751, 2011.
7. R. Gross and A. Acquisti. Information Revelation and Privacy in Online Social Networks. In *Proc WPES'05*.
8. B. Hecht, L. Hong, B. Suh, and E. Chi. Tweets from Justin Bieber's Heart: the Dynamics of the Location Field in User Profiles. In *Proc CHI'11*.
9. I.-F. Lam, K.-T. Chen, and L.-J. Chen. Involuntary Information Leakage in Social Network Services. In *Proc of IWSEC'08*.
10. N. Li and G. Chen. Sharing Location in Online Social Networks. *IEEE Network*, 2010.
11. J. Mahmud, J. Nichols, and C. Drews. Where Is This Tweet From? Inferring Home Locations of Twitter Users . In *Proc AAAI ICWSM'12*.
12. A. Mislove, B. Viswanath, K. Gummadi, and P. Druschel. You are Who You Know: Inferring User Profiles in Online Social Networks. In *Proc WSDM'10*.
13. S. M. News. http://www.socialmedianews.com.au/foursquare-reaches-15-million-users/.
14. A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. An Empirical Study of Geographic User Activity Patterns in Foursquare. In *Proc ICWSM'11*.
15. T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake Shakes Twitter Users: Real-Time Event Detection by Social Sensors. In *Proc WWW'10*.
16. K. Tang, J. Lin, J. Hong, D. Siewiorek, and N. Sadeh. Rethinking Location Sharing: Exploring the Implications of Social-Driven vs. Purpose-Driven Location Sharing. In *Proc UBICOMP '10*.
17. M. Vasconcelos, S. Ricci, J. Almeida, F. Benevenuto, and V. Almeida. Tips, Dones and ToDos: Uncovering User Profiles in Foursquare. WSDM '12.
18. C. Vicente, D. Freni, C. Bettini, and C. Jensen. Location-Related Privacy in Geo-Social Networks. *IEEE Internet Computing*, 2011.
19. T. Vgele, C. Schlieder, and C. Schlieder. Spatially- Aware Information Retrieval with Graph-Based Qualitative Reference Models. In *Proc FLAIRS'03*.
20. Y. Zheng. Location-based social networks: Users. In Y. Zheng and X. Zhou, editors, *Computing with Spatial Trajectories*, pages 243–276. Springer, 2011.
21. D. Zwillinger and S. Kokoska. *CRC Standard Probability and Statistics Tables and Formulae*. Chapman & Hall, 2000.

Improving Location Prediction Services for New Users with Probabilistic Latent Semantic Analysis

James McInerney, Alex Rogers, Nicholas R. Jennings
University of Southampton, Southampton, SO17 1BJ, UK
{jem1c10,acr,nrj}@ecs.soton.ac.uk

ABSTRACT
Location prediction systems that attempt to determine the mobility patterns of individuals in their daily lives have become increasingly common in recent years. Approaches to this prediction task include eigenvalue decomposition [5], non-linear time series analysis of arrival times [10], and variable order Markov models [1]. However, these approaches all assume sufficient sets of training data. For new users, by definition, this data is typically not available, leading to poor predictive performance. Given that mobility is a highly personal behaviour, this represents a significant barrier to entry. Against this background, we present a novel framework to enhance prediction using information about the mobility habits of existing users. At the core of the framework is a hierarchical Bayesian model, a type of probabilistic semantic analysis [7], representing the intuition that the temporal features of the new user's location habits are likely to be similar to those of an existing user in the system. We evaluate this framework on the real life location habits of 38 users in the Nokia Lausanne dataset, showing that accuracy is improved by 16%, relative to the state of the art, when predicting the next location of new users.

Author Keywords
Knowledge Representation and Reasoning::Geometric, Spatial, and Temporal Reasoning, Machine Learning::Data Mining, Machine Learning::Machine Learning (General/other, Reasoning under Uncertainty::Uncertainty in AI (General/other), Machine Learning::Unsupervised Learning

ACM Classification Keywords
H.5.2 User/Machine Systems: I.5 Pattern Recognition

General Terms
Experimentation, Performance.

INTRODUCTION

Location prediction of daily life mobility has been a topic of considerable interest in recent years [10, 5]. In general, location data is gathered about an individual user with global positioning system (GPS), cell tower or wifi data, and this history of locations is used to predict the user's future locations. Predicting user mobility gives the promise of many exciting real world ubiquitous services. Better mobile reminders, search results, and advertisements are likely to result from knowing where the user will be [10].

Existing approaches typically assume an adequate history of observations of user mobility in order to train a statistically accurate model of behaviour. Yet, in real life, we know that this will not usually be available for a new user. This presents an important barrier to the success of ubiquitous services. For any service to grow in its number of users, a high proportion of those users will necessarily be new. But performance of the service is at its worst (due to poor predictions) precisely at the time when the user has just started using it. Nevertheless, new users are important precisely because they often have the responsibility of deciding whether to commit to or abandon the service.

This problem suggests the need to exploit the similarities between new and existing users, an approach often used in recommender systems [12] and recently introduced in systems for activity recognition [8]. In the domain of mobility prediction, it is known that many people share a common set of mobility habits [5], such as going to work during weekdays, staying at home on weekend mornings, and going to new places in evenings. These similarities could be used to increase accuracy for new users. While this approach makes sense intuitively, it is hard to exploit in location prediction because it requires a semantic understanding of user location (e.g., home, work, sports centre, or restaurant). That is, a user's history of locations is made up of points in geographical space, but generalizing the habitual elements usually requires conversion to general and meaningful labels before behaviour correlation analysis can begin [5]. However, semantic labelling of locations is challenging to achieve, even for individuals with larger data sets.

Yet, arguably, deriving semantic labels for locations is an unnecessary requirement for the problem of learning user models of mobility. To get the benefits of accurate predictions, we only require that the states (i.e., the locations) of the new user's model be correctly linked to those of the much richer model of a similar established user. For example, if the habit of going to a certain train station to go home after work appears in a user's history, then this pattern may be used for prediction without explicitly knowing the meaning of the locations. Furthermore, if a similar pattern appears in the history of a new user, then linking their respective transport hubs, homes and places of work could enable richer predictions, such as a lower probability of going home from that station at weekends. In short, we can remain indifferent to

UbiComp '12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

the precise interpretation of locations because we primarily care about the dynamics of the model, which can be empirically accurate and generate good predictions even if the underlying semantic information (e.g., home, work) is missing.

To address this shortcoming, we present the first approach to model such *functional mappings* between existing and new users, to significantly improve location prediction for the latter group, without requiring any semantic labelling. In doing so, we make several contributions:

- We develop a hierarchical Bayesian model, based on probabilistic latent semantic indexing [7], for matching the locations of a new user with those of existing users. The model makes no cultural-specific assumptions about habits (such as going to work on weekdays) and uses no extra information about locations.
- We show how this mapping can be used to increase prediction accuracy for new users, as compared to a model that does not use the mapping. Specifically, it is possible to transform the transition matrix of a Markov model representing an established users' mobility, to a mobility model that approximates well the habits of the new user. In general, the benefit of inferring the mapping between the locations of new and established users is that it does not require commitment to any specific prediction model.
- We validate our approach using the location histories of 38 real world users from the Lausanne Nokia dataset. We simulate the arrival of new users to a location prediction system by truncating their location history and find that prediction accuracy is improved 16% relative to a state of the art predictor using our approach.

When taken together, these contributions open the way for improved performance for new users in predictive systems with a minimal amount of assumed knowledge.

In the remainder of this paper, we first introduce our model of functional mapping for locations in Section 2, illustrating how such mappings can be used to transform a Markov model from that of the established user to the new user (Section 3), and then applying it to the Nokia dataset to validate the approach (Section 4). Finally, Section 5 concludes.

MODEL OF FUNCTIONAL LOCATIONS

To formalise the observed temporal similarity between the mobility patterns of users, we assume a fixed number, T, of time slots, each with a probability distribution over L significant locations of the user. We represent the probability that a single user will be in significant location i at time t with a $T \times L$ probability matrix $\boldsymbol{M}$, responsible for generating the actual observations $\boldsymbol{X}$, of which we assume there are N. Hence, $\boldsymbol{X}$ is an $N \times L$ binary matrix, with one row for each observation. Clearly, the user can be in only one location at a time, resulting in a 1-of-L choice at each time step (i.e., for each row in $\boldsymbol{X}$). A natural assumption for such categorical variables is therefore that the probability distribution over the set of locations (i.e., presence) for each time slot t follows a multinomial distribution [3]. The *sufficient statistics* of the multinomial distribution, that is, the only information required from the raw observations, is the $T \times L$ matrix of integer counts, $\boldsymbol{C}$, representing the sum of presence counts for each time slot of the history $\boldsymbol{X}$, where element c_{ti} is the number of times the user was observed at location i at time slot t.

The likelihood of an observed history of presence counts $\boldsymbol{X}$ can therefore be found via the definition of a multinomial distribution [3], assuming that all observations are independently and identically distributed:

$$p(\boldsymbol{X}|\boldsymbol{M}) \propto \prod_{t=1}^{T} \prod_{i=1}^{L} m_{ti}^{c_{ti}} \tag{1}$$

The total number of observations (i.e., $N = \sum_{t=1}^{T} \sum_{i=1}^{L} c_{ti}$) will clearly be higher for an established user than for a new user. Our model attempts to address this disparity in the number of observations by explicitly linking the two users' $\boldsymbol{M}$ parameters. Throughout, we refer to the random variables specific to the established user (i.e., $\boldsymbol{X'}$, $\boldsymbol{C'}$ and $\boldsymbol{M'}$) with the apostrophe modifier, to distinguish them from those of the new user (i.e., $\boldsymbol{X}$, $\boldsymbol{C}$ and $\boldsymbol{M}$).

To address the disparity in the number of observations between the new and established user, we make the key assumption that the location behaviour of the new user is generated entirely from the probability distribution of the established user, *subject to some transformation of locations*. More formally, we assume that $\boldsymbol{M'R} = \boldsymbol{M}$, where $\boldsymbol{R}$ is an unknown transformation matrix. This work focuses primarily on inferring $\boldsymbol{R}$ from the observed location histories of both users, and using it to enhance prediction.

In more detail, $\boldsymbol{R}$ can be interpreted as the mapping of locations between two specific users, A and B. In general, there are two types of mapping. The first is one-to-many, in which user A's presence in a single location tends to co-occur with presence in multiple locations for user B. For example, user A may work in a single office, while user B may spend half her time working in a lab and the other half in the library. As another example, user A might tend to visit a single cafeteria for lunch, while user B might visit multiple sandwich shops nearby. The inverse relationship, many-to-one, is also possible. Multiple locations for user A can co-occur with just a single location for user B. Clearly, the one-to-one mapping, in which both users have a single location in which they tend to be at the same time (e.g., home) is just a special case of either of these transformations. Inferring this mapping goes beyond simply smoothing the probability densities of sparse presence observations [6] because it enables the reuse of rich densities from other real world users.

To find $\boldsymbol{R}$, the naïve derivation of the new user's probability distribution over locations uses $\boldsymbol{R}$ to directly transform the established user's probability matrix:

$$p(\boldsymbol{X}|\boldsymbol{R},\boldsymbol{M}) = \prod_{t=1}^{T}\prod_{i=1}^{L}\left(\sum_{j=1}^{L} r_{ji}m_{ti}\right)^{c_{ti}} \quad (2)$$

A full Bayesian approach then seeks the posterior distribution of the $\boldsymbol{R}$ parameter, i.e., $p(\boldsymbol{R}|\boldsymbol{X},\boldsymbol{M})$. However, we see that this cannot be done tractably, due to the inner summation in Equation 2. This is true even with a maximum likelihood approach (i.e., if we try to maximize Equation 2 without a prior). However, as is common for such situations, by introducing a set of latent variables (one for each observation vector $\boldsymbol{x_n}$) we can derive a tractable joint distribution over both observed and latent variables.

Let latent variable $\boldsymbol{z_n}$ be a binary vector of length L representing the (unobserved) location of the established user at time n (and $\boldsymbol{Z}$ the matrix in which these vectors correspond to the rows). Therefore, under our model of mobility, $\boldsymbol{z_n}$ has a multinomial distribution. This latent variable in turn is used to select which row of $\boldsymbol{R}$ is used as the generative probability distribution of the new user's location at time n. Given that we are dealing with small numbers of observations, the maximum likelihood approach is likely to overfit the data [3]. We therefore choose prior distributions over the multinomial random variables $\boldsymbol{M}$ and $\boldsymbol{R}$. The natural choice of prior is the Dirichlet distribution, which is conjugate to multinomials [3]. Conjugacy means that the posterior and prior have the same form, with respect to the likelihood function, and is extremely useful because it limits complexity. The Dirichlet has a hyperparmeter representing the prior observation count. $\boldsymbol{M}$ and $\boldsymbol{R}$ are assumed to have hyperparameters α and $\boldsymbol{\beta}$, respectively.

The resulting generative hierarchical model of new user location behaviour is presented graphically in Figure 1 and summarized in Algorithm 1. It is similar to probabilistic latent semantic indexing [7], which models text documents as distributions over topics, which in turn generate bags of words in the training set. Documents are analogous to the time slots of the established user, where each time slot has a different distribution over locations (i.e., topics).

Where our approach differs is that we care a lot about the sparsity of observations of new user locations (i.e., words), so assume that the generating conditional probabilities R also follow a Dirichlet distribution. Additionally, we have extra information that we need to integrate, in the form of observations, $\boldsymbol{X'}$, of the topics themselves (i.e., established user locations).

Our model also has strong similarities with latent Dirichlet allocation (LDA) [4], which also models documents, topics, and bags of words hierarchically, but is concerned with the problem of *unseen* documents, and so assumes that the distributions over topics are themselves randomly generated, forming a three-level hierarchy. In contrast, we are comfortable with a predefined set of time slots that repeatedly generate observed locations, as this conforms to strong daily and weekly periodicities found in human location behaviours [5].

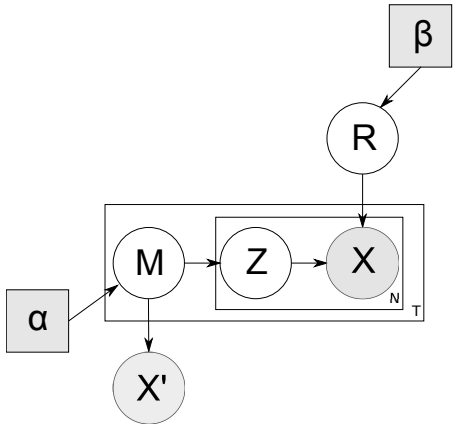


Figure 1. Hierarchical Bayesian model for new user behaviour. Shaded nodes represent observed variables.

Algorithm 1 Generative probabilistic model of new user mobility

1. $\boldsymbol{z_n} \leftarrow Dirichlet(\boldsymbol{m_t} + \boldsymbol{\alpha})$
2. $\boldsymbol{r} \leftarrow \boldsymbol{z_n}\boldsymbol{R}$
3. $\boldsymbol{x_n} \leftarrow Dirichlet(\boldsymbol{r} + \boldsymbol{\beta})$

Finding the MAP of $\boldsymbol{R}$ in the model shown in Figure 1 is done by maximizing $p(\boldsymbol{R}|\boldsymbol{X},\boldsymbol{X'},\boldsymbol{Z},\boldsymbol{M},\alpha,\boldsymbol{\beta})$. This can be achieved with a widely-used algorithm for models involving latent variables, namely, expectation-maximization (EM) [3]. The steps of the EM algorithm, as applied to our model, are given in Algorithm 2.

Algorithm 2 Expectation-maximization algorithm for estimating the maximum *a posteriori* of parameter $\boldsymbol{R}$

$\boldsymbol{R^{old}} \leftarrow$ initialize randomly
$\boldsymbol{M^{old}} \leftarrow$ initialize randomly
repeat
  E-step $\boldsymbol{\gamma} \leftarrow \mathbb{E}[\boldsymbol{z}]$ (Equation 3)
  M-step $\boldsymbol{M^{new}} \leftarrow \arg\max_{\boldsymbol{M}} p(\boldsymbol{M}|\boldsymbol{X'},\boldsymbol{Z},\alpha)$ (Equation 6)
  M-step $\boldsymbol{R^{new}} \leftarrow \arg\max_{\boldsymbol{R}} p(\boldsymbol{R}|\boldsymbol{M},\boldsymbol{X},\boldsymbol{Z},\boldsymbol{\beta})$ (Equation 7)
  $d \leftarrow \mathbf{1}^{\mathbf{T}} abs(\boldsymbol{R^{new}} - \boldsymbol{R^{old}})$
  $\boldsymbol{R^{old}} \leftarrow \boldsymbol{R^{new}}$
  $\boldsymbol{M^{old}} \leftarrow \boldsymbol{M^{new}}$
until $d \leq \epsilon$

We now detail the derivation of the equations used in the two steps of Algorithm 2, specifically, the three equations for the iterative updates of $\boldsymbol{M}$, $\boldsymbol{R}$, and $\boldsymbol{Z}$. Starting with the E-step, the expectation of the latent variable $\boldsymbol{Z}$ can be found by applying Bayes' theorem and the standard updating expression of the Dirichlet distribution over $\boldsymbol{R}$ (one of the assumptions of the model):

$$\begin{aligned}\mathbb{E}(z_{nk}) &= \frac{p(z_{nk}=1)p(\boldsymbol{x_n}|z_{nk}=1)}{\sum_{j=1}^{L_1} p(z_{nj}=1)p(\boldsymbol{x_n}|z_{nj}=1)} \\ &= \frac{m_{(n \mod T)k}\prod_{i=1}^{L_2} r_{ki}^{x_{ni}+\beta_i-1}}{\sum_{j=1}^{L_1} m_{(n \mod T)j}\prod_{i=1}^{L_2} r_{ji}^{x_{ni}+\beta_i-1}} \quad (3)\end{aligned}$$

The M-step consists of maximizing the posterior distributions of $\boldsymbol{M}$ and then $\boldsymbol{R}$ *as though* $\boldsymbol{Z}$ *had been observed.* Starting with $\boldsymbol{M}$, we can factorize the posterior distribution by remembering that all observations are independently and identically distributed. Thus, the observations of the established user, $\boldsymbol{X'}$, essentially contribute towards the prior counts for the Dirichlet distribution $\boldsymbol{M}$ (representing the probability distribution over locations for the established user):

$$\begin{aligned}p(\boldsymbol{M}|\boldsymbol{X'},\boldsymbol{Z},\alpha) &= p(\boldsymbol{M}|\boldsymbol{X'},\alpha)p(\boldsymbol{M}|\boldsymbol{Z},\alpha) \\ &\propto p(\boldsymbol{X'}|\boldsymbol{M},\alpha)p(\boldsymbol{M})p(\boldsymbol{Z}|\boldsymbol{M},\alpha)p(\boldsymbol{M}) \\ &= \prod_{k=1}^{L_1}\prod_{t=1}^{T} m_{tk}^{v_{tk}} \quad (4) \\ \text{where } v_{tk} &= \sum_{w=1}^{\lfloor \frac{N}{T} \rfloor} z_{(wt)k} + c'_{tk} + 2\alpha - 2\end{aligned}$$

Maximizing $p(\boldsymbol{M}|\boldsymbol{X'},\boldsymbol{Z},\alpha)$ with respect to $\boldsymbol{M}$ is achieved by maximizing its logarithm, with a Lagrangian added to constrain the rows of $\boldsymbol{M}$ to sum to 1 (giving Equation 5). G is then differentiated to find the set of multipliers $\boldsymbol{\lambda}$. The logarithm is used simply to make the differential equation easier to solve:

$$\begin{aligned}G &= \sum_{t=1}^{T}\sum_{k=1}^{L_1} v_{tk}\ln m_{tk} + \sum_{t=1}^{T}\lambda_t\left(1-\sum_{k=1}^{L_1} m_{tk}\right) \\ \implies \lambda_t &= \frac{v_{tk}}{m_{tk}} \quad (5)\end{aligned}$$

Solving for m_{tk} gives us the MAP:

$$m_{tk} = \frac{c'_{tk}+\alpha-1+\sum_{w=1}^{\lfloor \frac{N}{T} \rfloor}\left(z_{(tw)k}+\alpha-1\right)}{\sum_{k=1}^{L_1}\left(c'_{tk}+\alpha-1+\sum_{w=1}^{\lfloor \frac{N}{T} \rfloor}\left(z_{(tw)k}+\alpha-1\right)\right)} \quad (6)$$

The derivation of the posterior probability of $\boldsymbol{R}$ follows the same procedure to give:

$$r_{ab} = \frac{\left(\sum_{t=1}^{T} z_{ta}x_{tb}\right)+\beta_b-1}{\left(\sum_{t=1}^{T} z_{ta}C_t\right)+\left(\sum_{i=1}^{L_2}\beta_i\right)-L_2} \quad (7)$$

We now have all three equations necessary for running the EM algorithm on the model. The key output of the procedure is the functional mapping, $\boldsymbol{R}$, between locations of the established user and those of the new user. We next detail how this can be used to enhance prediction.

ENHANCING PREDICTION

Enhancing prediction requires the selection of an established user who has similar location habits to the new user. We do this by finding the established user with the highest posterior probability $p(\boldsymbol{X},\boldsymbol{X'}|\boldsymbol{R},\boldsymbol{Z},\boldsymbol{M},\alpha,\boldsymbol{\beta})$, evaluated on the observed data of the new user. The model of this established user is then mapped, using matrix $\boldsymbol{R}$, to a model approximating the habits of the new user. We next briefly discuss the choice of this mobility model.

There is a wide choice of models for user mobility, including eigendecomposition [5], non-linear time series analysis [10], and Markov models [1, 2]. In general, we leave open the question of which method to use, as this should be specific to the data and intended application. However, to give a concrete example, we detail how our model works with Markov models, which have been applied to mobility modelling in previous settings [1, 2].

A first-order Markov model is represented by a transition matrix of size $L \times L$, indicating the transition probabilities between a given context and the next possible L locations. To represent the transition probabilities between locations of a new user, we use the inferred $\boldsymbol{R}$ matrix to map both the columns and rows to the configuration personal to that user:

$$\boldsymbol{Y_{new}} = \boldsymbol{R}^{\mathbf{T}}\boldsymbol{Y_{est}}\boldsymbol{R} \quad (8)$$

where $\boldsymbol{Y_{new}}$ is the approximated transition matrix for the new user, and $\boldsymbol{Y_{est}}$ is the transition matrix of the established user who best matches that new user. For each observation, the transition probabilities of the next location can be found from the row in $\boldsymbol{Y_{new}}$ corresponding to the current context (i.e., current observed location). The location with the highest probability is always selected as the predicted next location.

REAL-LIFE DATA ANALYSIS

Applying the transformation of Equation 8 to the Markov models of 38 real people allowed us to assess the effectiveness of the approach. We trained a first-order Markov model on the real life location data from the Lausanne Nokia data set [9], which recorded the locations of 38 individuals over the course of a year. To simulate performance of the system on a new user, we truncated the history of a designated 'new' user to the first H hours of observed locations only. Running the EM algorithm (Algorithm 2)[1] and finding the posterior probability $p(\boldsymbol{X},\boldsymbol{X'}|\boldsymbol{R},\boldsymbol{Z},\boldsymbol{M},\alpha,\boldsymbol{\beta})$ allowed us to find the

[1] with hyperparameters α and $\boldsymbol{\beta}$ all set to 1.5

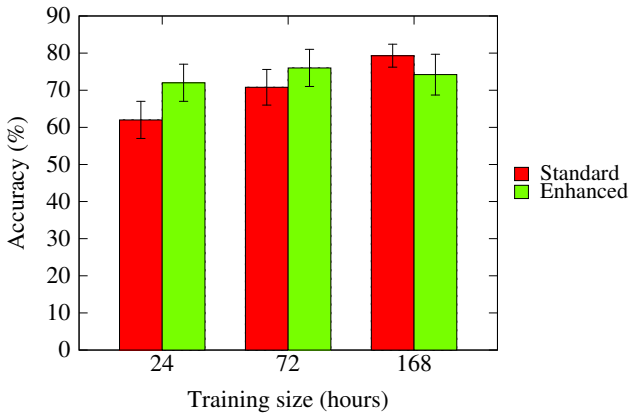


Figure 2. Comparison of performance of the baseline model to the model enhanced with our framework. Error bars indicate the 95% confidence range.

best matching established user from the remaining set of 37 users (whose histories were not truncated).

A first-order transition matrix was then learnt from the best matching established user. We then transformed this matrix, using mapping $\boldsymbol{R}$, to an approximate transition matrix for the new user (with Equation 8). The performance of the approach was evaluated by checking the accuracy of predictions on the rest of the available data for the new user. The same process of simulating a new user was repeated over all individuals in the data set.

As a benchmark, we trained a standard transition matrix on the small amount of truncated data of the new user, which allows us to determine the performance of the Markov model without our framework. Lower order Markov models were previously found to work best on low amounts of training data [11], making this a reasonable benchmark.

Figure 2 shows the results of this procedure for $H = 24$, 72 and 168, i.e., one day, three days, and seven days of observed behaviour as training data, respectively. We see that our framework performs better for very sparse observations (with 24 and 72 hours), implying that our approach is indeed effective at approximating the location habits of new users under these extreme conditions. At 168 hours, no additional improvement is observed in our framework. In contrast, the baseline model improves gradually as the training data size increases. This implies that our framework rapidly reaches its upper limit of performance after only a few days, so should be abandoned after sufficient training sets are made available. Intuitively, the best indicator of future behaviour of an individual is their own past behaviour, once a sufficient history has been gathered.

CONCLUSIONS AND FUTURE WORK

We introduced a new model of location behaviour, capturing the assumption that new users of location prediction services are similar to existing users, subject to an unknown transformation of locations. We applied this model to enhance the accuracy of a first-order Markov model in successfully predicting the next location of real people after just 24 hours of observations.

In future work, we intend to explore ways of making the pairwise choice of new and established users more efficient. Specifically, in a large database of established users, we need to be able to quickly retrieve a relatively small sample of established users most similar to the new user, before applying our training method.

REFERENCES

1. Bapierre et al. A variable order markov model approach for mobility prediction. In *STAMI Workshop at IJCAI*, Barcelona, Spain, 2011.
2. A. Bhattacharya and S. K. Das. LeZi-update: an information-theoretic approach to track mobile users in PCS networks. In *Proc. MobiCom '99*, pages 1–12, 1999.
3. C. M. Bishop. *Pattern recognition and machine learning*, volume 4. Springer New York, 2006.
4. Blei et al. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.
5. N. Eagle and A. S. Pentland. Eigenbehaviors: identifying structure in routine. *Behavioral Ecology and Sociobiology*, 63(7):1057–1066, 2009.
6. Gao et al. Mobile location prediction in spatio-temporal context. In *Nokia Mobile Data Challenge Workshop*, 2012.
7. T. Hofmann. Probabilistic latent semantic indexing. SIGIR '99, pages 50–57, 1999.
8. Lane et al. Enabling large-scale human activity inference on smartphones using community similarity networks. In *UbiComp*, pages 355–364, 2011.
9. Laurila et al. The mobile data challenge: Big data for mobile computing research. In *Proc. Mobile Data Challenge by Nokia Workshop*, 2012.
10. Scellato et al. Nextplace: A spatio-temporal prediction framework for pervasive systems. In K. Lyons, J. Hightower, and E. Huang, editors, *Pervasive Computing*, volume 6696 of *LNCS*, pages 152–169. Springer, 2011.
11. Song et al. Evaluating next-cell predictors with extensive wi-fi mobility data. *IEEE Transactions on Mobile Computing*, 5(12):1633–1649, 2006.
12. X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009:1–19, 2009.

Predicting Future Locations with Hidden Markov Models

Wesley Mathew
wesley.mathew@ist.utl.pt

Ruben Raposo
ruben.raposo@ist.utl.pt

Bruno Martins
bruno.g.martins@ist.utl.pt

INESC-ID
Instituto Superior Técnico
Av. Professor Cavaco Silva
2744-016 Porto Salvo,
Portugal

ABSTRACT
The analysis of human location histories is currently getting an increasing attention, due to the widespread usage of geopositioning technologies such as the GPS, and also of online location-based services that allow users to share this information. Tasks such as the prediction of human movement can be addressed through the usage of these data, in turn offering support for more advanced applications, such as adaptive mobile services with proactive context-based functions. This paper presents an hybrid method for predicting human mobility on the basis of Hidden Markov Models (HMMs). The proposed approach clusters location histories according to their characteristics, and latter trains an HMM for each cluster. The usage of HMMs allows us to account with location characteristics as unobservable parameters, and also to account with the effects of each individual's previous actions. We report on a series of experiments with a real-world location history dataset from the GeoLife project, showing that a prediction accuracy of 13.85% can be achieved when considering regions of roughly 1280 squared meters.

Author Keywords
Hidden Markov Models, Hierarchical Triangular Meshes, Location Prediction, GPS Trajectory Analysis.

ACM Classification Keywords
H.2.8 Database Applications: Data Mining.

General Terms
Experimentation, Human Factors

INTRODUCTION
The widespread usage of localization systems, such as the Global Positioning System (GPS), are making it possible to collect interesting data in many different domains. Modern geopositioning technologies have become ubiquous, and there are nowadays many possibilities for effectively tracking the position of individuals over time. Simultaneously, location-based social networks such as FourSquare[1], or GPS track sharing services such as GoBreadCrumbs[2], are nowadays being increasingly used as a means to store and share human location histories.

One way to use these data is to interpret the shared location histories as the observed portion of complex sequential systems which include hidden contextual variables, such as the activities and goals motivating individual's movements at each time period, in the traces of visits to particular locations. If a distribution over the possible values in such a system can be estimated, then we can use previously observed paths to make inference about hidden states, and afterwards to make informed guesses about other places likely to follow the observed part of these paths.

This paper addresses the development and evaluation of generative models that, by capturing the sequential relations between places visited in a given time period by particular individuals, support the analysis and inference of statistical patterns for predicting future locations to be visited. Specifically, we propose an hybrid method based on Hidden Markov Models, in which human location histories are first clustered according to their characteristics, and in which the clusters are then used to train different Hidden Markov Models, one for each cluster. By leveraging on HMMs for modeling the sequences of visits, we have that the proposed method can account with location characteristics as unobservable parameters, and also with the effects of each individual's previous actions. Through experiments with a real-world location history dataset from the GeoLife project, we measured the prediction accuracy of the proposed method under different configurations. The overall obtained results correspond to a prediction accuracy of 13.85%, when considering regions of roughly 1280 squared meters. In terms of the geospatial distance between the true location of the user, and the geospatial coordinates that are predicted, the best results correspond to an average distance of 143.506 Kilometers, and a median distance of 4.957 Kilometers.

[1]https://foursquare.com/
[2]http://www.gobreadcrumbs.com/

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.

The rest of this paper is organized as follows: Section 2 presents the most important related work. Section 3 details the proposed method, discussing the usage of a technique known as the hierarchical triangular mesh for representing individual locations, detailing the clustering of location histories, and presenting the training and inference of patterns with Hidden Markov Models. Section 4 presents the experimental validation of the proposed method, describing the evaluation methodology and discussing the obtained results. Finally, Section 5 summarizes our conclusions and presents possible directions for future work.

RELATED WORK

Many previous works have addressed the issue of computing with spatial trajectories, and a detailed survey is given in the book by Zheng and Zhou [26]. Moreover, several previous works have specifically focused on the analysis of human location histories, concluding that human trajectories show a high degree of temporal and spatial regularity, following simple and reproducible patterns [3, 8]. In brief we have that previously proposed methods for the analysis of location histories can be classified, according to the manner by which data are modeled, into three general distinct approaches, namely (i) state-space models, (ii) data mining techniques, and (iii) template matching techniques.

State-space models attempt to capture the variation in spatial sequences through sequence models such as generative Hidden Markov Models (HMMs) [17], discriminative Conditional Random Fields (CRFs) [21, 19], or extensions of these two well-known approaches [4, 15]. Generally, these models have been used successfully in dealing with uncertainty (i.e., they generalize well), but they also suffer from high training complexity. In the case of location prediction, generative approaches such as HMMs can naturally be used, since they support the generation of possible future visits and the estimation of an associated probability.

Data mining techniques, on the other hand, explore frequent patterns and association rules, by defining a trajectory as an ordered sequence of time-stamped locations, and using sequence analysis methods such as modified versions of the Apriori algorithm [13, 14]. Most previous data mining methods attempt to maximize confidence with basis on previous occurrences (i.e., they do not generalize as well as state-space models), and they often do not consider any notion of spatial and/or temporal distance.

The third type of approaches, which are based on template matching, compare extracted features to pre-stored patterns or templates, using similarity metrics specific for sequential or time-series data. These similarity metrics include dynamic time warping and other sequence alignment approaches that are essentially variations of the edit distance computed between the sequences (e.g., edit distance with real penalty or edit distance on real sequence). They also include algorithms based on the longest common subsequence, or even other heuristic algorithms similar to those used in more traditional string matching problems [18, 16]. Template matching techniques have also been reported to have issues with high runtime complexity, noise intolerance, or in dealing with spatial activity variation.

Asahara et al. proposed a state-space modeling method for predicting pedestrian movement on the basis of a mixed Markov-chain model (MMM), taking into account a pedestrian's personality as an unobservable parameter, together with the pedestrian's previous status [1]. The authors report an accuracy of 74.4% for the MMM method and, in a comparison over the same dataset, the authors reported that methods based on Markov-chain models, or based on Hidden Markov Models, achieve lower prediction rates of about 45% and 2%, respectively for each of these two cases.

Sébastien et al. extended a previously proposed mobility model called the Mobility Markov Chain (MMC), in order to consider the n previous visited locations [7]. This proposal essentially corresponds to a higher order Markov model. Experiments on different datasets showed that the accuracy of the predictions ranges between 70% to 95%, but they also show that improvements obtained by increasing $n > 2$ do not compensate for the computational overhead.

Morzy, in turn, proposed data mining methods for predicting the future location of moving objects [14]. He extracts association rules from the moving object database and, given a previously unseen trajectory of a moving object, he uses matching functions to select the best association rule that matches the trajectory, afterwards relying on this rule for the prediction. This author reports on an accuracy of 80% for the best configuration of the proposed system.

Other authors still have proposed hybrid sequence analysis approaches, combining multiple types of information. For instance Jeung et al. proposed a hybrid prediction approach which estimates future locations based on pattern information extracted from similar trajectories, together with motion functions based on recent movements [9]. Lu and Tseng used a sequence similarity measure to evaluate the similarity between location histories, afterwards using a clustering algorithm to form a user cluster model of the location histories, based on the similarity measure. Then, using the clusters, the authors predict the movement of individuals, i.e., their next location [12]. Ying et al. proposed an approach for predicting the next location of an individual based on both geographic and semantic features of trajectories, using a cluster-based prediction strategy which evaluates the next locations with basis on the frequent behaviors of similar users in the same cluster, as determined by analyzing common behavior in terms of the semantic trajectories [22].

Previous works have also attempted to analyze human location histories through non-sequential unsupervised approaches based on probabilistic topic models, such as the topic model known as Latent Dirichlet Allocation (LDA) [6]. Probabilistic generative models have been typically used for analyzing document collections, by identifying the latent structure that underlies a set of observations (i.e., words contained within documents). In the case of location histories, the idea behind these models is to represent trajectories as a mixture

of topics, which in turn are modeled as probabilistic distributions over the possible locations.

In this paper, we study the problem of modeling human location histories for predicting the next places to be visited, using a very simple approach based on Hidden Markov Models, and evaluating its limitations on real-world data.

THE PROPOSED METHOD

This paper proposes a simple method for predicting the future locations of mobile individuals, on the basis of their previous visits to other locations, and leveraging on Hidden Markov Models for capturing the patterns embedded in previously collected location histories.

In the proposed method, the previously collected location histories are first clustered according to their characteristics (i.e., according to the temporal period in which they occurred). Afterwards, the clusters are used to train different Hidden Markov Models (HMMs) corresponding to the different types of location histories (i.e., one HMM for each cluster). Given a new sequence of visits, from which we want to discover the particular location that is more likely to be visited next, we start by finding the cluster that is more likely to be associated to the particular sequence of visits being considered in the prediction task, and then we use inference over the corresponding HMM in order to discover the most probable following location.

Each of the places in a given sequence of visits is associated to a timestamp and to the corresponding geospatial coordinates of latitude and longitude. The initial clustering of sequences is based on the temporal period associated to each sequence, and we use the timestamp associated to the last place visited in the sequence, in order to group sequences according to three clusters, namely (i) a cluster for sequences whose visits are made on weekdays by daytime (i.e., from 7AM to 7PM), (ii) a cluster for sequences whose visits are made on weekdays by nighttime (i.e., from 7PM to 7AM in the next day), and (iii) a cluster for sequences containing places visited in weekends. Notice that the proposed clustering approach is only a very simple approximation that uses the timestamp of the last visited location, and not the covered temporal periods themselves. More sophisticated clustering algorithms could indeed be used in the first step of the proposed approach (e.g., clustering through the usage of a mixture of Gaussians), but we leave this for future work.

The Hidden Markov Models for each cluster use only information regarding the geospatial positions (i.e., information derived from the geospatial coordinates of latitude and longitude) for each visited place. Prior to using HMM models, each of the places in a given sequence is first pre-processed in order to convert the real continuous values associated to the geospatial coordinates of latitude and longitude, into discrete codes associated to specific regions. This is done so that the learning of HMMs can be done more efficiently, using categorical distributions over the possible regions in the HMM states, instead of using more complex distributions (e.g., multivariate Gaussian distributions or Von Mises-Fisher spherical distributions) over the real values associated to the geospatial coordinates. To do this discretization, we used the hierarchical triangular mesh[3] approach to divide the Earth's surface into a set of triangular regions, each roughly occupying an equal area of the Earth [20, 5].

In brief, we have that the Hierarchical Triangular Mesh (HTM) offers a multi-level recursive decomposition of a spherical approximation to the Earth's surface. It starts at level zero with an octahedron and, by projecting the edges of the octahedron onto the sphere, it creates 8 spherical triangles, 4 on the Northern and 4 on the Southern hemisphere. Four of these triangles share a vertex at the pole and the sides opposite to the pole form the equator. Each of the 8 spherical triangles can be split into four smaller triangles by introducing new vertices at the midpoints of each side, and adding a great circle arc segment to connect the new vertices with the existing ones. This sub-division process can repeat recursively, until we reach the desired level of resolution. The triangles in this mesh are the regions used in our representation of the Earth, and every triangle, at any resolution, is represented by a single numeric ID. For each location given by a pair of geospatial coordinates on the surface of the Earth, there is an ID representing the triangle, at a particular resolution, that contains the corresponding point. Notice that the proposed representation scheme contains a parameter k that controls the resolution, i.e., the area of the triangular regions. In our experiments, we made tests with the values of 18 and 22 for the parameter k, which correspond to triangles of roughly 1280 squared meters, and of 5.002 squared meters, respectively. With a resolution of k, the number of regions n used to represent the Earth corresponds to $n = 8 \times 4^k$. Figures 1 and 2 illustrate the decomposition of the Earth's sphere into a triangular mesh.

Figure 1. Decomposition of the Earth's surface for triangular meshes with resolutions of zero, one, and two.

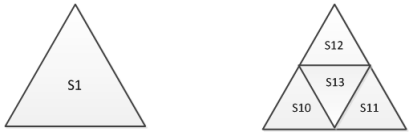


Figure 2. Decomposition of circular triangles.

HMM Training and Inference

Hidden Markov Models (HMMs) are a well-known approach for the analysis of sequential data, in which the sequences are assumed to be generated by a Markov process with unobserved (i.e., hidden) states [17]. Thus, by using HMMs for

[3] www.skyserver.org/htm/Old_default.aspx

modeling location sequences, the states governing the moving agent's decisions are not directly visible, but the visited locations, dependent on the state, are visible. Each state has a probability distribution over the possible locations to be visited, in our case a categorical distribution over the triangles representing each possible area in the Earth's surface. Each state has also a probability distribution over the possible transitions to the state that is going to govern the decision about the next place to be visited in the sequence.

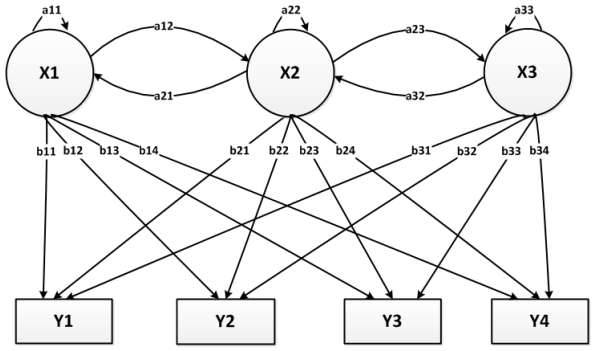


Figure 3. Example Hidden Markov Model.

The diagram in Figure 3 shows the general architecture of an instantiated HMM. Each shape in the diagram represents a random variable that can adopt any of a number of values. The random variable $x(t)$ is the hidden state at time t (in the model from the above diagram, $x(t) \in \{x1, x2, x3\}$). The random variable $y(t)$ is the location visited at time t (with $y(t) \in \{y1, y2, y3, y4\}$). The arrows in the diagram denote conditional dependencies. From the diagram, it is clear that the conditional probability distribution of the hidden variable $x(t)$ at time t, given the values of the hidden variable x at all times, depends only on the value of the hidden variable $x(t-1)$, and thus the values at time $t-2$ and before have no influence. This is called the Markov property. Similarly, the value of the observed location $y(t)$ only depends on the value of the hidden variable $x(t)$, at time t.

Several inference problems can be addressed on top of instantiated HMMs, one of them being the computation of the probability for a given observation sequence (i.e., a sequence of visits to locations). The task is to compute, given the parameters of the instantiated model, the probability of a particular output sequence being observed. This requires computing a summation over all possible state sequences. The probability of observing a particular sequence in the form $Y =< y(1), y(2), \ldots, y(L) >$, of length L, is given by:

$$P(Y) = \sum_{X} P(Y \mid X)P(X) \tag{1}$$

In the formula, the sum runs over all possible hidden-node sequences $X =< x(1), x(2), \ldots, x(L) >$. Applying the principle of dynamic programming, this problem can be handled efficiently, using a procedure known as the the forward algorithm, which we outline further ahead.

As for HMM parameter learning, the task is to find, given an output sequence X or a set of such sequences, the best set of state transition and output probabilities, i.e., the values for $a_{i,j}$ and $b_i(k)$ from Figure 3. The task is usually to derive the maximum likelihood estimate of the parameters of the HMM, given the set of output sequences. No tractable algorithm is known for solving this problem exactly, but a local maximum likelihood can be derived efficiently using the Baum-Welch algorithm, which is an example of a forward-backward algorithm and a special case of the well-known expectation-maximization (EM) algorithm.

For the following presentation of the Baum-Welch algorithm, we can describe an HMM by $\lambda = (A, B, \pi)$, where A is a time-independent stochastic transition matrix between states, B is a stochastic matrix with the probabilities of outputting a particular observation in a given state, and π is the initial state distribution (i.e., a matrix enconding transition probabilities for the particular case of the first position in the sequences). Given a single observation sequence corresponding to $Y =< y(1); y(2); \ldots; y(L) >$, the Baum-Welch algorithm finds $\lambda^* = \max_\lambda P(Y|\lambda)$, i.e. the HMM λ that maximizes the probability of the observation sequence Y.

The Baum-Welch initialization sets $\lambda = (A, B, \pi)$ with random initial conditions. The algorithm then updates the parameters of λ iteratively until convergence, or until a given number of steps has been reached.

Is an expectation step, the forward algorithm is first used to define $\alpha_i(t) = P(y(1) = o_1, \ldots, y(t) = o_t, X(t) = i|\lambda)$, which is the probability of seeing the partial observable sequence $o_1, \ldots, o_t$ and ending up in state i at time t. We can efficiently calculate $\alpha_i(t)$ recursively through dynamic programming, applying the following two equations:

$$\alpha_i(1) = \pi_i b_i(o_1) \tag{2}$$

$$\alpha_i(t+1) = b_i(o_{t+1}) \sum_{j=1}^{N} \alpha_j(t) \times a_{j,i} \tag{3}$$

In the above formula N refers to the size of the set of possible states, a_{ij} refers to transition probabilities, and $b_j(o)$ refers to the emission probabilities. A backwards procedure is also used to compute the probability of the ending partial sequence $o_{t+1}, \ldots, ot$, given that we started at state i at position L. Similarly to the previous case, we can compute a value $\beta_i(t)$ recursively, through:

$$\beta_i(L) = 1 \tag{4}$$

$$\beta_i(t) = \sum_{j=1}^{N} \beta_j(t+1) a_{i,j} b_j(o_{t+1}) \tag{5}$$

Using α and β we can calculate the following variables:

$$\gamma_i(t) \equiv p(Y(t) = i|X, \lambda) = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\beta_j(t)} \tag{6}$$

$$\begin{aligned} \xi_{i,j}(t) &\equiv p(Y(t) = i, Y(t+1) = j|X, \lambda) \\ &= \frac{\alpha_i(t) a_{i,j} \beta_j(t+1) b_j(o_{t+1})}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i(t) a_{i,j} \beta_j(t+1) b_j(o_{t+1})} \end{aligned} \tag{7}$$

Having γ and ξ, in the maximization step, one can define update rules for the HMM as follows:

$$\overline{\pi_i} = \gamma_i(1) \tag{8}$$

$$\overline{a_{i,j}} = \frac{\sum_{t=1}^{L-1} \xi_{i,j}(t)}{\sum_{t=1}^{L-1} \gamma_i(t)} \tag{9}$$

$$\overline{b_i}(k) = \frac{\sum_{t=1}^{L} \delta_{o_t,o_k} \gamma_i(t)}{\sum_{t=1}^{L} \gamma_i(t)} \tag{10}$$

Note that the summation in the nominator of $\overline{b_i}(k)$ is only made over observed symbols equal to o_k, i.e., $\delta(o_t, o_k) = 1$ if $o_t = o_k$, and zero otherwise. Using the updated values of A, B and π, a new iteration of the above procedure is preformed, and we repeat this until convergence.

For more information about the algorithms typically used in hidden Markov modeling problems, please refer to the tutorial by Rabiner [17].

In the particular application addressed in this paper, we used the Baum-Welch approach for estimating the parameters of our Hidden Markov Models, and the task of human movement prediction is reduced to the particular HMM inference problem of computing, given a set of sequences of the form $Y =< y(0), y(1), \ldots, y(L), y(L_{next}) >$, in which the first L positions correspond to places already visited by a particular pedestrian, and in which position L_{next} encodes a possible location to be visited next, the sequence that has the highest probability and, correspondingly, the place L_{next} that is more likely to be visited next. From a given sequence of previous visits, we (i) compute the set of sequences corresponding to all possible next places to be visited, (ii) use the forward algorithm to compute the probability of all such sequences, and (iii) return the next place corresponding to the sequence with the highest probability.

The general approach that is proposed for addressing the location prediction task could also be made to rely on more sophisticated models, such as high-order HMMs of even the recently proposed infinite-HMM model [2]. However, in this paper, we only report on experiments with regular first-order HMMs, and leave other options to future work.

EXPERIMENTAL EVALUATION

We implemented the proposed approach for pedestrian movement prediction through an existing implementation of the algorithms associated with Hidden Markov Models (i.e., the Baum-Welch and the forward-backward procedures), latter validating the proposed ideas through experiments with the GeoLife dataset, i.e. a GPS trajectory dataset collected in the context of the GeoLife project from Microsoft Research Asia, by 178 users in a period of over three years from April 2007 to Oct. 2011 [24, 23, 25]. In the GeoLife dataset, a GPS trajectory is represented by a sequence of time-stamped points, each of which containing latitude and longitude coordinates. The full dataset contains 17,621 trajectories with a total distance of about 1.2 million kilometers and a total duration of 48,000+ hours. The trajectories were recorded by different GPS loggers and GPS-phones, and have a variety of sampling rates, with 91% percent of the trajectories being logged in a dense representation, e.g. every 15 seconds or every 10 meters per sample. The authors claim that the GeoLife dataset recorded a broad range of user's outdoor movements, including not only life routines like going home and going to work, but also some entertainment and sports activities, such as shopping, sightseeing, dining, or hiking. This fact makes the dataset ideal for the purpose of validating pedestrian movement prediction methods.

We experimented with different configurations of the propose method, namely by (i) varying the number of states in the Hidden Markov Models between 10, 15 and 20 states, (ii) varying the resolution of the triangular decomposition of the Earth between resolutions of 18 and 22, and (iii) using the clustering method prior to the training of Hidden Markov Models, versus using a single HMM.

When using the GeoLife dataset, we converted the latitude and longitude coordinates to triangular regions, according to the approach based on the hierarchical triangular mesh, and then we removed from the considered sequences all elements $x(t+1)$ that corresponded to the same triangular region given in position $x(t)$ (i.e., we removed all duplicate consecutive locations from each trajectory in the dataset). All the considered trajectories had a minimum length of 10 locations, and a maximum length of 25 locations (i.e., we only kept the last 25 different locations that were visited). In order to use an equal number of sequences in the training of our different Hidden Markov Models (i.e., in each of the clusters) we randomly selected 3465 different sequences for each of the three clusters. This value corresponds to the number of sequences in the smallest cluster, when considering the entire dataset. Similarly, 3465 different sequences were selected for the case of the experiments with the single HMM. The considered evaluation procedure was based on removing the last location in each of the validation trajectories, afterwards generating predictions for the next place to be visited, and finally checking the prediction against the true location that a given pedestrian visited next.

The training of the HMM models took between 21 and 318 minutes on a standard laptop PC, with results varying according to the number of states.

Table 1 shows the number of different visited locations considered during the training process. In the case of the multiple HMMs, and since each cluster used a different set of sequences, the values that are shown correspond to the average of the values for the different clusters.

Table 2 shows the obtained results for the different configurations, measuring the quality of the obtained results through the Precision@1, Precision@5, and the Mean Reciprocal Rank (MRR) evaluation metrics. Precision@1 measures the percentage of times in which we found the correct next location, whereas Precision@5 measures the percentage of times in which the correct next location was given in the top-five most probable locations. The mean reciprocal rank metric

	Geospatial Resolution	
	18	22
Multiple HMMs	26066	66063
Single HMM	24807	59778

Table 1. Number of different visited sites.

measures the average of the reciprocal ranking positions for the correct next location, in each of the ranked lists with all possible locations that are produced for each trajectory.

Table 2 also illustrates the quality of the obtained results through the average geospatial distance, computed through Vincenty's formulae[4], between the true coordinates of latitude and longitude associated to next place visited by the user, and the centroid coordinates for the triangular region, corresponding to the next location that is predicted. These distance values were measured both for the cases in which the predicted location was correct (i.e., the predicted triangular region contained the next geospatial coordinates of latitude and longitude), and for the cases in which the predicted location was incorrect.

Table 3 shows, instead, the best results obtained with the multiple HMMs corresponding to the different clusters, presenting the results obtained for each of the different clusters.

Finally, in Figure 4 we plot the cumulative distance errors obtained with the best configurations, for the cases in which we used the single or the multiple HMMs, i.e., the distances between the expected locations and the predicted locations, in each sequence. The red line corresponds to the errors for the case of the multiple HMMs, while the black line corresponds to distance errors in the case of the single HMM the lines shown in the chart are not directly comparable, since there are 3 times more sequences in the case of the red line. Still, one can notice that most of the cases correspond to small errors in terms of distance, and indeed we have that the median value for the best configuration corresponds to 4.957 Kilometers.

Overall, the approach were a single HMM was used achieved better accuracies. The single HMM with the configuration using a geospatial resolution of 18, and considering 15 hidden states, achieved the best results in our experiments. The errors in terms of the geospatial distance are generally smaller when we consider smaller values of geospatial resolution (i.e., large prediction areas), and also when we consider higher values for the number of states in the HMMs. Notice that the area of the triangular regions with a resolution of 22 is much smaller than the area of the triangular regions with a resolution of 18. Hence, the accuracies from the experiments using a resolution of 22 are smaller than the accuracies obtained with the resolution of 18, leading also to a worse overall performance in terms of the geospatial distance.

CONCLUSIONS

[4] http://en.wikipedia.org/wiki/Vincenty's_formulae

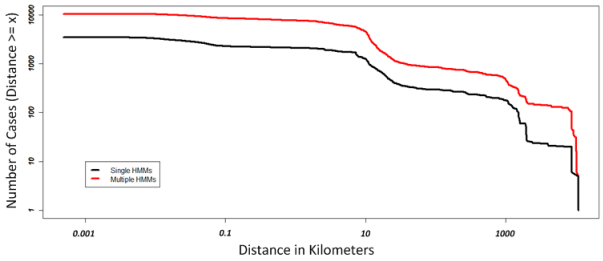


Figure 4. Distribution of the distance errors.

This paper presented an hybrid method for predicting individual's movements on the basis of Hidden Markov Models. The proposed approach clusters human location histories according to their characteristics (i.e., according to the temporal period in which the visits where made), and latter trains a Hidden Markov Model for each cluster, this way accounting with location characteristics as unobservable parameters, and also accounting with the effects of each pedestrian's previous actions. We report on a set of experiments with different configurations of the proposed method, and using a real-world location history dataset from the GeoLife project. We measured a prediction accuracy of about 13.85% with the best performing method. In terms of the geospatial distance between the true location of the user, and the geospatial coordinates that are predicted, the best results correspond to an average distance of 143.506 Kilometers, and a median distance of 4.957 Kilometers.

Despite the interesting results, there are also many ideas for future work. A particular idea that we would like to try relates to improving the training of Hidden Markov Models through posterior regularization, a method that allows one to incorporate indirect supervision (e.g., the fact that consecutive locations that are located close-by should be more probable to occur than others) via constraints on posterior distributions of probabilistic models with latent variables [6].

Many previous works have also addressed the subject of analyzing and classifying sequential data collected from multiple domains, using methods such as sliding window classifiers, Conditional Random Fields (CRFs), Averaged Perceptrons (APs), Structured Support Vector Machines (SVM struct), Max Margin Markov Networks (M3Ns), Search-based Structured Prediction (SEARN) models, or Structured Learning Ensemble models (SLEs). Authors like Thomas G. Dietterich, or Nguyen and Guo, have provided good surveys in the area [15, 4]. For future work, we would also like to experiment with discriminative models, such as those referenced above, in order to address related sequence analysis methods, such as the classification of human location histories according to semantic categories [11, 10].

Acknowledgements

The authors would like to express their gratitude to Fundação para a Ciência e a Tecnologia (FCT), for the financial support offered through the project grant corresponding to

	Parameters		Accuracy			Average Distance in Km		
	Nr. States	Geo. Resolution	P@1	P@5	MRR	Correct	Incorrect	Average
Multiple HMMs	10	18	05.89	14.55	0.105	0.0143	197.855	186.189
	15	18	05.56	**15.69**	0.109	0.0145	197.164	186.201
	20	18	**06.02**	15.44	**0.110**	0.0142	198.197	186.262
	10	22	00.12	00.56	0.005	**0.0008**	193.399	193.157
	15	22	00.17	00.72	0.006	0.0009	**186.098**	185.776
	20	22	00.14	00.72	0.006	0.0008	186.382	186.113
Single HMM	10	18	07.09	24.79	0.149	0.0125	154.797	143.808
	15	18	**13.85**	**26.40**	**0.201**	0.0118	166.580	**143.506**
	20	18	09.26	25.59	0.169	0.0129	158.490	143.808
	10	22	00.08	00.20	0.004	**0.0008**	149.750	149.620
	15	22	00.14	00.89	0.008	0.0010	148.949	148.734
	20	22	00.31	00.75	0.008	0.0010	**148.636**	148.164

Table 2. Results for different configurations of the proposed location prediction method.

	Accuracy			Average Distance in Km		
Cluster	P@1	P@5	MRR	Correct	Incorrect	Average
Weekday 7AM to 7PM (Morning)	08.16	15.55	0.122	0.0142	195.582	179.609
Weekday 7PM to 7AM (Evening)	04.90	16.47	0.110	0.0150	216.451	205.832
Weekend	04.99	14.31	0.097	0.0133	182.455	173.346

Table 3. Results with the multiple HMMs corresponding to different clusters, considering 20 states and a geospatial resolution of 18.

reference PTDC/EIA-EIA/109840/2009 (SInteliGIS).

REFERENCES

1. A. Asahara, K. Maruyama, A. Sato, and K. Seto. Pedestrian-movement prediction based on mixed Markov-chain model. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 25–33. ACM, 2011.

2. M. J. Beal, Z. Ghahramani, and C. E. Rasmussen. The Infinite Hidden Markov Model. In *Proceedings of the 16th Conference on Neural Information Processing Systems*, pages 29–245. MIT Press, 2002.

3. Y. Chon, H. Shin, E. Talipov, and H. Cha. Evaluating mobility models for temporal prediction with high-granularity mobility data. In *Proceedings of the 10th IEEE International Conference on Pervasive Computing and Communications*, pages 206–212. IEEE Computer Society, 2012.

4. T. G. Dietterich. Machine Learning for Sequential Data: A Review. In *Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, pages 15–30. Springer-Verlag, 2002.

5. G. Dutton. Encoding and Handling Geospatial Data with Hierarchical Triangular Meshes. In *Proceedings of the 7th Symposium on Spatial Data Handling*, pages 505–518. Spatial Effects, 1996.

6. K. Farrahi and D. Gatica-Perez. Discovering routines from large-scale human locations using probabilistic topic models. In *ACM Transaction Intelligent System Technology*, pages 1–27. ACM, 2011.

7. S. Gambs, M. Killijian, D. P. Cortez, and N. Miguel. Next place prediction using mobility Markov chains. In *Proceedings of the 1st Workshop on Measurement, Privacy, and Mobility*, pages 1–6. ACM, 2012.

8. M. C. Gonzalez, C. A. Hidalgo, and A. Barabasi. Understanding individual human mobility patterns. *Nature*, (7196):779–782, 2008.

9. H. Jeung, Q. Liu, H. T. Shen, and X. Zhou. A Hybrid Prediction Model for Moving Objects. In *Proceedings of the 24th IEEE International Conference on Data Engineering*, pages 70–79. IEEE Computer Society, 2008.

10. J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. Do, O. Dousse, J. Eberle, and M. Miettinen. The Mobile Data Challenge:Big Data for Mobile Computing Research. In *Proceedings of the Workshop on the Nokia Mobile Data Challenge, in Conjunction with the 10th International Conference on Pervasive Computing*, pages 1–8. ACM, 2012.

11. L. Liao, D. Fox, and H. Kautz. Extracting Places and Activities from GPS Traces Using Hierarchical Conditional Random Fields. *International Journal of Robotics Research.*, (1):119–134, 2007.

12. E. H. Lu and V. S. Tseng. Mining Cluster-Based Mobile Sequential Patterns in Location-Based Service Environments. In *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware*, pages 273–278. IEEE Computer Society, 2009.

13. A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. WhereNext: a location predictor on trajectory pattern

mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 637–646. ACM, 2009.

14. M. Morzy. Mining Frequent Trajectories of Moving Objects for Location Prediction. In *Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 667–680. Springer-Verlag, 2007.
15. N. Nguyen and Y. Guo. Comparisons of sequence labeling algorithms and extensions. In *Proceedings of the 24th International Conference on Machine Learning*, pages 681–688. ACM, 2007.
16. O. Ossama and H. M. O. Mokhtar. Similarity Search in Moving Object Trajectories. In *Proceedings of the 15th International Conference on Management of Data*, pages 1–6. Computer Society of India, 2009.
17. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Readings in Speech Recognition*, pages 267–296. Morgan Kaufmann Publishers, 1990.
18. D. E. Riedel, S. Venkatesh, and W. Liu. Recognising online spatial activities using a bioinformatics inspired sequence alignment approach. *Pattern Recognition*, (11):3481–3492, 2008.
19. C. Sutton and A. McCallum. An Introduction to Conditional Random Fields. *Foundations and Trends in Machine Learning*, To be published.
20. A. S. Szalay, J. Gray, G. Fekete, P. Z. Kunszt, P. Kukol, and A. Thakar. Indexing the Sphere with the Hierarchical Triangular Mesh. *Computing Research Repository*, pages 58–65, 2007.
21. D. L. Vail, M. M. Veloso, and J. D. Lafferty. Conditional random fields for activity recognition. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1–8. ACM, 2007.
22. J. J. Ying, W. Lee, T. Weng, and V. S. Tseng. Semantic trajectory mining for location prediction. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 34–43. ACM, 2011.
23. Y. Zheng, Q. Li, Y. Chen, X. Xie, and W. Ma. Understanding mobility based on GPS data. In *Proceedings of the 10th International Conference on Ubiquitous Computing*, pages 312–321. ACM, 2008.
24. Y. Zheng, X. Xie, and W. Ma. GeoLife: A Collaborative Social Networking Service among User, Location and Trajectory. *IEEE Data Data Engineering Bulletin*, (2):32–39, 2010.
25. Y. Zheng, L. Zhang, X. Xie, and W. Ma. Mining interesting locations and travel sequences from GPS trajectories. In *Proceedings of the 18th International Conference on World Wide Web*, pages 791–800. ACM, 2009.
26. Y. Zheng and X. Zhou. *Computing with Spatial Trajectories*. Springer, 2011.

Beyond "Local", "Categories" and "Friends": Clustering foursquare Users with Latent "Topics"

Kenneth Joseph
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15212
kjoseph@cs.cmu.edu

Chun How Tan
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15212
chunhowt@cmu.edu

Kathleen M. Carley
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15212
kathleen.carley@cs.cmu.edu

ABSTRACT

In this work, we use foursquare check-ins to cluster users via topic modeling, a technique commonly used to classify text documents according to latent "themes". Here, however, the latent variables which group users can be thought of not as themes but rather as factors which drive check in behaviors, allowing for a qualitative understanding of influences on user check ins. Our model is agnostic of geo-spatial location, time, users' friends on social networking sites and the venue categories- we treat the existence of and intricate interactions between these factors as being latent, allowing them to emerge entirely from the data. We instantiate our model on data from New York and the San Francisco Bay Area and find evidence that the model is able to identify groups of people which are of different types (e.g. tourists), communities (e.g. users tightly clustered in space) and interests (e.g. people who enjoy athletics).

Author Keywords

location-based service, foursquare, topic modeling

ACM Classification Keywords

H.5.m Information Interfaces and Presentation: (e.g. HCI); J.4 Social and Behavioral Sciences: Sociology

INTRODUCTION

There has long been an interest in understanding how, when, where and why people move from place to place [14]. In recent years, such studies have begun to focus more on the large amount of geo-spatially tagged data being produced from mobile devices, as such data allows one to approach questions of societal-level interest in an entirely data-driven manner [26, 7, 27]. Data drawn from mobile devices on the whereabouts of their users has led to an influx of interesting findings in explaining patterns of human mobility [6, 17, 5], predicting friendship on social networking sites based on location data [23, 20], and better understanding different aspects of cities, both at the city level as a whole [24] and at the neighborhood level [7, 8].

A significant amount of previous work on human mobility has come to the conclusion that people tend to stay within relatively small geographic areas for the majority of their time [18, 17, 16, 6]. These areas, particularly in cities, are often representative of different neighborhoods - areas in cities with dynamic, fuzzy boundaries [7, 8] whose residents exhibit homophilic tendencies, both in their demographics and social interactions [9]. Because of the importance of neighborhoods in a diverse set of social processes (e.g. [21]), a natural way of conceptualizing groups of people in cities is to cluster them based on the neighborhoods in which they reside or are most active.

Yet while heterogeneity in a population can to a large extent be explained by closeness of social and geodesic distances [4] (a closeness inherent in neighborhoods), there are other ways of defining "groups" of people in cities. For example, one could consider tourists, who almost by definition are not bound to specific parts of a city, as being a group of interest. Similarly, sports enthusiasts, bound not to neighborhoods but more to the specific places (e.g. stadiums) they frequent, may be of interest as well. These two examples represent interesting and useful groupings of people moving within cities which the concept of a neighborhood cannot fully capture.

In this work, we consider an alternate approach to defining groups of people in a city by characterizing people simply by the places they go. Such an approach has previously been found effective for uncovering social and interest relationships between users as well as for location and friend recommendations, though the approach taken also considered temporal information and utilized a different type of location data [10, 27]. We work with a large dataset of foursquare check ins, obtained from the authors of [7], which details where and when people were at specific locations around different U.S. cities. While the dataset gives a diverse set of information, we describe each user simply by the places they go and how often they go there, thus choosing to ignore geospatial and social information which exists in the data. In addition, we ignore information on the category of different places, as explained in later sections.

Using such a simple feature set for users is somewhat counterintuitive, however, we do so with a specific purpose. First, the relationship between social ties, geodesic distances between people and their temporal coevolution are intertwined in an intricate manner which has only recently begun to be understood in a strong quantitative manner [4, 9, 10, 19]. By only implicitly considering these variables in our data rep-

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.
978-1-4503-1224-0/12/09

resentation, we allow for them to emerge as latent factors whose association we need not explicitly predefine. This allows a further understanding of how these variables may affect the check in behaviors of users. In addition, by not specifying any presumed factors to be responsible for similar check in locations between users, we avoid restrictions of the types of groups our model might find. For example, explicitly using geo-spatial features may restrict our ability to understand groups of users with similar interests which are spread throughout a city, such as the tourists described above.

Given that our feature set is simple and we desire an understanding of what causes users to check in at various locations, the model we use must allow us to posit that latent factors affect where users go within cities, quantify how each user is affected by them, and give some intuition as to what these latent factors might be. We use a clustering model based on the idea of topic modeling, a method of clustering which captures these very concepts. Specifically, our model assumes that every user can be represented by multiple hidden factors, and that each check in by that user is motivated by one or more of these hidden variables. These hidden factors may represent, for example, interests or needs of a user, but the range of their distinction is broad, as topic models force the researcher to determine the qualitative meaning behind the hidden factors it discovers. Users can then be grouped by how strongly they are affected by these hidden factors, and the hidden factors themselves can be defined by a certain set of locations (or venues) which are frequented by the users clustered together by it.

In order to understand what might be gained from a topic model approach to location-based social network data, we instantiate Latent Dirichlet Analysis (LDA) on foursquare data from check ins in New York City and the Bay Area. We find that our model produces latent collections of people which represent both geo-spatially close groups and people who appear to have similar interests, thus suggesting social factors are at play. Our model therefore captures drivers of user check in behavior found to be so important in grouping people in cities into neighborhoods, lending another piece of support to previous work in this area [7, 9, 4]. However, in addition to latent factors indicating groupings due to social and geo-spatial closeness, we also find clusters of different "types" of people, such as tourists, who tend to visit specific venues which do not seem, qualitatively, to have any clear geo-spatial or social relevance. Thus, by using a model agnostic of place category, geo-spatial information and friendship information, we create a model which is rich enough to incorporate all of them and extend beyond to include these user "types". This allows for a deeper understanding of user check in behavior on location based social networking sites.

After describing our findings, we discuss some potential applications of a topic-model approach, such as venue recommendation. We also consider ways in which our model is different from preexisting methods of recommendation using similar data [27, 10], with particular concern to note the limitations of the current approach.

RELATED WORKS

The Data

The data that we use, obtained from the authors of [7], is a set of approximately 18 million data points from across the United States of users of foursquare. Foursquare is a socially-driven location sharing application [11], where users can "check in" to different locations and have these check ins be shared with friends both on foursquare and on various other social networking sites. In addition to allowing users to share check ins with others, foursquare uses various gamification techniques to encourage contributions, including rewarding users with badges and points for various actions.

Indeed, these gamification techniques have been found to be a strong determinant in use of foursquare. Lindqvist et al. [11] found the most likely reason for a check in was for the gamification aspects of the site, followed by social aspects (e.g. to interact with friends), to visit and discover new places, and to keep track of personal history and accomplishments. Users were also asked for reasons why they would not check in at a location- these centered on privacy concerns and issues of self-representation. Self-representation concerns the fact that users have a desire to be represented as being of a certain type, leading to the possibility that a user's check ins could mis-represent his or her interests. For example, users surveyed tended to not want to check in to places which they perceived to be uninteresting (e.g. work) or embarrassing (e.g. fast food). Such findings are pertinent to our understanding of the clusters resulting from our topic model in that they must be analyzed from the perspective of the typical user- one that is at least partially interested in gamification and self-representation.

Each of our 18 million data points represents one check in which was published to Twitter by a foursquare user. In the data set used, a check in provides a unique user Twitter id, the timestamp of the user check in, an optional user description (e.g. "the coolest place ever!"), and also the venue id of the check in location. Using this venue's id, the original data collectors also obtain the venue's name, geo-location, and "category" information. These categories are drawn from a set of hierarchical categories given by foursquare itself - there are over three hundred categories, the full list of which can be found by querying the foursquare API [1]. We utilize these categories extensively in defining the hidden factors which our topic model generates. From the data set collected by the authors of [7], we consider check ins located in the metropolitan areas of New York City and the San Francisco Bay Area.

Human Mobility

Noulas et al, in [17], study distances users travel between successive check ins, noting that nearly 80% of the total check ins for a user occur within 10 kilometers of the previous check in. Though larger than the typical neighborhood, this lends support to the idea that people tend to stay within small sections of cities and hence can be grouped in this manner. Similarly, user displacement, or distance between two successive check ins, follows a power law which

[1] https://developer.foursquare.com/index

can be modeled by a Lévy Flight [5]. Work in [16] shows that across various American cities, the density of the city is negatively correlated with user displacement.

While such work suggests that users tend to stay within reasonably small areas, particularly in dense areas, they provide little evidence of the specific places people are traveling to at various distances. Cho et al. [6] explain mobility using the concept of two places, home and work, but do not go in depth into travel to places which might represent user interests. In contrast, Cranshaw et al., in [7], present an approach which utilizes geographic proximity in combination with user check in history, thus incorporating a better understanding of the specific places people travel to. This information is utilized to understand how the existence of "neighborhoods" can be approximated by foursquare check ins. Our work is different from the work of Cranshaw et al. in that we focus on clustering users, as opposed to venues, into groups. It is important to notice, however, that the method in [7] can be applied without modification to show a variety of neighborhoods around a city which similar users might frequent. Such a model suggests that incorporating venue categories as a feature describing user movement restricts the formation of neighborhood clusters- each neighborhood has its own set of venues within a variety of categories.

The question of user clustering, as opposed to venue clustering, has been previously approached [12, 22, 10], most noteably in [10], where a hierarchical, temporally aware user clustering mechanism (HGSM) is proposed. This method is extended to show its abilities as a recommender system in [27], where it is shown to perform well on tasks involving recommending both places and social connections for users. We discuss how this model compares to the one presented here in Section .

One might be tempted to assume that human movement can be much better understood if it is conditioned on the movement of friends. Indeed, much work has been done to show that one can reliably predict the location of a user based on the location of their friends (see [20, 23] for recent examples). Furthermore, recent work has shown that neighborhoods implied by census boundaries can be inferred from social graphs [9]. However, evidence from [6] suggests that while location prediction is possible, predicting where a user will go based on where their friends on location-based services go is not as straightforward. Cho et al. [6] state that people who are friends on Brightkite and Gowalla have a check in in common less than ten percent of the time. Furthermore, the authors find that travel over short distances is not heavily impacted by the social network structure - friendship links on location-sharing social networks only can explain about 30% of all check ins. These findings suggest to us that explicitly adding in features of friendship on social networking sites may restrict clusters to community-based groups, perhaps overpowering other latent variables such as user interests which exist in the data.

MODEL DESCRIPTION

To cluster foursquare users into meaningful groups which are representative of different factors driving check in behavior, we apply the idea of topic modeling. Specifically, we apply a topic model known as Latent Dirichlet Allocation (LDA), first introduced in [3]. LDA is a latent space model commonly used to better understand text corpora by representing a large collection of documents in a much more compact set of hidden topics. In a typical LDA model (as discussed in [3]) a text document is represented as a set of words, where each word is assumed to belong to one or more hidden topics. Thus, each document can be described by considering how heavily the words within it relate to the various hidden topics, and each hidden topic can be described by the words which are most heavily associated with it. For example, a document about the opening of a new Italian restaurant might contain the words "restaurant" and "dinner", associated with Topic 1, and the words "pizza" and "spaghetti", associated with Topic 2. LDA would give us information on how heavily the document was related to the two topics, and we could understand what these two topics were about by considering the words which are associated with them (e.g. Topic 2 is likely about "food" or "Italian Food"). By considering documents which are highly associated with the same topic, we can begin to understand "clusters" (or groups) of documents in the corpora, where each cluster represents a set of documents which are related to a given topic.

Ref. [8] has successfully applied LDA to location check in data. Specifically, [8] applied a topic-model approach on socially-tagged data from a location-sharing social network in order to understand boundaries that might exist on neighborhoods and characteristics of these neighborhoods. In their topic model, the "documents" were regions generated by splitting geo-spatial coordinates into grids, their "words" were venue category tags, and their topics were hypothesized to be archetypal neighborhoods. Note, however, that LDA was still performed on text.

In contrast, we use an instantiation of the data which does not revolve around the concept of themes in text. Rather, in order to model user check in behaviors using LDA, we use the analogy of a document to represent a user, and thus each check in for a user can be thought of as a word in a document. As each venue has a unique identifier, we can model each as a unique word. This means, for instance, that the Starbucks on 5th Street will be different than the Starbucks on 10th Street. Similar to text documents, where documents can have the same word multiple times, we define a multinomial distribution for the check ins for each user by using the counts of check ins for each venue as features.

Using our representation of user check in behavior, we obtain a set of hidden topics which can each be described by a set of venues (words), and which can be used to categories users (documents) according to these topics. Because these topics have to do with check ins at different venues, we can associate them not with textual themes but rather with factors which drive users to check in at various locations (e.g. interest). More specifically, for each user, we obtain a set

of weights corresponding to each hidden topic, allowing us to understand multiple facets of the behavior of each user. Thus, a benefit of using LDA is that each user can be represented as a distribution of a variety of drives in behavior. This coincides well with our intuition that check in behavior is driven by multiple underlying factors, each of which may be used to correlate the behavior of a user with others and thus may help to better understand user check in behavior at a general level.

In the case studies below, we set the number of hidden topics to be twenty. A shortcoming of LDA, addressed in later topic models (as shown in [3]) is that an arbitrary number of hidden topics need be chosen. We complete sensitivity tests, as suggested in [3], and find that our model is most effective and most interpretable when we use twenty clusters. In addition, we remove those users with less than 5 unique venue check ins and those venues with less than 10 check ins, repeating the pruning iteratively until all such venues and users are removed. This approach of pruning data points is common in document modeling, as in imperfect documents there tend to be spelling mistakes which occur rarely and are obviously not of interest. Similarly, in our case, those users who have few check ins might be newcomers to foursquare who quickly stop using the application and thus might not be well-represented by their check ins. However, this pruning criteria is selected arbitrarily and future work in the direction of data selection is important.

RESULTS

In this section, we present two exploratory case studies of the results of running LDA on check in data from New York City and the San Francisco Bay Area. The New York data set initially had a total of 448,156 check ins, 36,388 users and 44,312 venues. After pruning, we were left with 288,029 check ins, 10,459 users and 7,432 venues. Note that we still keep more than half of the check ins, although the number of venues and users decreases significantly. While future work may make use of these data points, we find that including them in the model makes clusters more difficult to interpret, as is often the case when incorporating analogous words and documents when running LDA on text corpora. Similarly, for San Francisco Bay Area data, we initially have a total of 181,572 check ins, 18,650 users and 20,844 venues, and were left with 102,851 check ins, 4,269 users and 3,439 venues after applying the same pruning criteria.

In our analysis, we examine the "top" venues in each cluster, as given by weights from the LDA. By observing these venues, we are able to better understand the latent factor which is representative of the users in each cluster. In the following sections, we provide a qualitative analysis of three different "kinds" of latent factors that our model uncovers as being hidden drivers in where users check in. We develop this intuition by considering the geo-spatial distribution and categorical information (garnered from foursquare category information) of the venues which represent each cluster. Thus, the cluster types that we generate are in some cases quite similar to the typical category of the venues within that cluster, though we will show that this is not always the case.

Category	Venue Name
"Sport Enthusiast" Cluster	
Baseball Stadium	Yankee Stadium
Football Stadium	MetLife Stadium
Baseball Stadium	Stadium Citi Field
Hockey,Basketball Arena	Madison Square Garden
"Art Enthusiast" Cluster	
Performing Art Venue	NBC Studio 1A
Art Museum	Brooklyn Museum
Art Museum	Metropolitan Museum of Art
Art Museum	Museum of Modern Art (MOMA)

Table 1. Top venues of two clusters found in the New York data. These two clusters can be thought of as clustering users based on similar interests

Interest Factors

We define interest factors as those where the top venues within that cluster are all associated with a specific action which can be performed, such as eating ice cream or watching a sporting event. Two examples of such clusters in the New York data are shown in Table 1 - similar clusters, not shown, are observed in the San Francisco data. In many respects, one would question any model of user behavior which does not in some way account for the interests of the user, and as such the fact that our model discovers interests as a hidden driver of user action is not of particular surprise. However, such clusters are of interest in suggesting that certain arguments recently put forth for describing human mobility are too simplistic. In particular, claims that simple concepts of geo-spatial phenomenon, as suggested in [16], are sufficient to explain human mobility cities should be treated with some skepticism - our model suggests contextual factors, including points of interest at different locations in cities are clearly of high importance [27]. This can be observed by the relatively wide geo-spatial spread of the two interest clusters from the New York data, as can be seen in the clusters with these names in Figure 1.

In addition to casting doubt on the simplicity of human mobility modeling, the existence of clusters of users driven by what we presume to be similar interests suggests that social factors have a strong underlying effect on locations that users check in to. This claim is based on the well-supported notion that social acquaintances tend to have similar interests [13]. Such a finding indicates that a topic model approach to clustering users may be an effective route to generating friendship recommendations [27] with the added benefit of being able to give reasons for the recommendation which are less intrusive. For example, instead of suggesting to two users in New York that they may want to become friends because they have both visited Yankee Stadium and MetLife stadium within the past few months, a topic model approach (with the assistance of a practitioner) could instead suggest that these two users become friends because they are both "Sport Enthusiasts". Though other approaches might provide similar functionality, it would likely be on a more case-by-case basis.

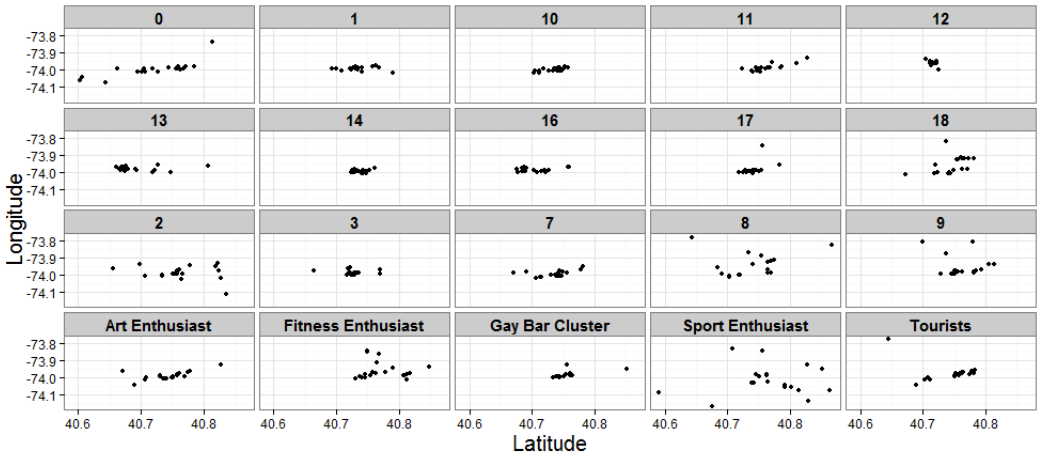


Figure 1. The geo-spatial distribution of the twenty clusters our model discovers in New York - each point represents one of the top twenty venues in each cluster. Those clusters assessed which could be qualitatively associated with a name are labeled with these names, as discussed in the paper. Those with numbers are not qualitatively assessed in this work-they are shown to give a better understanding of spatial distribution of clusters

Category	Venue Name
Bridge	George Washington Bridge
Gay Bar	Therapy NYC
Gay Bar	Boxers NYC Sportsbar
Gay Bar	Ritz Bar and Lounge
Gay Bar	Splash Bar
American Restaurant	Elmo Restaurant and Lounge
Gym	Equinox
Gay Bar	Posh
Train Station	New York Penn Station
Gay Bar	Pieces Bar
Coffee Shop	Starbucks
Gay Bar	XES Lounge
Gay Bar	Barrage

Table 2. A cluster consisting mainly of gay bars, found in the New York data

Category	Venue Name
Gay Bar	Toad Hall
Park	Mission Dolores Park
Gay Bar	Badlands
Gay Bar	QBar
Gay Bar	Club Trigger
Gay Bar	The Lookout
Burger Joint	Harvey's
Gay Bar	440 Castro
Gay Bar	Blackbird Bar
Gay Bar	The Mix
Supermarket	Safeway
Movie Theater	AMC Loews Metreon 16
Gay Bar	Moby Dick
Train Station	Castro MUNI Metro Station

Table 3. The San Francisco "gay bar" cluster. These venues are all found in The Castro, an area with a large gay population. Notice that venues of other types are included in the cluster

Community Factors

Given the previous work which suggests that geo-spatial (and thus social [4]) factors influence user mobility, it is also not surprising to see several clusters which are tightly clustered in space. However, as our model ignores the geo-spatial coordinates of venues, it is interesting to note that such clusters are purely the result of a group of users which are all driven by some hidden factor driving them to check in within a small geographical area. It is important to note, however, that this factor could be representative of issues of self-representation, or to some genuine factor influencing users to stay within that area. We thus define these clusters as "communities", in that users either feel strongly that they are associated with this specific area, or are indeed frequenters of the area.

In the results from both the New York and San Francisco Bay Area data, a "community" cluster exists where the representative venues are nearly all of the category "Gay Bar". A list of places which describe these clusters in the two data sets are shown in Table 2 and Table 3, and the closeness of these places in space can be seen for the New York data in Figure 1. Thee San Francisco gay bar cluster is similarly close in space- in fact nearly all venues in the cluster are located in "The Castro", a neighborhood well-known for its gay population. This can be seen in Figure 2, where each marker represents one of the top twenty venues associated with the given cluster. What is particularly interesting is that the observed hidden factor associated with these clusters correlates well with a segment of the population which is heavily discriminated against, fitting traditional notions which suggest that people who are discriminated against tend to coalesce into tight communities [2]. Indeed, while many other types of venues carrying explicit demographic information about their users, such as churches, exist in foursquare's categories, this was the only one to repeatedly appear as a topic across both cities and various model configurations. The ability of foursquare data to reveal such segregations even when geo-spatial properties of venues are ignored is a rather interesting finding which we hope to explore in later work.

User Type Factors

The final kind of cluster we uncover in our results groups users by hidden factors we refer to as a "type". We define type clusters as those which group users into a recognizable form which is clearly distinguishable quantitatively but rep-

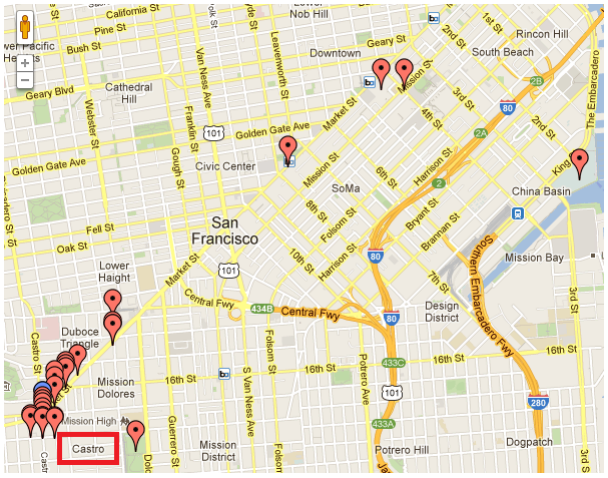


Figure 2. The geo-spatial distribution of the community cluster found in the San Francisco data. Each marker denotes a location, and the name of the traditionally gay area in San Francisco, the Castro, is boxed in red for the reader.

Category	Venue Name
Electronics Store	Apple Store
Train Station	New York Penn Station
Train Station	Grand Central Terminal
Park	Central Park
Airport Terminal	Terminal 5
Art Museum	Museum of Modern Art (MOMA)
Park	Bryant Park
Art Museum	Metropolitan Museum of Art
Department Store	Macy
Bridge	Brooklyn Bridge
Plaza	Rockefeller Center
Science Museum	American Museum of Nat. History
Historic Site	National September 11 Memorial
Toy or Game Store	FAO Schwarz
Monument	Statue of Liberty
Hotel	Hilton New York
Art Museum	Guggenheim Museum

Table 4. A cluster consisting mainly of tourist attractions in New York

resents users across a varied geo-spatial setting and across a variety of possible interests. As such, these type clusters could be considered a kind of "catch-all", however in noting that we only suggest that a few clusters qualify, we do not consider it as such. A type cluster for New York can be seen in Table 4, where the users grouped into this cluster appear to be of the type "tourist". We make this claim based on the fact that most of the places representing this cluster are sites which tourists would visit, and includes several travel venues, in particular the airport. Similarly in the San Francisco Bay Area data, we identify a type cluster corresponding to Stanford students, as shown in Table 5. Here, the top venues are either establishments within Stanford University or common places that college students might visit in San Francisco (e.g. movie theaters and bars), along with a few public transport stations.

Type clusters are interesting in that they show the ability of a latent model to capture relationships between users which cannot be easily expressed in a parameterized model. As such, our model can be seen here to transcend simple categorical, geo-spatial and social factors which influence users

Category	Venue Name
Subway Station	Civic Center BART Station
Subway Station	Balboa Park BART Station
University	The Quad
University	Jordan Hall
University	Gates CS Building
University	Hoover Tower
Nightclub	The Ambassador
Movie Theater	AMC Bay Street 16 and IMAX
Bridge	San Francisco-Oakland Bay Bridge
Light Rail	BART - Transbay Tube
Stanford	Stanford Golf Course
Stanford	Stanford University
Plaza	The Claw
Sculpture Garden	Rodin Sculpture Garden
Movie Theater	Regal Emery Bay 10
Movie Theater	AMC Loews Metreon 16

Table 5. A cluster from the San Francisco Bay Area which seems to consist of Stanford students

to check in a different locations, and thus gives evidence of topic models as being a useful approach for location based data. In particular, as many users utilize foursquare to present themselves as being a specific type of person [11], topic models which expose different characteristic types of people, in addition to user interests, may be much more apt to make recommendations based not on superficial factors but on more internalized ones such as geo-spatial homophily. While such clusters are interesting in defining non-obvious, latent factors affecting where users check in, a drawback to type clusters is that they do not have as distinctive features, such as common venue categories or tight geo-spatial locations, as the other kinds of clusters we observe. Thus, these clusters clearly require increased knowledge of the city at hand to justify their existence, and as such should be approached with caution until more quantified means of analyzing their existence are examined.

FUTURE WORK AND LIMITATIONS

There are several application areas which could be pursued with the output from our model. One obvious use of the clusters discovered would be to design a recommendation system, as is done in [27] with the user similarity metric developed first in [10]. As noted, our model can recommend related places which are not necessarily of same category, leading to robust and interesting recommendations based on representative latent factors which have driven previous check ins of users. For example, Figure 3 shows a cluster of users that seem to frequent fitness centers. We observe that these "fitness enthusiasts" also frequent NikeTown, an athletic clothing store which would make for a very reasonable recommendation. One advantages to our approach in location recommendation over previous user similarity approaches in [27, 15] is that we cluster specifically on Points of Interest (POI), as opposed to geographic coordinates, thus allowing us to recommend areas without the need to have an additional filtering step. In addition, users are already grouped according to factors which drive their interests or needs, thus allowing us to avoid costly user similarity calculations in situations which are time critical, such as on-the-fly recommendations.

Category	Venue Name
Arts & Entertainment::Stadium::Tennis	USTA Billie Jean King National Tennis Center
Professional & Other Places::Office	DXagency
Arts & Entertainment::Stadium::Tennis	Arthur Ashe Stadium
Shop & Service::Gym or Fitness Center	Equinox
Shop & Service::Gym or Fitness Center::Gym	New York Sports Club
Shop & Service::Gym or Fitness Center	Equinox
Shop & Service::Gym or Fitness Center::Gym	New York Sports Club
Shop & Service::Clothing Store	NikeTown
Shop & Service::Gym or Fitness Center::Gym	New York Sports Club
Professional & Other Places::Office	Definition 6: NYC
Professional & Other Places::Office	Razorfish NYC
Shop & Service::Department Store	Walmart Supercenter
Food::Coffee Shop	Starbucks
Great Outdoors::Park	Central Park
Shop & Service::Gym or Fitness Center::Gym	Train Daly

Figure 3. A cluster found by LDA using New York data that consists mainly of gyms (in blue), yet also including a sporting apparel store (red).

Venue Names	
Century Cinema 16	Computer History Museum
AT&T Park	Stanford Stadium
Stanford University	In-N-Out Burger
Cafe Borrone	Mission Bay Conference Center
Hacker Dojo	Apple Inc.
Googleplex	Microsoft SVC
LinkedIn	Googleplex - 43
Google San Francisco	Googleplex - Charlie's Cafe
Facebook	Plug And Play Tech Center
The Company Store	Stanford Shopping Center
San Francisco Caltrain	Mountain View Caltrain

Table 6. A cluster in the San Francisco Bay Area data made up of check ins from what one would expect to be a tech event(s).

Another possible use of our model, if applied in an online form [1], would be to better understand the dynamics of check ins due to groups of users being in cities for various events. An example of a group formed by a possible event is observed in the data from San Francisco in Table 6. The cluster discovered is represented by a collection of places in the San Francisco Bay Area which relate strongly to famous technology sites across the city, transportation and conference centers. While we could not confirm it, we believe that this cluster is representative of a technology event which brought technology fans into the city, who in turn toured important sites in the field around the Bay Area.

The existence of a possibly fleeting group of users points to one clear limitation of our model - by ignoring temporal information in the data, we assume that groupings of users (and thus the factors affecting their check in behaviors) are heavily static, which is likely not the case. Topic models which consider temporal information, such as periodicity [25], may be able to garner interesting clusters over time. We also ignore temporal information with respect to sequences of user actions, a significant shortcoming of our model as compared to the user similarity model proposed in [10, 27]. At least five other limitations exist. First, the results of the model are not always interpretable - it is difficult, particularly if one is not familiar with the city in which the check ins occurred, to understand certain clusters. Foursquare categories help, but can only explain so much about the intricacies of user behavior.

A second problem of our model is that, in addition to being difficult to interpret, the resulting clusters are also not predictable - because the model is probabilistic, results can change slightly with each run. Furthermore, results are not predictable across cities, for example, we did not find a tourist cluster in San Francisco Bay Area. Third, the model is sensitive to the amount of data it is given. In noting that San Francisco Bay Area had much less data compared to New York, we also find that the clusters are not as well defined. This was also true for cities for which we had even less data, such as Pittsburgh and Chicago. One possible solution would be to incorporate other, similar data types, such as Yelp data. Fourth, in the text modeling domain, "stop words" are often removed, words such as "a" and "the" which are highly frequent. It might behoove a model of places to do the same. However, while it might make sense to remove uninteresting places such as airports and bus stations, it is unclear if popular places representative of interests, like stadiums, should really be removed. While we considered this avenue, we did not obtain rigorous findings in this direction. Finally, a hierarchical approach, as implemented in [10], would allow us to extend beyond the current categorizations.

CONCLUSIONS

The model we present is simplistic in the features of the data it incorporates. We group users into clusters based only on the places they go and thus do not incorporate explicit representations of geo-spatial, categorical or social aspects of our data. Because of this simplistic methodology, significant limitations, suggested above, exist in the practical usage of the model we present. Indeed, we do not discuss how such a model would compare to those which incorporate more rich features, and discuss how in some ways, previous user similarity metrics are more desirable than the one presented here. Future work, particularly those keen on understanding the applicability of our approach to recommendation technologies, should indeed incorporate a user study which allows for the comparison of a topic model approach, using various feature sets, to different mechanisms for recommendation.

However, the simplicity of our model allows us to generate a more data-driven understanding than has been previously explored of the latent factors which may be driving user check in behavior in data from location based social networks. Our findings confirm that geo-spatial and social homophily are powerful factors in grouping user into different types, interests and communities, thus supporting a large amount of work which suggests the same (e.g. [7, 4, 9, 13]). However, in addition to supporting previous work, we extend their efforts in two ways. First, we find that by typifying different users with a categorical, qualitative type such as "tourist", one can understand check in behavior beyond patterns in social, geo-spatial and venue categories. Second, for those groups which are in fact bonded by social and geo-spatial factors, our model allows for interpretation of the groupings beyond these variables to specific traits, such as homosexuality, which define a part of the community itself.

ACKNOWLEDGEMENTS

We would like to thank Justin Cranshaw for the data and for his assistance in developing the concepts presented. We would also like to thank Robert Kraut, Jason Hong and our

anonymous reviewers for their invaluable suggestions.

REFERENCES

1. L. Al Sumait, D. Barbara, and C. Domeniconi. On-line LDA: adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, Proc. ICDM'08, pages 3–12. IEEE Computer Society.
2. P. Blau. *Inequality and heterogeneity: A primitive theory of social structure*. New York: Free Press, 1977.
3. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.
4. C. Butts. *Space and Structure: Methods and Models for Large-Scale Interpersonal Networks*. Springer, 2012, expected.
5. Z. Cheng, J. Caverlee, K. Lee, and D. Z. Sui. Exploring millions of footprints in location sharing services. In *Proc. ICWSM '11*, pages 81–88. AAAI, 2011.
6. E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proc. SIGKDD '11*, pages 1082–1090. ACM, 2011.
7. J. Cranshaw, R. Schwartz, J. I. Hong, and N. Sadeh. The livehoods project: Utilizing social media to understand the dynamics of a city. In *Proc. ICWSM '12*. AAAI, 2012.
8. J. Cranshaw and T. Yano. Seeing a home away from the home: Distilling proto-neighborhoods from incidental data with latent topic modeling. In *CSSWC Workshop at NIPS 2010*, NIPS '10. AAAI, 2010.
9. J. R. Hipp, R. W. Faris, and A. Boessen. Measuring neighborhood: Constructing network neighborhoods. *Social Networks*, 34(1):128 – 140, 2012. Capturing Context: Integrating Spatial and Social Network Analyses.
10. Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W. Ma. Mining user similarity based on location history. In *Proc. SIGSPATIAL '08*, GIS '08, pages 34:1–34:10. ACM, 2008.
11. J. Lindqvist, J. Cranshaw, J. Wiese, J. Hong, and J. Zimmerman. I'm the mayor of my house: examining why people use foursquare - a social-driven location sharing application. In *Proc. CHI '11*, pages 2409–2418. ACM, 2011.
12. M. Loecher, D. Rosenberg, and T. Jebara. Citysensetm: Multiscale space time clustering of gps points and trajectories. In *Joint Statistical Modeling*, JSM '09, 2009.
13. M. McPherson, L. Lovin, and J. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, (1):415–444, 2001.
14. S. Milgram. A Psychological Map of New York City. *American Scientist*, 60:194–200, Mar. 1972.
15. J. Moore. Building a recommendation engine, foursquare style, Mar. 2011.
16. A. Noulas, S. Scellato, R. Lambiotte, M. Pontil, and C. Mascolo. A tale of many cities: universal patterns in human urban mobility. *ArXiv e-prints*, Aug. 2011.
17. A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. An empirical study of geographic user activity patterns in foursquare. In *Proc. ICWSM '11*, pages 570–573. AAAI, 2011.
18. R. Park and E. Burgess. *Introduction to the Science of Sociology*. University of Chicago, 1921.
19. M. T. Rivera, S. B. Soderstrom, and B. Uzzi. Dynamics of dyads in social networks: Assortative, relational, and proximity mechanisms. *Annual Review of Sociology*, 36(1):91–115, 2010.
20. A. Sadilek, H. Kautz, and J. P. Bigham. Finding your friends and following them to where you are. In *Proc. WSDM '12*, pages 723–732. ACM, 2012.
21. R. J. Sampson, S. W. Raudenbush, and F. Earls. Neighborhoods and violent crime: A multilevel study of collective efficacy. *Science*, 277(5328):918–924, 1997.
22. C. Sato, S. Takeuchi, and N. Okude. Experience-based curiosity model: Curiosity extracting model regarding individual experiences of urban spaces. In A. Marcus, editor, *Design, User Experience, and Usability. Theory, Methods, Tools and Practice*, volume 6770 of *Lecture Notes in Computer Science*, pages 635–644. Springer Berlin / Heidelberg, 2011.
23. S. Scellato, A. Noulas, and C. Mascolo. Exploiting place features in link prediction on location-based social networks. In *Proc. SIGKDD '11*, pages 1046–1054. ACM, 2011.
24. S. Wakamiya, R. Lee, and K. Sumiya. Crowd-based urban characterization: extracting crowd behavioral patterns in urban areas from twitter. In *Proc. SIGSPATIAL International Workshop on Location-Based Social Networks*, pages 77–84. ACM, 2011.
25. Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Lpta: A probabilistic model for latent periodic topic analysis. In *Proc. ICDM '11*, pages 904 –913, dec. 2011.
26. Y. Zheng. Location-Based social networks: Users. In Y. Zheng and X. Zhou, editors, *Computing with Spatial Trajectories*, pages 243–276. Springer New York, 2011.
27. Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W. Ma. Recommending friends and locations based on individual location history. *ACM Trans. Web*, 5(1):5:1–5:44, Feb. 2011.

Exploring Trajectory-Driven Local Geographic Topics in Foursquare

Xuelian Long
University of Pittsburgh
xul10@pitt.edu

Lei Jin
University of Pittsburgh
lej17@pitt.edu

James Joshi
University of Pittsburgh
jjoshi@pitt.edu

ABSTRACT

The location based social networking services (LBSNSs) are becoming very popular today. In LBSNSs, such as Foursquare, users can explore their places of interests around their current locations, check in at these places to share their locations with their friends, etc. These check-ins contain rich information and imply human mobility patterns; thus, they can greatly facilitate mining and analysis of local geographic topics driven by users' trajectories. The local geographic topics indicate the potential and intrinsic relations among the locations in accordance with users' trajectories. These relations are useful for users in both location and friend recommendations. In this paper, we focus on exploring the local geographic topics through check-ins in Pittsburgh area in Foursquare. We use the Latent Dirichlet Allocation (LDA) model to discover the local geographic topics from the check-ins. We also compare the local geographic topics on weekdays with those at weekends. Our results show that LDA works well in finding the related places of interests.

Author Keywords

Location-based Social Networking Services, *Foursquare*, Geographic Topics, Trajectory

ACM Classification Keywords

J.4 Computer Applications: social and behavior science; H.3.5 Online Information Services: web-based services

General Terms

Measurement, Human Factors

INTRODUCTION

Nowadays, Location-based Social Networking Services (LBSNSs) [1] are becoming more and more popular. Because of the rapid developments of the fast 4th generation mobile networks, the powerful interfaces supporting map services, and the smartphones in which GPS modules are embedded, it is very easy for the mobile users to identify their locations and share them in LBSNSs. In a LBSNS, users can explore the places of interests, check in at their current locations, leave tips or comments and add new friends. Therefore, LBSNSs, such as *Foursquare*, *Facebook Places*, etc., have recently attracted a lot of users by using different mechanisms; further, they find ways to motivate them to share their locations in their systems. For example, *Foursquare* had nearly 20 million users in March 2012 [2]. *Facebook Places* reported 200 million monthly active users creating 2 billion actions tagged with locations by April, 2012 [3].

Sometimes, it may be daunting for users to find the right places of interests in LBSNSs, when they go to a new city and/or are not familiar with the city. Thus, LBSNSs, such as *Foursquare*, have launched categories associated with the venues to facilitate the search. For example, if a user is interested in finding a nearby Mexican restaurant, he can choose to explore the venues with "Mexican" as the category in *Foursquare*. But such help is limited because these categories are static and predefined. Moreover, users are interested in other types of categorizations of the venues, i.e. categorizing the venues in accordance with the crowd level, which we call the *geographic topics* in this paper. For example, users may be more interested in knowing which restaurants people usually go to after shopping at a mall; which cafe is more popular around their current locations. To support such user needs, it is important to provide geographic topics in LBSNSs.

Topic models are very common and useful in text classification. In this paper we propose a topic-model based approach for venue classification based on users' trajectories. The key premise is that the venues that appear together in many users' trajectories will probably be taken as geographic topics. Therefore, the venues in the same trajectory-driven geographic topic are potentially and intrinsically related by the human mobility. These geographic topics can be used to (1) understand users' preferences of venues, (2) recommend venues to users based on previous preference, (3) recommend friends and (4) design business strategies.

In this paper, we use the check-in data related to Pittsburgh area collected from *Foursquare* to explore the trajectory-driven local geographic topics. We employ the Latent Dirichlet Allocation (LDA) approach to discover the local geographic topics. Our main contributions are as follows:

- Our proposed topic model based approach investigates the potential and intrinsic relations among the different venues in Pittsburgh area. Our approach can dynamically categorize the venues in *Foursquare* according to the users' trajectories that indicate the crowds' preferences of the venues. The results of the local geographic topics can be used for recommending both locations and friends to users in LBSNSs.

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

	Check-ins	Venues	Users
All	813,221	16,461	32,113
Weekdays	574,372	16,222	26,224
Weekends	238,849	13,780	22,868

Table 1. Summary of the Data Set

- We consider the temporal differences in discovering geographic topics. Users' check-in patterns on weekdays are quite different from those at weekends. Thus, we apply our model for users' weekday as well as weekend trajectories separately in this paper and identify the differences of the check-in patterns between them.
- Our data set is collected from *Foursquare* directly. Other data sets used in many current research related to *Foursquare* data analysis are typically gathered from *Twitter*, e.g. [17, 12]; hence, these data sets are less complete because: 1) users may not push every check-in in *Foursquare* to their *Twitter* accounts; and, 2) there are about less than 25% *Foursquare* users who connect their *Foursquare* accounts with their *Twitter* accounts [5]. Thus, our analysis is more comprehensive and accurate because our data set is more complete than the *Twitter* based data sets.

In the rest of the paper, we summarize our data set and then describe the geographic topic modeling. We also present our experiments and discuss some interesting findings based on the local geographic topics generated by our experiments. We review the related work and conclude the paper with a discussion of our future work.

DATA SET

We crawled users' check-in data in Pittsburgh area in *Foursquare* from Feb 24 to May 23, 2012. We define Pittsburgh area as a square with sides of around 40 miles and centered at Pittsburgh Downtown. We use *Foursquare* APIs to discover as many venues as possible in this region and we collect the check-ins at these venues. We have removed the venues with only one check-in. Such single check-ins are not useful for topic categorization and may introduce noise in the LDA model that we use. Our data set is summarized in Table 1.

Foursquare defines a hierarchical list of categories applied to venues. There are 9 top categories in the hierarchical structure and they are: *Arts & Entertainment*, *College & University*, *Food*, *Professional & Other Places*, *Nightlife Spot*, *Great Outdoors*, *Shop & Service*, *Travel & Transport* and *Residence*. In our data set, there are 45,125 check-ins at the venues that do not belong to any category. Figure 1 shows the venue distribution of the other 768,096 check-ins in the top 9 categories.

From Figure 1, we can see that the check-ins in *Food* and *Shop & Service* categories are always the most for weekdays, weekends as well as overall. However, the check-ins in *College & University* and *Professional & Others* categories on weekdays are far more than those at weekends. This is not difficult to understand as users usually do not work at

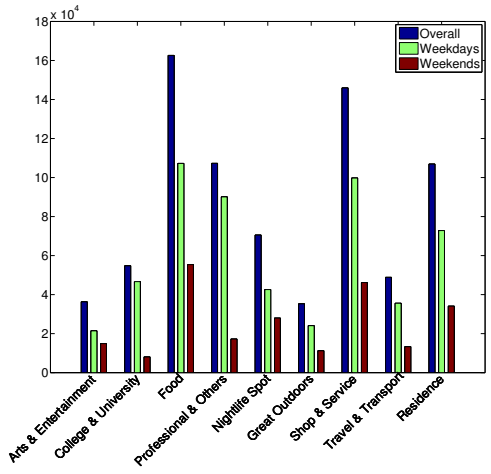


Figure 1. Check-ins distribution in top 9 categories

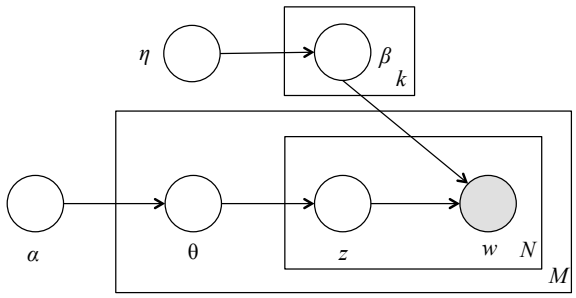


Figure 2. Graphic model representation of LDA [6]. The boxes are "plates" representing replicates. The plate M represents documents, while the plate N represents the repeated choice of topics k and words within a document.

weekends. These categories can help to understand the local geographic topics as described in the experiment section.

GEOGRAPHIC TOPIC MODELING

In this section, we first introduce the LDA model and then show how to use it to discover the geographic topics.

LDA Model

Blei *et al.* present Latent Dirichlet Allocation (LDA), which is a probabilistic model in [6]. LDA is usually used to cluster documents based on the topics contained in a corpus of documents. Ferrari and Mamei present two reasons to choose the LDA in analyzing users' mobility patterns and routine behaviors in [13]. One is that there is no need to define topics a priori and the other is that the topic results represent meaningful probabilistic distributions over words and documents [13].

The graphic model of LDA is shown in Figure 2. α and η are parameters of the Dirichlet prior on the per-document topic distributions and per-topic word distributions, respectively. θ_i is the topic distribution for document i. β_k is the word distribution for topic k. w_{ij} is the jth word in document i and z_{ij} is the topic for w_{ij}.

Trajectory-driven Geographic Topic Modeling

We adopt the LDA to identify the geographic topics in our data set. The basic unit is word in the LDA and, in this paper, the venue in a single check-in record represents a word. A user's trajectory consisting of all the venues of his check-ins represents a document, which is a set of words. An advantage of using LDA is that we do not need to predefine the topics and only need to set the number of the topics.

We focus on the local geographic topics in this paper; so M users' trajectories within a specific city make up the corpus in the model. Every user's trajectory can be described as a mixture of the geographic topics that are essentially distributions over the geo-locations. Therefore, the LDA can be applied to the mobility data in this way.

EXPERIMENT

In this section, we first introduce our data set and describe how we process the data set to make it suitable for the LDA model. After that, we present several experiments to evaluate the proposed approach. In particular, we first investigate the overall local geographic topics in our data set. Since the users' check-ins patterns differ for weekdays and weekends [8, 12], we also explore the geographic topics on weekdays and at weekends. We use the MATLAB Topic Modeling Toolbox to run our experiments[16].

Data Preparation

In each check-in record, we have the user who created the check-in and the venue where the check-in was created. Moreover, we can get the creation timestamp of the check-in. We conduct three experiments. In the first experiment, we do not consider the creation timestamps of the check-ins. In the second and third experiments, we divide our data set into two subsets according to the creation timestamps of the check-ins, i.e. the weekday subset consists of the check-ins created on weekdays and the weekend subset consists of the check-ins created at weekends.

In these three experiments, the basic units in the trajectory-driven geographic topic model are the venues of the check-ins. A document consists of the venues from a user's check-ins, e.g. ($venue_{check-in1}$,...,$venue_{check-inN}$). We set the number of topics $T = 30$, $\alpha = 50/T$ and $\eta = 0.1$ in all these experiments.

Local Geographic Topics

In this subsection, we present the overall local topics discovered by the first experiment where the timestamps of the check-ins are not counted. We get a total of 30 local geographic topics and use OTopics to denote the topics in the first experiment. Table 2 lists the 8 OTopics derived from the first experiment. In each OTopic, we also list top 10 venues. In addition, we discuss the spatial features in these topics.

Overall Local Geographic Topics

In the section where we introduce our data set, we plot the distribution of check-ins in top 9 categories. From Figure 1 we can see that the check-ins in *Food* and *Shop & Service* are the largest in number. In the generated OTopics, we find that many of them are related to these two categories. For instance, OTopic 7, 9, 19 and 28 are topics related to food and shopping and most of the top 10 venues in these four topics are food and shopping venues. It indicates that users would go to restaurants after shopping. The different topics give an overview of clusters of the shops and restaurants people usually check in at together.

We also have OTopics that are related to education. OTopic 3 is associated with the University of Pittsburgh and OTopic 5 is associated with the Carnegie Mellon University. Both the universities are located at Oakland neighborhood in Pittsburgh. The food venues in these two topics indicate places that students and faculty members most likely go frequently.

Some OTopics are related to sports. OTopic 4 is a case in point. CONSOL Energy Center is the home stadium of the hockey team–the Pittsburgh Penguins and PNC Park is the home stadium of the baseball team–the Pittsburgh Pirates. Moreover, the probabilities of these two venues assigned to this topic are ten times higher than the other venues in this topic. There is also a very famous football team in Pittsburgh–the Pittsburgh Steelers, whose home stadium is Heinz Field. The reason that Heinz Field is not in this topic is because our data collection period does not overlap with the football season.

There are also OTopics that are related to businesses such as OTopic 11, since there are several professional buildings in this topic. An interesting observation is that there are two tunnels in this topic, which are located in the major route of Pittsburgh (I–376). The existence of these two venues indicates that many users commute between work and home through these tunnels.

In summary, the advantages of the user-driven local geographic topics include the following: (1) the topics generated depend only on the users' trajectories but not on the physical locations or the pre-defined categories; and (2) the topics provide a novel view of the location classifications based on the human mobility. That is, the venues in a geographic topic imply that people usually go there together with high probability.

The Spatial Features of the Topics

We are also interested in the spatial features of the generated local geographic topics, i.e. whether the venues in each topic are close to each other or not? Whether there are any spatial relations among the venues in the same topic?

Figure 3 shows the spatial distribution of the top 10 venues in the eight OTopics illustrated in Table 2. We can see that the venues in some OTopics are very close, e.g. the top 10 venues in OTopic 5. The venues in some other OTopics are very sparse, e.g. the top 10 venues in OTopic 4. Considering the categories of the topics, venues in the topics related to education (OTopic 3 and OTopic 5) are usually geographically closer. Venues in topics related to businesses or entertainment may not be close to each other. It is not difficult to understand because people usually commute between home

Topic 7	Topic 9	Topic 19	Topic 28
South Hills Village Mall	AMC Loews Waterfront 22	Galleria at Pittsburgh Mills	Walmart Supercenter
Giant Eagle Market District	P.F. Chang's	Cinemark IMAX Theater	Quaker Steak & Lube
Starbucks	Target	Walmart Supercenter	Buffalo Wild Wings
Red Robin Gourmet Burgers	Giant Eagle	Do Drop Inn	Primanti Brothers
Houlihan's Mt. Lebanon	Planet Fitness	Giant Eagle	Costco
T.G.I Friday's	Red Robin Gourmet Burgers	Walmart Supercenter	Giant Eagle Market District
Trader Joe's	Eat'n Park	Target	Giant Eagle
Giant Eagle	T.G.I. Friday's	Applebee's	Starbucks
Walmart	Costco	Giant Eagle	North Park Lounge in HD
GetGo	The Waterfront	UPMC St. Margaret Hospital	Target
Topic 3	**Topic 5**	**Topic 4**	**Topic 11**
Cathedral of Learning	University Center	CONSOL Energy Center	BNY Mellon Center
Hillman Library	Gates-Hillman Complex	PNC Park	Comcast
Hemingway's Cafe	USX Tower	PNC Park	American Eagle HQ
Benedum Hall	Wean Hall	Olive Garden	Fort Pitt Tunnel
Posvar Hall	Hunt Library	Starbucks	Fitness 247
Petersen Events Center	Morewood Gardens	Joe's Crab Shack	PNC Firstside Center
William Pitt Union	Starbucks	Verizon Wireless	"Bellevue, PA"
Peter's Pub	Panther Hollow Inn	St. Clair Hospital	Squirrel Hill Tunnel
Schenley Plaza	Doherty Hall	CCAC Milton Hall	Crawford Square Apartments & Townhomes
Chipotle Mexican Grill	Hamburg Hall	U.S. Steel Clairton Works	Element Church 205 North

Table 2. Examples of Trajectory-driven Overall Local Geographic Topics

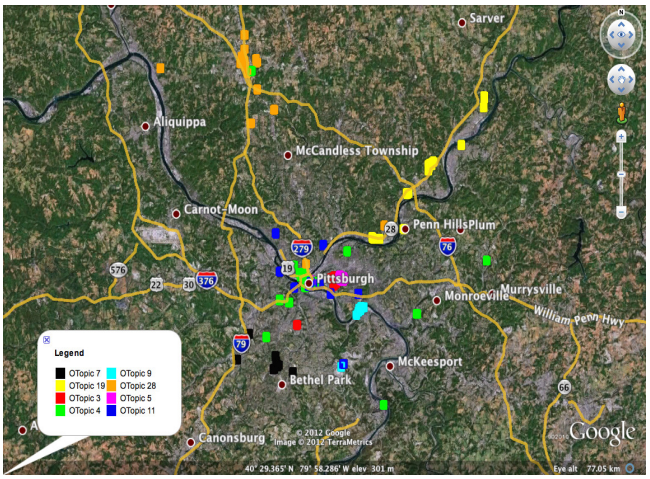


Figure 3. The spatial features of the topics

and work (e.g. OTopic 11) and the stadiums are usually not close to the fans' home (e.g. OTopic 4). However, the case is different for the OTopics related to shopping. Some people would like to go shopping and dining at venues that are not far away. For example, OTopic 9 and OTopic 7; the top 10 venues in these two topics are very close. Some people would like to go shopping and dining at venues that are along the freeway, e.g. OTopic 19; the top 10 venues in this topic are along the freeway.

Figure 3 essentially indicates that venues in the user trajectory-driven local geographic topics are not always geographically close to each other. Our work is a strong complement to the recent work based on the topic model or clustering of the venues in LBSNSs presented in [8, 12]. Moreover, our work can be used for business planning. For example, if there is a store or restaurant that is not close to the majority of top venues in a topic, the owner of the store or restaurant may need to consider if a branch office should be opened in the area with the majority of venues in the topic.

Local Geographic Topics on Weekdays vs. Those at Weekends

We analyze the geographic topics on weekdays and at weekends in the second and third experiments in this subsection as a complementary for Figure 1 that show the differences between the weekday check-ins and the weekend check-ins.

Local Geographic Topics on Weekdays

We expect that there would be local geographic topics related to university, professional work and business on weekdays and we use WDTopics to denote the topics on weekdays. The results confirm our expectation. We list some topics in Table 3. WDTopic 25 relates to the University of Pittsburgh and WDTopic 26 relates to the Carnegie Mellon University. The top 10 venues in these two topics are almost the same with those in OTopic 3 and OTopic 5.

We have several WDTopics related to professional work and business. For instance, WDTopic 5 is related to medical work, as 8 out of 10 top venues in this topic are hospitals. One possible reason could be that patients usually go to

WDTopic 25	WDTopic 26	WDTopic 5	WDTopic 17
Cathedral of Learning	University Center	UPMC Presbyterian Hospital	Rivers Casino
Hillman Library	Gates-Hillman Complex	UPMC Shadyside	IKEA
Benedum Hall	Wean Hall	Allegheny General Hospital	American Eagle HQ
Hemingway's Cafe	Hunt Library	UPMC Mercy	Comcast
Posvar Hall	BNY Mellon Client Service Center	UPMC Montefiore	Emerald Gardens Apartments
William Pitt Union	Morewood Gardens	UPMC St. Margaret Hospital	Oakmont Tavern
David Lawrence Hall	Porter Hall	Verizon Wireless	Fort Pitt Tunnel
Schenley Plaza	Doherty Hall	Western Pennsylvania Hospital	Rivertowne North Shore
Petersen Events Center	Starbucks	Forbes Regional Hospital	HSLS Falk Library: 200 Scaife Hall
Mad Mex	Tepper School of Business	"Plum, PA"	Squirrel Hill Tunnel
WDTopic 1	**WDTopic 18**	**WDTopic 7**	**WDTopic 9**
CONSOL Energy Center	PNC Park	Pittsburgh International Airport (PIT)	AMC Loews Waterfront 22
Urban Active Fitness	Stage AE	Pittsburgh International Arprt	P.F. Chang's
PNC YMCA	Heinz Hall	The Westin Convention Center	Walmart Supercenter
PITT School Of Information Sciences	Forbes Tower	David L. Lawrence Convention Center	Target
Buffalo Wild Wings	Hamburg Hall	Wyndham Grand Pittsburgh Downtown	Century III Mall
DoubleTree - Green Tree	Heinz Field	T.G.I. Friday's	Giant Eagle
Oakland	PNC Park	Freedom High School	Red Robin Gourmet Burgers
Bear Run Village	Olive Garden	Heritage Hills Townhomes Apartments	Eat'n Park
Renaissance Pittsburgh Hotel	Starbucks	EFI, Inc. (Pittsburgh Office)	Amberson Towers
Ariba Inc.	Pittsburgh International Airport (PIT)	Tonic Bar And Grill	Jefferson Regional Medical Center

Table 3. Examples of Trajectory-driven Geographic Topics on Weekdays

these hospitals for different purposes. Since the University of Pittsburgh Medical Center (UPMC) is the largest medical center in western Pennsylvania area, patients may have been referred to different specialists who work in different hospitals belonging to the UPMC. Another reason could be that the doctors, nurses and other medical staff work at different locations at different times. We do not identify a topic related to the medical work in the overall topics.

We also have a topic (WDTopic 17) with two tunnels–Fort Pitt Tunnel and Squirrel Hill Tunnel that are also included in OTopic 11. However, the top 10 venues in WDTopic 17 are a little different from those in OTopic 11. Rivers Casino and IKEA are in WDTopic 17 but not in OTopic 11. The reason may be that there are many people who work at Rivers Casino and IKEA and usually commute on the main route I–376 on weekdays.

The WDTopics related to sports are a little different with the OTopic related to sports. We can see that CONSOL Energy Center is in WDTopic 1 and PNC Park is in WDTopic 18. In both these topics there are hotels in the top 10 venues that may imply that hockey fans or baseball fans fly to Pittsburgh for a game on weekdays. We also see University of Pittsburgh School of Information Sciences and Oakland in WDTopic 1. It may be because university students are more likely to watch the early hokey game on weekdays at a very cheap price [4].

WDTopic 7 shows that airport on weekdays are related to businesses, as there are many conferences or meetings at The Westin Convention Center, David L. Lawrence Convention Center and Wyndham Grand Pittsburgh Downtown.

WDTopic 9 gives an example of the topics in weekdays related to shopping and food. The venues in WDTopic 9 are almost the same as the venues in OTopic 9.

Local Geographic Topics at Weekends

At weekends, we can expect more topics related to entertainment and shopping. Besides, there would be very few topics related to education, business and professional work. Our results indeed confirm these. In the 30 topics, there is no topic related to any university, which indicates that there are not as

WETopic 4	WETopic 16	WETopic 10	WETopic 27
Pittsburgh International Airport (PIT)	Heinz Hall	Target	The Cheesecake Factory
Pittsburgh International Arprt	Carnegie Science Center	Eat'n Park	Bar Louie
Hard Rock Cafe Pittsburgh	Pittsburgh Zoo & PPG Aquarium	Giant Eagle	Claddagh Irish Pub
Robert Morris University Island Sports Center	Petersen Events Center	The Waterfront	Pittsburgh Marriott City Center
T.G.I. Friday's	Red Robin Gourmet Burgers	Dave & Buster's	Emerald Gardens Apartments
3:36	Carnegie Museum of Natural History	T.G.I. Friday's	The Altar Bar
Baggage Claim	Joe's Crab Shack	AMC Loews Waterfront 22	Grand Concourse
Hard Rock Cafe	Station Square	Barnes & Noble	Megabus Pittsburgh
"Freedom, Pa"	Bar Room	Costco	Starbucks
House	Eat'n Park	Giant Eagle	Joe's Crab Shack

Table 4. Examples of Trajectory-driven Geographic Topics at Weekends

many users at weekends checking in at universities as those on weekdays in *Foursquare*. There is no topic about professional work or business, either. All the topics are related to food, shopping and entertainment. We use WETopics to denote the local geographic topics at weekends and Table 4 lists some examples.

WETopic 4 shows the geographic topic related to airport. We do not see any convention centers in the top 10 venues in this topic. It may imply that users usually do not take flight for business reasons at weekends. WETopic 16 is mainly related to arts and entertainment. There are typically many concerts, operas and stage shows in Heinz Hall. Carnegie Science Center and Carnegie Museum of Natural History are very famous museums in Pittsburgh. Petersen Center usually host university sport events. WETopic 10 is related to shopping and food. The top 10 venues in this topic are very similar with those in OTopic 9. WETopic 25 is mainly related to food, which is different from the OTopics as the food venues usually appear together with the venues in shopping or the venues in entertainment.

Comparisons Between WDTopics and WETopics

Comparing all the local geographic topics on weekdays and those at weekends, we observe the following:

- There is no topic related to education, business or professional work out of the 30 WETopics. This is not difficult to understand as people usually do not go to school or work at weekends; rather they most likely engage in social, family events and recreations at weekends. Thus, the topics at weekends are mostly about entertainment and recreations. We also have topics about shopping and various entertainment venues, which are more than those on weekdays.
- There are topics related to food at weekends. However, there is no such a topic on weekdays. Food venues usually appear together with shopping, entertainment or business venues in the local geographic topics on weekdays.
- A few art and science centers appear in the topics on weekdays but they show up as a majority in several topics at weekends. One possible reason could be that these art and science centers are usually closed after 5pm on weekdays, thus people may have no time to visit such centers after the work. Another possible reason could be that it usually takes a long time to visit such museums and art centers so people would like to visit them on weekends.

In summary, the differences between the local geographic topics on weekdays and those at weekends correspond with the differences of the human mobility on weekdays and at weekends. Therefore, our proposed approach can characterize the human mobility patterns.

DISCUSSION

Interesting Observations

From the three experiments in the last section, we observe several interesting findings.

Hospitals & Fitness Centers In our OTopics, we find that hospitals and fitness centers appear in the same topics frequently. There are 9 hospitals and 11 fitness centers in the top 10 venues of the 30 OTopics. 5 hospitals and 6 fitness centers appear together in 5 different topics. One reason of the co-appearances of the hospitals and fitness centers may be that medical workers do exercises frequently. It also may be that patients often go to fitness centers to improve health.

Topics involving Airports We have one topic related to airport on weekdays (WDTopic 7) and one topic related to airport at weekends (WETopic 4). We find significant differences between the other venues of the top 10 venues in these two topics. On weekdays, venues are mainly related to business. For example, The Westin Convention Center and David L. Lawrence Convention Center in WDTopic 7 are big convention centers for conferences. We do not see such venues in WETopic 4. This difference also complies with the human mobility pattern as there are many people taking flights for business purposes on weekdays.

Local Supermarkets vs. Global Supermarkets Giant Eagle

is a supermarket chain in Pittsburgh and Walmart is a global supermarket chain. Both of these supermarkets provide very similar groceries and services. However, they are often in the same topics indicating that perhaps many people go to both of them.

Local Coffee Shop Chain vs. Global Coffee Shop Chain Crazy Mocha is a famous local coffee shop chain in Pittsburgh and Starbucks is a very famous global coffee shop chain. In our local geographic topics, Starbucks appears to be one of the top 10 venues in many topics, while Crazy Mocha is not included by one of them. In our data set, the number of Starbucks is almost 4 times the number of Crazy Mocha; however, the number of check-ins at Starbucks is ten times more than that at Crazy Mocha. Thus, it seems that the customers do not check in at Crazy Mocha as often as they do at Starbucks.

Applications

Next, we also discuss how our proposed approach and results can be used in different applications.

Friend Recommendation Our proposed topic model based approach can be used in friend recommendation. Since each user's trajectory can be described by the topics, the similarity of the topics could be helpful in recommending friends to users. For example, for two PITT students who check in at Pitt's Hillman Library, School of Information Sciences and William Pitt Union frequently, although they may not know each other right now, they may register in the same course, or join the same student organization or interest group in the future, and the probability of being friends will be very high.

Location Recommendation & Prediction Our proposed topic model based approach can also be used in location recommendation. Our proposed topic model is based on the users' trajectories, thus the venues in the same topic are the ones that many people usually go to. Personalized location recommendation should also consider such venues as candidates for recommendations because they are more or less "hot spots" for a group of people. For example, some parents would like to take their children to visit the museums or science centers at weekends. Thus, the topics about entertainment could provide very valuable references since many people have been there. The topic model also can be used in location prediction. If some similar users or friends of a user have been to a very good venue, the user may also go to the venue with high probability. In the case of the entertainment example above, the venues in the entertainment topic may be the next check-in venues for the parents and their children. Even without the topic model based recommendation, the parents still will go to the museums or science centers they have never been to before, as they would likely bring their kids there. Our proposed topic model based approach can still help to predict locations as it can characterize human mobility pattern.

Business Strategy Design Users' trajectory-driven local geographic topics can be helpful in designing business strategies. The topics can help a business owner to find out whether there is any complementary relationship between his venues and other venues and then help him explore the potential locations for a new chain store. It is also helpful for the business owners to identify his competitors' advantages to improve his own business. For example, it could be interesting for the Crazy Mocha to figure out the reasons why it has far fewer check-ins compared to that of Starbucks.

RELATED WORK

Li and Chen present their work of large-scale quantitative analysis of LBSNS in [15]. Their work is very general in analyzing the user profiles, update activities, mobility characteristics, social graphs and attribute correlations.

Cheng *et al*. explore the check-ins to analyze human mobility patterns in the spatial, temporal, social, and textual aspects [17]. Their work uses the global data collected from *Twitter*, which is different from ours, as our data set are local data set from *Foursquare*. Moreover, they do not use topic models to analyze the check-ins.

Noulas *et al*. study the user behavior in *Foursquare* in [5]. They investigate the check-in dynamics in spatio-temporal aspects. However, they do not analyze the relations among the check-ins.

Ferrari *et al*. employ LDA to extract the urban patterns from location-based social networks in [12]. Our work is different from theirs mainly in two ways. First, our topics are based on the users' trajectories, i.e., we use a user's trajectory as a document and a venue in a check-in as a word in the LDA model. However, their work uses a venue and a time slot as a word and a day of the city is a document in their LDA model. Thus, their work focuses on the human mobility pattern during different times within a city and our work is more human centric than theirs as we investigate the topics based on trajectories of large groups of users. The data set used in their work is crawled from *Twitter* but not from *Foursquare* directly. In *Foursquare*, a user can use his *Twitter* account to login and post his check-ins on *Twitter*, but not every user has a *Twitter* account and neither is every user likely to post his check-ins on *Twitter*. Thus, our data is more comprehensive and complete than the *Twitter* data set.

Ferrari and Mamei also investigate the topics based on the user's trajectory in [13]. The data set in [13] is the daily whereabouts of two persons over the period for almost one year. They divide a day into 48 time slots and each time slot lasts for 30 minutes, so the 48 places each day form a document. Thus, their work in [13] is still time based topics, which focuses on a single user's mobility pattern at different times and is thus very different than our approach.

Farrahi and Gatica-Perez's work in [11] also use LDA to discover the routine behaviors. They use the Reality Mining data set [14] from MIT that contains a one-year mobile phone sensor data recording 97 subjects from 2004 to 2005. The routine behaviors in their work are still temporal based topics, which are different from ours, as we do not consider the temporal factors in our model. Besides, the locations in

their work are simply labeled by "Home", "Work", "Other" and "No Reception". Thus, the rich venue information is lost in their work.

Yuan *et al*. also propose a framework to discover regions of different functions in a city by using human mobility among both regions and points of interests in the region in [10]. Their topic model is based on LDA and Dirichlet Multinomial Regression. In their work, they use the GPS trajectory datasets. Besides, their work aims to discover the region topics, which is different from ours.

Cranshaw and Yano employ LDA to distill the proto-neighborhoods from *Foursquare* data set in [9]. In their work, the word is the category of the venue and the document is the check-ins in a region. Regions are small grids that divide space according to the latitude and longitude space. Thus, each region can be described by the topics, which can help to understand the neighborhood.

Chang and Sun analyze users' check-ins in general on *Facebook Place* in [7] and LDA is also used in their work to investigate the user membership in a low-dimensional representation of the place space. Since their work is in general about the check-in analysis so the topic model is only a small part and they only give three topics without analyzing the topics in details as we do. Besides, they do not consider the differences in mobility patterns of users in the weekdays and weekends.

CONCLUSION AND FUTURE WORK

In this paper, we have employed the LDA model to investigate the local geographic topics based on the users' check-ins in *Foursquare*. Since our topics are derived from the trajectories of 32,113 different users, the local geographic topics in this paper indicate the co-check-in among such users. That is, the venues in the same topics are usually co-occurrences in many users' trajectories. The analysis of the local geographic topics also verifies the effectiveness of the model. For example, we can see there are topics describing universities, entertainment venues, etc. Besides, we also explore the spatial features of the top 10 venues in some topics and we find that the venues in the same topic can be either close or far away from each other. Thus, the topics are not limited by the spatial information. Moreover, we study the local geographic topics on weekdays and at weekends, respectively. The differences between them comply with the human routine behaviors; the results also demonstrate the effectiveness of our approach. Furthermore, we discuss several interesting findings of our topics and the applications of the proposed topic model based approach.

One future research direction is to use the topic model in location and friend recommendations. We plan to also investigate approaches other than LDA in discovering the local geographic topics from the large-scale LBSNS data set.

ACKNOWLEDGMENTS

This research has been supported by the US National Science Foundation award IIS-0545912. We would like to thank Mark Steyvers and Tom Griths for providing their MATLAB Topic Modeling Toolbox which is used in this paper.

REFERENCES

1. Zheng, Y. Location-based social networks: Users. *Computing with Spatial Trajectories*, Zheng, Y and Zhou, X., Eds. Springer, 2011.
2. Foursquare Nears 20 Million Users And Crowley Talks About His Co-founder's Recent Departure. `http://articles.businessinsider.com/2012-03-10/tech/31142426_1_foursquare-sxsw-dennis-crowley`.
3. 200M users include location in Facebook posts; company looks to expand location APIs. `http://www.insidefacebook.com/2012/04/05/200m-users-include-location-in-facebook-posts-company-looks-to-expand-location-apis/`.
4. Get tickets for select penguins home games with american eagle student rush. `http://www.ticketmaster.com/promo/u3c7b3?brand=penguins`.
5. Noulas, A., Scellato, S., Mascolo, C. and Pontil, M. Empirical study of geographic user activity patterns in foursquare. In *ICWSM'11*, 2011.
6. Blei, D.M., Ng, A.Y., and Jordan, M.I. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
7. Chang, J. and Sun, E. Location3: How users share and respond to location-based data on social. In *ICWSM'11*, 2011.
8. Cranshaw, J., Schwartz, R., Hong, J.I. and Sadeh, N. The livehoods project: Utilizing social media to understand the dynamics of a city. In *ICWSM'12*, 2012.
9. Cranshaw, J. and Yano, T. Seeing a home away from the home: Distilling proto-neighborhoods from incidental data with latent topic modeling. In *CSSWC Workshop at NIPS'10*, 2010.
10. Yuan, J., Zheng, Y. and Xie, X. Discovering regions of different functions in a city using human mobility and pois. In *KDD'12*, 2012.
11. Farrahi, K. and Gatica-Perez, D. Discovering routines from large-scale human locations using probabilistic topic models. *ACM Transactions on Intelligent Systems and Technology*, 2(1), 2011.
12. Ferrari, L., Rosi, A., Mamei, M. and Zambonelli, F. Extracting urban patterns from location-based social networks. In *LBSN'11*, 2011.
13. Ferrari, L. and Mamei, M. Discovering daily routines from google latitude with topic models. In *CoMoRea'11*, 2011.
14. Eagle, N., Pentland, A. and Lazer, D. Inferring social network structure using mobile phone data. In *Proceedings of the National Academy of Sciences (PNAS)*, volume 106, pages 15274–15278, 2009.
15. Li, N. and Chen, G. Analysis of a location-based social network. In *CSE'09*, 2009.
16. Griffiths, T. and Steyvers, M. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(supple.1):5228–5235, 2004.
17. Cheng, Z., Caverlee, J., Lee, K. and Sui, D.Z. Exploring millions of footprints in location sharing services. In *ICWSM'11*, 2011.

Crowd-sourced Cartography: Measuring Socio-cognitive Distance for Urban Areas based on Crowd's Movement

Shoko Wakamiya
University of Hyogo
Japan
ne11n002@stshse.u-hyogo.ac.jp

Ryong Lee
National Institute of Information and Communications Technology, Japan
lee.ryong@gmail.com

Kazutoshi Sumiya
University of Hyogo
Japan
sumiya@shse.u-hyogo.ac.jp

ABSTRACT
On behalf of the rapid urbanization, urban areas are gradually becoming a sophisticated space where we often need to know ever evolving features to take the most of the space. Therefore, keeping up with the dynamic change of urban space would be necessary, while it usually requires lots of efforts to understand newly visiting and daily changing living spaces. In order to explore and exploit the urban complexity from crowd-sourced lifelogs, we focus on location-based social network sites. In fact, due to the proliferation of location-based social networks, we can easily acquire massive crowd-sourced lifelogs interestingly indicating their experiences in the real space. In particular, we can conduct various novel urban analytics by monitoring crowd's experiences in an unprecedented way. In this paper, we particularly attempt to exploit crowd-sourced location-based lifelogs for generating a socio-cognitive map, whose purpose is to deliver much simplified and intuitive perspective of urban space. For the purpose, we measure socio-cognitive distance among urban clusters based on human mobility to represent accessibility of urban areas based on crowd's movement. Finally, we generate a socio-cognitive map reflecting the proposed socio-cognitive distances which have computed with massive geo-tagged tweets from Twitter.

Author Keywords Socio-cognitive Map, cartography, urban analytics, location-based social network

ACM Classification Keywords J.4 [Social and Behavioral Sciences]: Sociology; H.1.2 [User/Machine Systems]: Human factors processing; H.2.8 [Database Applications]: Spatial databases and GIS

General Terms Experimentation, Human Factors, Measurement

INTRODUCTION

Urban space is a complicated mixture, which includes a variety of elements; physical objects such as local facilities and landmarks, natural phenomena like climates and disasters, and social activities such as cultural or political events, etc. In such a complex space, we are always required to conduct various location-based decision makings from looking for a restaurant for daily lunch to exploring a new dwelling. In such situations, final decisions would be often elicited depending on individual experiences in an urban area or limited knowledge about the area; if a person frequently visits a city, s/he intuitively thinks that the city is more familiar than other cities where s/he has been less. Therefore, based on personal experiences to a sophisticated urban structure, we become to have a bound image of the urban space and eventually make an unsatisfactory choice. Here, we regard such individually distorted image of urban areas as a cognitively recognized urban space.

For instance, let's assume a situation where a person is looking for a place to live with his family. He would like to find an ideal place which can meet various demands; not only accessibility to his workplace in terms of public transportation, but also convenience for shopping, safety, cleanliness, educational environment of the place, etc. For this, he may first consult a general reference map for making a short list of candidate places to live by considering the accessibility based on transportation convenience. General reference maps like Google Maps[1] would be the first step to look up general features of urban space; cities, roads, railways, local facilities, etc. Especially, a travel time map[2] or commute map[3] can show the adjacent neighborhood of the workplace in terms of not only geographical proximity but also the accessibility based on transportation convenience in the urban space. However, such a map just shows a commutable area which includes lots of candidate places to move. Therefore, in order to find out complex and dynamic local characteristics, general reference maps are not enough to provide appropriate answers for much sophisticated questions such as 'Which place can give better educational environment for his children?' Hence, in order to examine this kind of local characteristics or knowledge, we further need to search for the Web, variously local statistics information by public administration, word-of-mouth from acquaintances, etc. However, it is not easy to acquire the local characteristics without huge costs and efforts as illustrated in Figure 1 (a).

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

[1] Google Maps: https://maps.google.com/.
[2] More travel-time maps and their uses by mysociety.org: http://old.mysociety.org/2007/more-travel-maps/.
[3] Commute map by Trulia: http://www.trulia.com/local\#commute/new-york-ny.

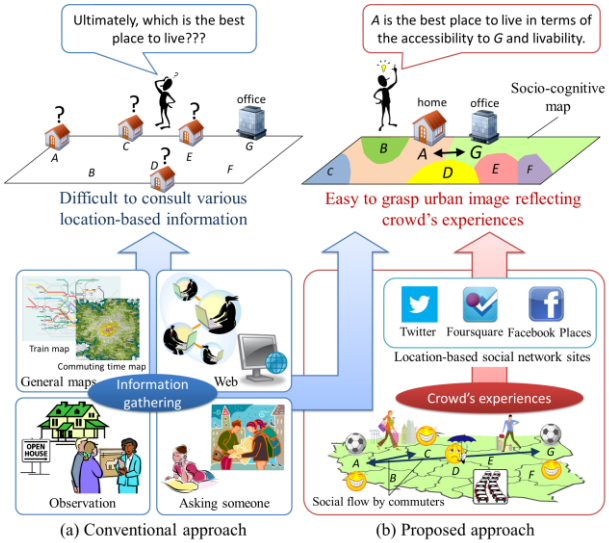


Figure 1. Motivation for socio-cognitive map generation: (a) conventional approach for exploring local information and (b) our approach for cartography based on crowd's experience

In particular, this paper will focus on the novel crowd-sourced lifelogs which are represented by Twitter; we will explore new values of the massive urban people's experiences as a source to explore various urban dynamics and characteristics relevant to crowd's lifestyles utilizing the recent location-based social network sites such as Twitter[4], Foursquare[5], and Facebook Places[6]. Furthermore, with the extracted crowd's lifestyles in an urban area, we aim to generate an integrated map which can represent various local characteristics in an urban space as shown in Figure 1 (b). Particularly, we focus on the accessibility by measuring socio-cognitive proximity based on crowd's movements in an urban space.

Furthermore, we present a method to generate a socio-cognitive map as a kind of thematic map based on Twitter-based crowd's movements. Obviously, cartography based on crowd-sourced lifelogs would be an interesting and important challenge to provide much useful local information. Generally, in a specialized map called cartogram [7], mapping variables such as travel time, population or incomes are substituted for land area or distance; the geometry or space of the map would be significantly distorted to represent the information of this alternate variable. For the crowd-sourced cartography, we first look for significant places as social urban clusters in an urban space based on the density of crowd through location-based social networks. Next, we measure influential strength of each social urban cluster for an area variance. Then, we compute socio-cognitive distance between the clusters based on the crowd's movements for a distance variance. Finally, we generate a socio-cognitive map by projecting urban clusters on two-dimensional space by means of MDS (Multi-Dimensional Scaling) as well as by emphasizing urban clusters based on their influential strengths with a Weighted Voronoi Diagram.

The contributions of this work are summarized as follows.

- We generated a socio-cognitive map by exploiting crowd-sourced lifelogs on location-based social networks.
- We defined crowd-sourced cognitive distance by expanding the concept of cognitive distance.
- We developed a technical method to generate socio-cognitive cartogram.

In this remainder of this paper, Section 2 describes our research model with recent related work. Section 3 presents the detailed procedure of our socio-cognitive map generation method using Twitter-based crowd's movement lifelogs. Section 4 provides our experimental results. Finally, Section 5 concludes this paper with a brief description of future work.

COMPUTING SOCIAL URBAN STRUCTURE THROUGH LOCATION-BASED SOCIAL NETWORK

In this section, we describe our research model for socio-cognitive map generation. First, we explain our research model to extract crowd's experiences from location-based social networks. Then, we briefly review some related work.

Research Model

In this work, in order to represent complex and dynamic urban space, we attempt to generate a socio-cognitive map of urban space by exploiting local crowd's experiences. For this, we utilize crowd's lifelogs publicly shared on recent location-based social network sites. In order to monitor crowd's experiences using social network sites, we modeled crowd's experiential features. In general, lifelogs on most location-based social network sites are consisted of a set of metadata such as user ID, timestamp, location information and textual message. In case of Twitter, we can extract further useful metadata such as reply words in a textual message, hashtags, followers or following relationships, retweets, links to external media, etc. On the basis of such metadata, we can first define personal experiential features consisting of five indicators relevant to individual experience; 1) user's existence in an urban cluster which is represented by user ID and location information, 2) user's activity in terms of publishing tweets and moving in an urban cluster which is computed by using user ID, timestamp and location information, 3) user's sentiment which can be computed by determining sentimental words or the ratio of positive or negative words in a textual message, 4) user's interest which is represented based on textual hints such as topic keywords and hashtags as well as links to external media like Web pages, photos, video clips,

[4] Twitter: http://twitter.com/.
[5] Foursquare: https://foursquare.com/.
[6] Facebook Places: http://www.facebook.com/about/location.
[7] Cartogram: http://www.ncgia.ucsb.edu/projects/Cartogram_Central/types.html

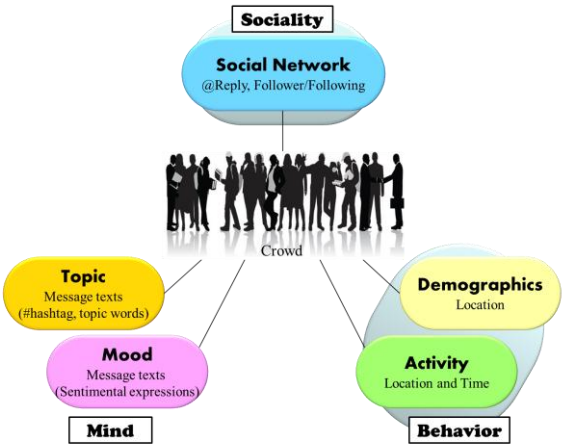


Figure 2. Crowd experiential features extractable from location-based social networks

etc., and 5) user's relationships and interactions with other users which are computed based on followers or following relationships, replies and retweets.

Next, on the basis of the five types of indicators above constituting personal experiential features, we can define crowd experiential features as shown in Figure 2. In sum, by aggregating individual existences in an urban cluster, we can easily grasp demographics such as crowd's population and density populated points in the urban cluster [11]. Individual activities can represent crowd's activity and movements in terms of congestion and activation. Users' sentiments show a mood in an urban cluster [12]. For instance, when lots of people in an urban cluster relatively feel happier than other clusters, we can expect that the mood of the urban cluster reflected on the location-based social networks would be also positive. In addition, the aggregated personal interests could be regarded as crowd's topics and social trends. Crowd's social networks are clearly connected with social communications and we can grasp socio-geographical relationships among urban clusters through human relationships. In this paper, we will measure a socio-cognitive distance based on accessibility between urban clusters by especially focusing on crowd's movements as one of the indicators.

Related Work

Due to the fast urbanization in the modern age, it is not easy to draw images of urban areas in mind, especially, to unfamiliar cities. For this problem, lots of interdisciplinary studies computationally to investigate urban characterization have been proposed. In our previous work [10, 14], we proposed a method to characterize urban areas by detecting significant latent patterns of crowd's behavior exploiting geo-tagged tweets extractable from Twitter. Yuan et al. [15] proposed a framework for discovering different functional regions such as educational areas, entertainment areas, and regions of historic interests in a city using both human mobility based on taxies trajectory and POIs. Kurashima et al. [8] developed a system for browsing actual experiences related to a specific location and time period extracted by means of association rules from blog articles. Advanced from our previous work, we focus on measuring proximities cognitively recognized among urban areas based on massive crowd's experiences through location-based social network sites for generating a socio-cognitive map which can intuitively represent complicated urban structure.

In order to represent specific information with maps, thematic maps are required rather than general reference maps. Therefore, lots of technologies for cartography satisfying various purposes have been studied recently. Grabler et al. [6] presented a method for automatically generating destination maps to navigate to a given location from anywhere in a given area of interest. The system decided map elements using both vision-based image analysis and web-based information extraction techniques. This method aimed to create a navigational map in much simplified and personalized way. In addition, there are interesting and cognitive maps like cartogram which can represent location-based statistical information by deforming area or distance. Shu et al. [13] developed a system to generate animated map by means of interactions with a user and the system. Mislove et al. investigated demographics [11] and mood throughout a day in the U.S. [12] by examining Twitter and visualized hourly deformed maps in terms of Twitter users' mood observable from their textual messages. In our proposed method, we create a socio-cognitive map by computing an influential strength of an urban cluster and relations among urban clusters based on crowd's mobility.

GENERATING A SOCIO-COGNITIVE URBAN MAP

In this section, we describe our map generation method in an order as shown in Figure 4, which begins with collecting crowd's lifelogs from location-based social network. Finally, we generate a map of our interest which aims to deliver an intuitive urban structure focusing on practical proximity between urban clusters; that is, accessibility, by investigating crowd's movements.

Collecting Crowd's Movements from Twitter

We first gather geo-tagged tweets from Twitter to monitor crowd's movements in an urban space as shown in Figure 4 (1). However, it takes a considerable amount of efforts to acquire a significant number of geo-tagged tweets due to practical limitation of an open API[8] provided by Twitter which only supports the simplest near-by search based on a specified center location and a radius and obtains a limited number of tweets. In order to overcome this problem, we developed a geographic tweets gathering system, in our previous work [9], which can monitor crowd behavior for a specific region of any size depending on the density of massive geographic microblogs for overcoming these

[8] Twitter Open API. http://apiwiki.twitter.com/Twitter-Search-API-Method%3A-search.

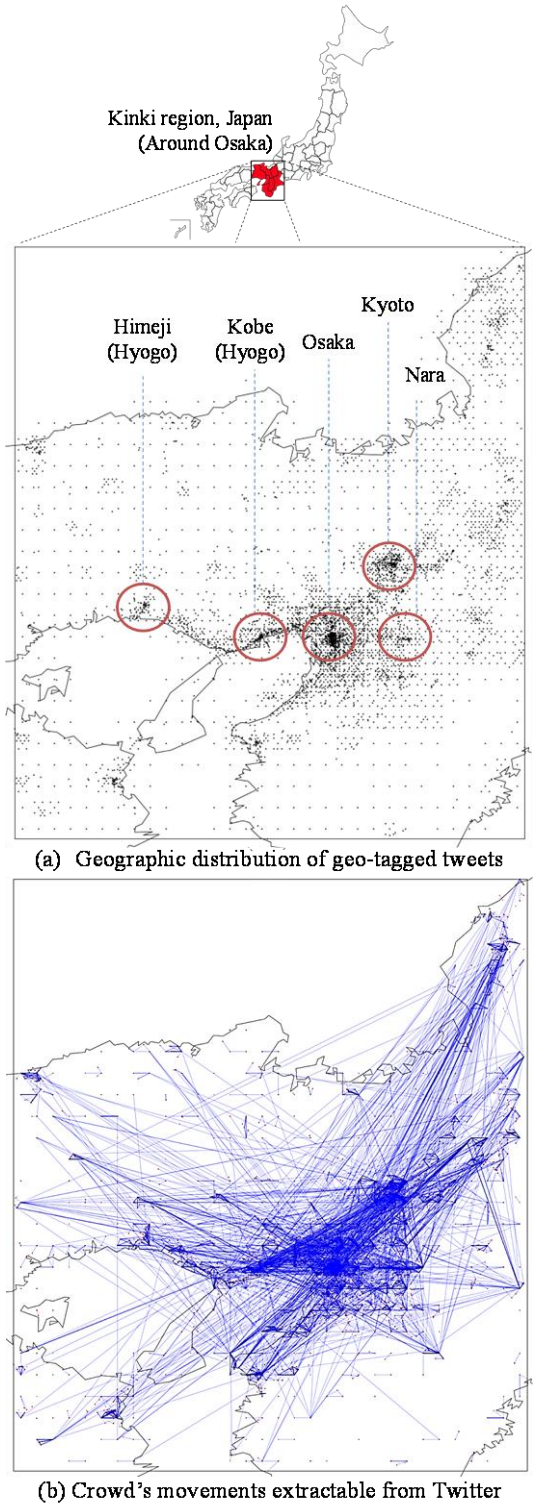


Figure 3. (a) Geographic distribution of geo-tagged tweets and (b)crowd's movements monitored through Twitter (Kinki region in Japan: longitude range= [134.122433, 136.337186], latitude range= [33.810804, 36.785050])

limitations and carry on monitoring of any size of user-specified regions.

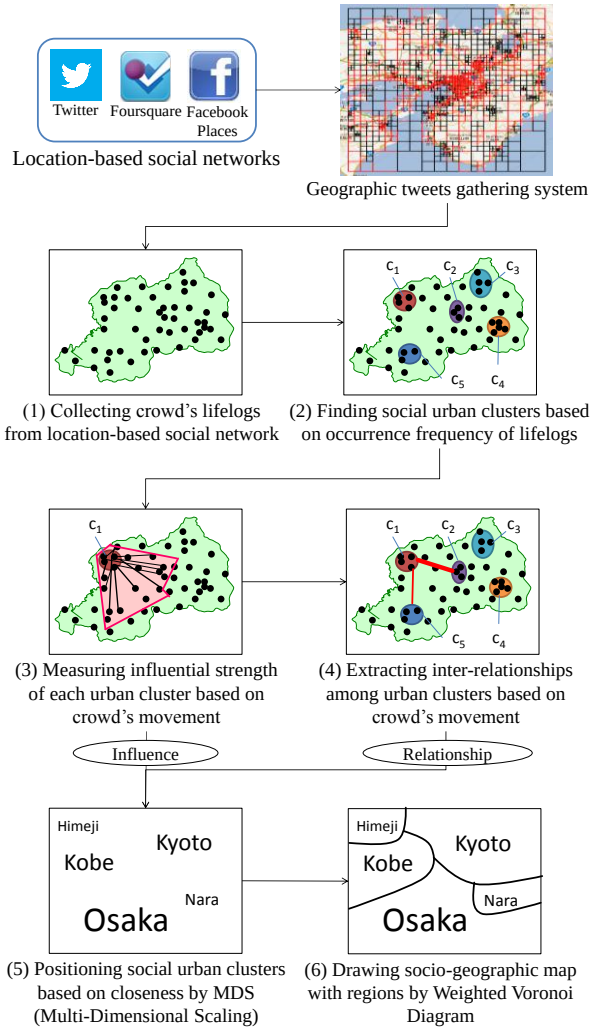


Figure 4. Procedure for generating cognitive map based on crowd's movements

Based on the geo-tagged tweets, we extract crowd's moving segments by exploiting primary metadata of geo-tagged tweet; user ID, timestamp, and location information. After we determine when and where each tweet was written, we can plot their location points on the map as shown in Figure 3 (a). Furthermore, in order to effectively search moving segments with the unified dataset, we assemble the tweets based on user ID and sort each user's tweets in the order of the timestamp. Consequently, Figure 3 (b) shows moving segments of crowds observed through Twitter.

Locating Urban Clusters

Most thematic maps are generated by emphasizing some landmarks or characteristic areas according to various purposes of cartographers respectively. We will also generate such kind of cognitive map, where some geographic features appeared on the maps are appropriately selected. For this, we locate social urban clusters by utilizing the density of crowd as shown in Figure 4 (2).

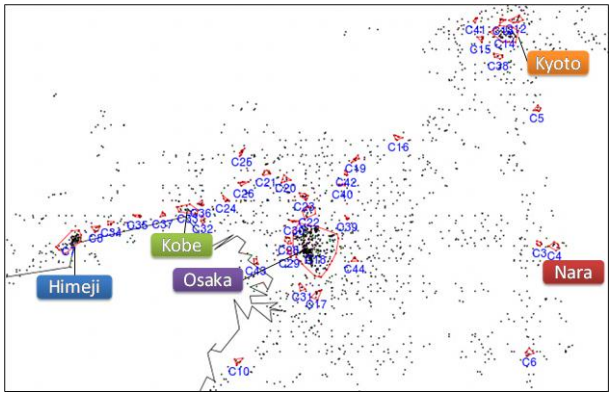


Figure 5. Urban clusters generated based on crowd's lifestyles

However, in this paper, we are based on massive number of crowd's lifelogs found on Twitter, hence, it would require unbearable computational efforts to find out social urban clusters. Therefore, we need to reduce the data size in much smaller and compact size without loss of essential quality of the original data. For this, we adopted the NNClean algorithm [3] to split the data into two classes of high-frequency and low-frequency parts.

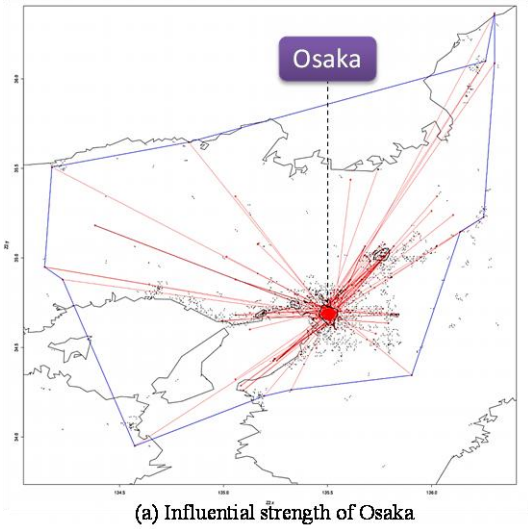
Then, we look for social urban clusters with the ideally reduced dataset. In order to find high-density areas, we apply one of conventional clustering methods; DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm [4]. The benefit of this algorithm is the ability to deal with non-Gaussian distributed data because it features a cluster model called density-reachability. This algorithm connects points that satisfy a density criterion, in the original variant defined as a minimum number of points *MinPts* within a specified radius *radius*. A cluster consists of all density-connected points which can form a cluster of an arbitrary shape.

In the experiment, we generated a socio-cognitive map with 44 social urban clusters located by empirically setting *MinPts* and *radius* to 6 and 0.0065, respectively. Then, we eventually represented the clusters by convex-hull based boundary polygons [1] which center points in a bounding convex as shown in Figure 5.

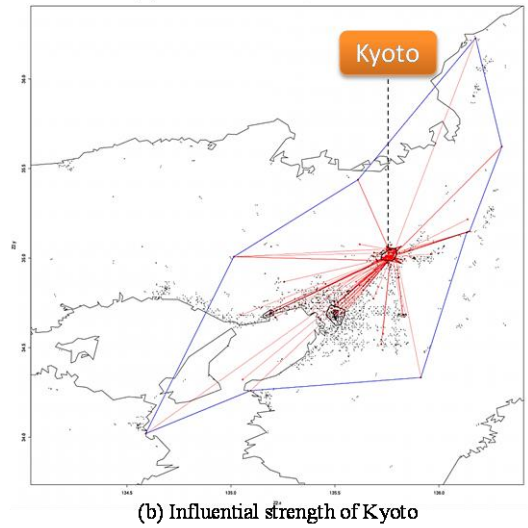
Measuring Influential Strength of an Urban Cluster

Next, by examining where people come and go out to an urban cluster, we can know its influential strength in an urban space as shown in Figure 4 (3). As shown in Figure 6, it would be possible to assume that the influential strength is the overall area including all of connected places with the urban cluster. The measured influential strength of each social urban cluster is utilized when visualizing a socio-cognitive map.

Here, we illustrate Figure 6 (a) and (b) which show urban clusters where influences are the maximum in our finding. Its effective ranges cover the most major landmark areas in our experimental area of Kinki region in Japan (including



(a) Influential strength of Osaka



(b) Influential strength of Kyoto

Figure 6. Influential strengths of urban clusters

Osaka, Kyoto and Kobe). From this result, we confirmed that the influential strength of an urban cluster can be simply computed by the total number of moving segments.

Calculating Cognitive Distance between Urban Clusters

Then, we measure cognitive distances between urban clusters respectively as shown in Figure 4 (4). We assume that if the shorter the physical distance between urban clusters is and the more frequently crowds move the clusters, then the closer the clusters would be regarded by people. Therefore, we consider that urban clusters which are closely associated with each other should be projected closely on a cognitive map. On the basis of the hypothesis, we made a formula to calculate a cognitive distance between urban clusters as follows:

$$CogDist(c_i, c_j) = w_1 \cdot EucDist(c_i, c_j) + w_2 \cdot ExpDist(c_i, c_j) \quad (1)$$

$$(w_1 + w_2 = 1.0,\ w_1, w_2 \geq 0)$$

$$ExpDist(c_i, c_j) = 1/(\#MovSeg(c_i, c_j) + 1) \quad (2)$$

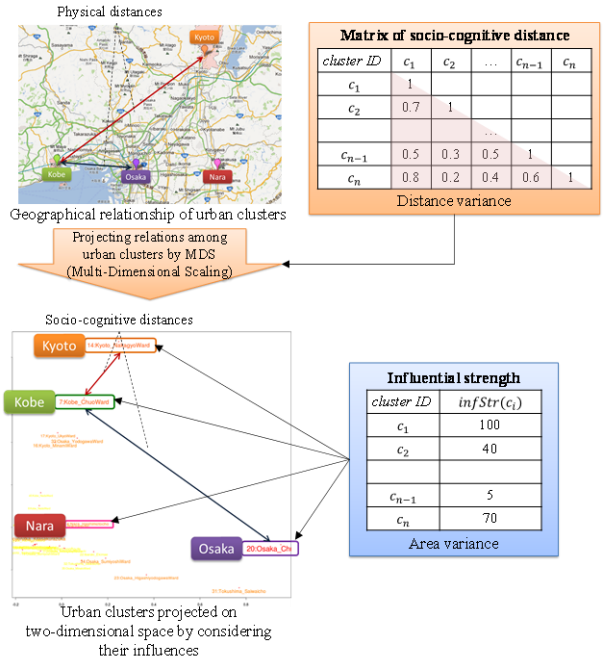


Figure 7. Projection of urban clusters in terms of area and distance based on MDS

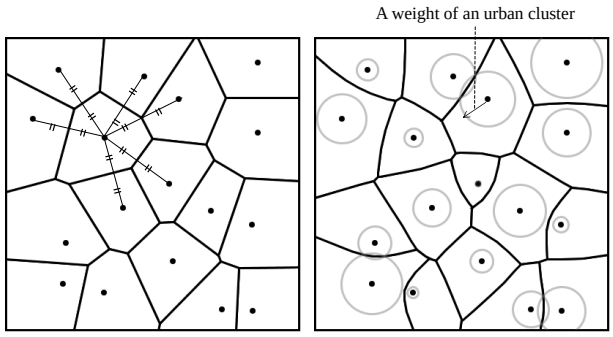
where three functions, *CogDist*, *EucDist* and *ExpDist*, calculate distances between urban clusters (c_i, c_j) in terms of cognitive, physical, and experiential, respectively. Specifically, the function *CogDist* is calculated by *EucDist* and *ExpDist*. The function *EucDist* calculates normalized Euclid distance between urban clusters, and the function *ExpDist* calculates normalized experiential distance between them based on the quantity of crowd's movements given by a function *#MovSeg* which counts the number of moving segments between the clusters.

The values computed by *EucDist* and *ExpDist* are weighted based on given values, w_1 and w_2, respectively. The weight values can be freely set on a user's purpose for generating a cognitive map. For example, if a user wants to generate a cognitive map by emphasizing on the crowd's movements, s/he can set a high weight to w_2. In the experiment, we show cognitive maps generated with different pairs of the weight values.

Projecting Closeness between Urban Clusters

Next, we plot the computed closeness among urban clusters based on crowd's movements as shown in Figure 4 (5). In order to intuitively represent a socio-cognitive distance of urban clusters by measuring closeness cognitively recognized among them, we need to appropriately allocate the clusters on a socio-cognitive map.

In this paper, as shown in Figure 7, we decided to apply Multi-Dimensional Scaling (MDS) [7], which allocates given dataset of multi-dimensional space into a low-dimensional space by considering similarities or dissimilarities in the dataset. In sum, it can allocate two



(a) Voronoi Diagram (b) Weighted Voronoi Diagram

Figure 8. Examples of Voronoi Diagrams

urban clusters in the neighborhood if the similarity between them is high. In contrast, if the similarity between them is low, it allocates them far away. Specifically, the MDS algorithm starts with a matrix of item-item similarities, then assigns a location to each item in *N*-dimensional space, where *N* is specified a priori. In the experiment, we mapped social urban clusters in a two-dimensional space.

We also labeled names of generated clusters by a Reverse Geocoding service[9] which can translate a given location coordinate into a textual place name. For each cluster, we obtain its representative place name by using this service.

Drawing Socio-cognitive Regions

In Section 3.5, we plotted social urban clusters which are labeled by approximate geographic names. However, it is hard to regard the result as a cognitive map because the representation can just show socio-cognitive closeness of urban clusters. Urban clusters on a socio-cognitive map would be better allocated with region-based space partitioning as shown in Figure 4 (6). For the purpose, we applied a Weighted Voronoi Diagram [2] to the result of MDS. Generally, a Voronoi Diagram depicted in Figure 8 (a) is known as an algorithm for partitioning a space by drawing a line between two points keeping with the same distance. However, urban clusters do not have the same influential strength; we can solve this problem by means of a Weighted Voronoi Diagram, where each cell can have a weight extending its size compared to normal Voronoi Diagram as shown in Figure 8 (b).

EXPERIMENT

In this section, we describe our experiment to generate a socio-cognitive map. For this, we collect massive geo-tagged tweets for a day from Twitter in an area of Japan. Then, we located social urban clusters based on the geographic distribution of crowd's lifelogs and measured an influential strength of each urban cluster based on crowd's moving segments. Next, we computed cognitive relationships among the clusters in terms of physical and social experiential distances. Finally, we generated a socio-

[9] Google Reverse Geocoding API. https://developers.google.com/maps/documentation/ geocoding/#ReverseGeocoding.

Figure 9. Positional relationship of urban clusters

cognitive map based on the extracted crowd's moving patterns.

Dataset

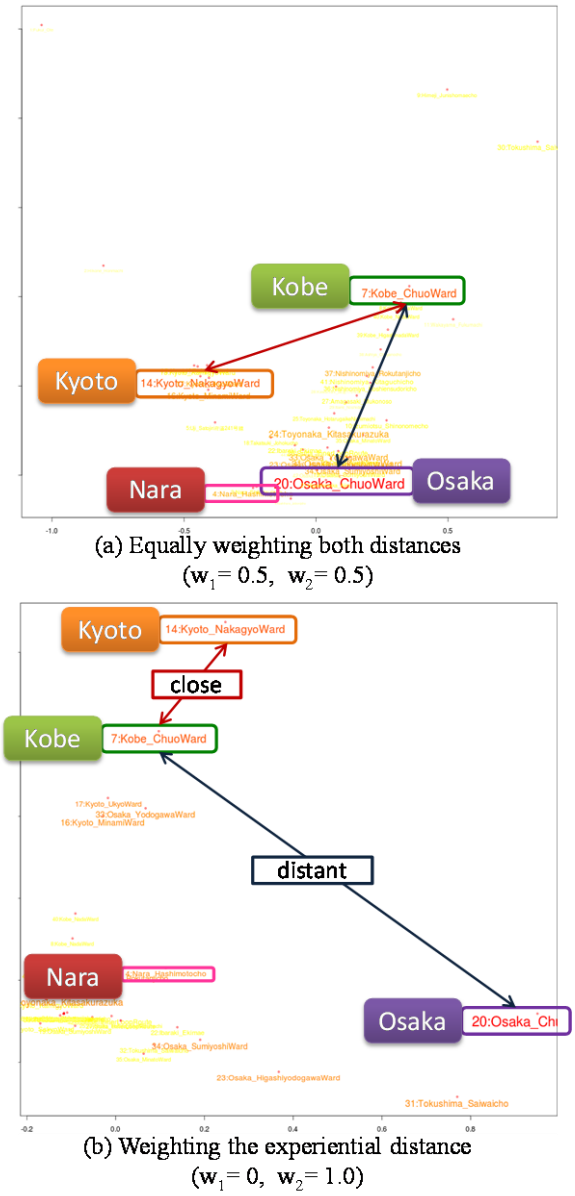
We collected 157,097 geo-tagged tweets from 25,674 distinct users in a day, April 23rd, 2012 in a surrounding area including Kobe, Osaka, Kyoto and Nara in Japan (longitude range = [134.122433, 136.337186], latitude range = [33.810804, 36.785050]) from Twitter using the geographical tweet gathering system [9]. From each geo-tagged tweet, we utilized spatio-temporal clues; user ID, timestamp, and location coordinate for monitoring crowd's movements in an urban space.

Generated Socio-cognitive Map

We first located convex-hull based urban clusters by applying DBSCAN algorithm with location points of the dataset reduced by NNClean method as shown in Figure 5. Next, we determined the influential strengths of the urban clusters for showing their relations by computing crowd's moving segments. Figure 6 shows examples of urban clusters measured strong influence on wide surrounding areas. From this result, we confirmed that we could capture the influential strength of each urban cluster in a simple way through the quantity of crowd's moving segments relevant to each urban cluster.

Then, we calculated socio-cognitive distances between social urban clusters for counting crowd's moving segments between the clusters and projected the clusters with their relationships to socio-cognitive maps by applying the MDS algorithm. Here, we can freely set weight values for physical distance and experiential distance which bringing the total to 1.0. As aforementioned, a user can easily adjust the degree of his/her requirements.

We show the plotted urban clusters which set different weight values in the formula (1) by MDS algorithm as illustrated in Figure 10. Texts appeared on a two-dimensional space are local address at the center of each social urban cluster, and their sizes and colors mean their influential strengths; the bigger the size of a cluster's label is and the darker its color is, the more influential the cluster

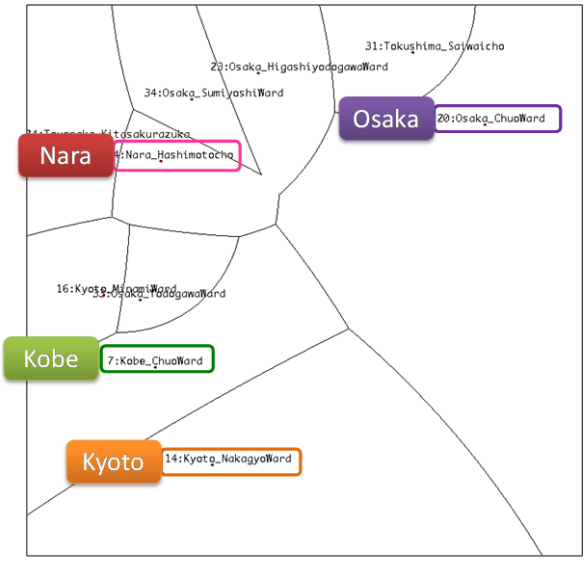


(a) Equally weighting both distances ($w_1 = 0.5$, $w_2 = 0.5$)

(b) Weighting the experiential distance ($w_1 = 0$, $w_2 = 1.0$)

Figure 10. Social urban clusters projected by MDS

is regarded. In detail, Figure 10 (a) shows a result based on socio-cognitive distance measured with the same weight (=0.5) to both physical and experiential distances among urban clusters. In this case, geographically close urban clusters were relatively aggregated closely after the MDS computation. In contrast, Figure 10 (b) is the result by attaching a high weight (= 1.0) to experiential distance based on crowd's moving segments between urban clusters. Thus, we can grasp localized social relationships reflecting crowd's movements. Furthermore, in both figures, we interestingly found that socio-cognitive distances relevant to Kobe are different from the physical distances in the real world as shown in Figure 9; the physical distance between Kobe and Osaka is closer than the one between Kobe and Kyoto as shown in Figure 9, but the socio-cognitive distance between Kobe and Osaka is measured more distant



Socio-cognitive map generated by weighting the experiential distance ($w_1 = 0$, $w_2 = 1.0$)

Figure 11. Generated a socio-cognitive map

than the one between Kobe and Kyoto as shown in Figure 10 (b).

Next, we allocated regions for each urban cluster just plotted on the two-dimensional space. As shown in Figure 11, we can successfully generate a socio-cognitive map consisting of the top 10 urban clusters ranked by their influential strengths by means of a Weighted Voronoi Diagram. As shown in Figure 11, we can obtain a socio-cognitive map having regions of urban clusters in terms of crowd's movements. Here, we can find two influential urban clusters; ChuoWard, Osaka (cluster ID is 20) and NakagyoWard, Kyoto (cluster ID is 14). Subsequently, we explain an interesting result, which is different from the real space. For example, regions of clusters in Osaka mostly covered a large part on this map. As described above, ChuoWard, Osaka (cluster ID is 20) and NakagyoWard, Kyoto (cluster ID is 14) are close to each other. In addition, there are other clusters in Osaka such as SumiyoshiWard (cluster ID is 34), YodogawaWard (cluster ID is 33), etc. are close to the cluster in Nara, Hashimotocho (cluster ID is 4). Finally, we were able to take advantages of the representation based on the Weighted Voronoi Diagram which can help us understand more intuitive and simplified understanding among urban clusters.

CONCLUSIONS

In this paper, we proposed a method to generate socio-cognitive map by measuring influential strength of a social urban area and examining social relationships among social urban areas based on crowd's movements monitored by exploiting geo-tagged tweets over Twitter.

In future work, we will conduct on meaningful socio-cognitive map generation by further observing crowd's experiences extractable from social networks. This method can apply not only with geo-tagged tweets but also with crowd's lifelogs over contemporary location-based social networks.

ACKNOWLEDGMENTS

This research was supported in part by the Microsoft Research IJARC Core Project and Grant-in-Aid for JSPS Fellows 24.9154 from the Ministry of Education, Culture, Sports, Science, and Technology of Japan.

REFERENCES

1. A. M. Andrew. Another efficient algorithm for convex hulls in two dimensions. *Information Processing Letters*, 9(5):216-219, 1979.
2. F. Aurenhammer. Voronoi diagrams-a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23(3):345-405, 1991.
3. S. Byers and A. Raftery. Nearest-neighbour clutter removal for estimating features in spatial point processes. *Journal of the American Statistical Association*, 93:577-584, 1998.
4. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of the Second Intl. Conference on Knowledge Discovery and Data Mining*, pp. 226-231, 1996.
6. F. Grabler, M. Agrawala, R. W. Sumner, and M. Pauly. Automatic generation of tourist maps. *ACM Trans. Graph.*, 27(3):100:1-100:11, Aug. 2008.
7. J. Kruskal and M. Wish. Multidimensional scaling. In Sage University Papers on Quantitative Applications in the Social Sciences, pp. 07-011, 1978.
8. T. Kurashima, T. Tezuka, and K. Tanaka. Blog Map of Experiences: Extracting and Geographically Mapping Visitor Experiences from Urban Blogs. In *Proc.of the 13th Intl. Conference on Web Information System Engineering*, pp. 496-503, 2005.
9. R. Lee, S. Wakamiya, and K. Sumiya. Discovery of unusual regional social activities using geo-tagged microblogs. *World Wide Web Special Issue on Mobile Services on the Web*, 14(4):321-349, 2011.
10. R. Lee, S. Wakamiya, and K. Sumiya. Urban area characterization based on crowd behavioral lifelogs over Twitter. *Personal and Ubiquitous Computing*, pp. 1-16, 2012.
11. A. Mislove, S. Lehmann, Y. Y. Ahn, J. P. Onnela, and J. N. Rosenquist. Understanding the Demographics of Twitter Users. In *Proc. of the 5th Intl. AAAI Conference on Weblogs and Social Media*, pp. 133-140, 2011.
12. Pulse of the Nation: U.S. Mood Throughout the Day inferred from Twitter: http://www.ccs.neu.edu/home/amislove/twittermood/
13. H. Shu, C. Qi, and G. Edwards. Computing geographical reality with animated map language. In *Proc of the 16th Intl. Conference on Artificial Reality and Telexistence-Workshops*, pp. 52-56, 2006.
14. S. Wakamiya, R. Lee, and K. Sumiya. Crowd-based urban characterization: extracting crowd behavioral patterns in urban areas from Twitter. In *Proc. of the 3rd ACM SIGSPATIAL Intl. Workshop on Location-Based Social Networks*, pp. 10:1-10:9, 2011.
15. J. Yuan, Y. Zheng, and X. Xie. Discovering Regions of Different Functions in a City Using Human Mobility and POIs. In *Proc. of the 18th ACM SIGKDD Conference. on Knowledge Discovery and Data Mining*, pp. 133-140, 2012.

Mining the Semantics of Origin-Destination Flows using Taxi Traces

Wangsheng Zhang, Shijian Li, Gang Pan
Department of Computer Science, Zhejiang University
{zws10, shijianli, gpan}@zju.edu.cn

ABSTRACT

Origin-destination(OD) flows reflect both human activity and urban dynamic in a city. However, our understanding about their patterns remains limited. In this paper, we study the GPS traces of taxis in a city with several millions people, China and find that there are significant patterns under the OD flows constructed from taxis' random motion. Our spatiotemporal analysis shows that those patterns have close relationship with the semantics of OD flows, hence we can mine the semantics of OD flows from raw GPS trace data. The approach we proposed offers a novel way to explore the human mobility and location characteristic.

Author Keywords Urban computing, GPS trace, spatiotemporal analysis, LBSN

ACM Classification Keywords I5.2 [Patter Recognition]: Pattern analysis

General Terms Algorithms, Experimentation

INTRODUCTION

In recent years, as advanced technologies in sensor and communication, such as GPS and 3G, make massive urban data collecting and processing feasible, ubiquitous sensing has been widely applied in various areas(city planning [6], traffic engineering [9], public health [2, 7], and so on) to enable us better understand and coordinate the relationship between human and city. Location is a kind of critical information for building smart environments from smart vehicles to smart cities [13, 15, 18]. It helps bridge the gap between the physical world and cyber social network. People can expand their social structure with the new interdependency derived from their locations. These kinds of location-embedded and location-driven social structures are known as location-based social networks(LBSN) [22]. It can be used for location-based services, and also reveals mobility information of local residents.

One of the most important information source of LBSN that represent the relationship among communities in a city is origin-destination (OD) flows, which count the number of individual movements between locations in city. OD flows reflect not only human activity but also urban dynamic and they are widely used in city planning and traffic engineering. However, our knowledge about their patterns remains limited, partly due to the inefficient and expensive census-based methodologies. With the help of pervasive computing devices(mobile phone, travel card, GPS, and so on), we can improve our ability to gather and analyze raw data about OD flows. As a kind of frequently-used public vehicle which conveys passengers to location of their choice, taxi's trace corresponds precisely to individual movement. Hence it is a good data source for estimating OD flows.

In this paper, we first estimate OD flows from the GPS traces of taxis and find some significant patterns under those OD flows via clustering. Then we do spatiotemporal analysis to those patterns and reveal that they have close relationship with the semantics of OD flows. After that we propose a method to mine the semantics of OD flows via those relationship and execute our method on real data.

Based on the above steps, the main contributions of this paper are:

- We propose a new way to estimate the OD flows among locations in a city. Previous researches on OD flows mainly rely on inefficient and expensive census-based methodologies, limited our knowledge about OD flows. As a kind of frequently-used public vehicle, taxi's trace corresponds precisely to individual movement. It is a cheap and efficient data source for estimating OD flows;
- We find that there are significant patterns under OD flows and those patterns have close relationship with the semantics of OD flows. For example, the commute flow (Station to Market) aggregate in early morning while the transfer flow (Station to Station) is flat distributed in day-time;
- We exploit the relationship observed to mine the semantics of OD flows. According to the relationship, we designed three types of feature vectors extracted from taxi traces data. The best type of feature vector achieves a recognition accuracy of 83.7% using Neural Network.

The remainder of this paper is organized as follows. In the next section we review the related work. In the third section we descript the taxi traces data set we used. In the fourth section we estimate the OD flows from taxis' trace data and analyze the patterns under those OD flows. In the fifth section we mine the semantics of those OD flows and use

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

this knowledge to infer location characteristic. Finally, our concluding remarks are given.

RELATED WORK

In this section, we briefly review the related works on human mobility, location-based social networks and taxi traces data.

Recent researches have revealed that there are significant patterns under human mobility. Gonzalez *et al.* [5] find that human traces show a high degree of temporal and spatial regularity, each individual being characterized by a time independent characteristic travel distance and a significant probability to return to a few highly frequented locations. Jiang *et al.* [8] find that the human mobility pattern is mainly attributed to the underlying street network. The goal-directed nature of human movement has little effect on the overall traffic distribution. Calabrese *et al.* [1] use an algorithm to analyze opportunistically collected mobile phone location data and estimate weekday and weekend travel patterns of a large metropolitan area with high accuracy.

Human mobility data also have close relation with social networks. Eagle *et al.* [4] show that data collected from mobile phones have the potential to provide information about the relational dynamics of individuals. Cranshaw *et al.* [3] examine the traces of users of a location sharing social network for relationships between the users' mobility patterns and structural properties of their underlying social network.

As a kind of float sensors in city, taxis attract many researchers' attentions. Veloso *et al.* [17] present a spatiotemporal analysis of taxis GPS traces collected in Lisbon, Portugal and discuss the taxi driving strategies and respective income. They also carry out the analysis of predictability of taxi trips for the next pick-up area type given history of taxi flow in time and space [16]. Other researchers propose many useful ideas based on taxi. Zheng *et al.* [23] detect flawed urban planning using the GPS traces of taxis traveling in urban areas and find that pairs of regions with salient traffic problems and the linking structure as well as correlation among them. Zhang *et al.* [21] propose a method to discover anomalous driving patterns from taxi's GPS traces, targeting applications like automatically detecting taxi driving frauds or road network change in modern city. Li *et al.* [11] develop an improved ARIMA-based prediction method to forecast the spatiotemporal distribution of passengers in urban environment. Li *et al.* [10] present a trip analysis system which identifies the travel mode and purpose of the trips sensed by mobile devices and provides trip summaries and insights to mobile subscribers.

One major application of taxi traces is discovering regions of different functions in city. Qi *et al.* [12, 14] establish and confirm the relationship between the pick-up/drop-off characteristics of taxi passengers and the social function of city regions with qualitative and quantitative analysis. Yuan *et al.* [19] propose a framework that discovers regions of different functions in a city using both human mobility among regions and points of interests (POIs) located in a region. They segment an urban road network into regions by an image-processing-based approach [20]. In their work, a region is represented by a distribution of functions, and a function is featured by a distribution of mobility patterns.

DATASET DESCRIPTION

We use trace dataset provided by the Traffic Bureau of Hangzhou City, which contains 7952 taxis and covers a period of 385 days. Taxis' state is sampled in a fixed time interval of 1 minutes and an extra sampling will be performed when the taximeter turn on or off. The position was obtained by GPS equipped in a taxi, so its precision was not affected by local tower density, which limited the spatial resolution of mobile-phone data. Each state consists of following fields:

- TAXI ID: the unique ID of sampled taxi;
- GPS POSITION: the longitude and latitude of that taxi at the sampling time;
- SPEED: the taxi speed at the sampling time, in kilometer per hour;
- ORIENTATION: the direction of that taxi at the sampling time, from 0° to 360° in clockwise with 0° indicates the north;
- METER STATE: indicates whether the taxi is heavy at the sampling time, 1 means the taxi is heavy(with passenger) and 0 means the taxi is empty(without passenger);
- TIME: the sampling time, with timestamp format 'YYYY-MM-DD HH:MM:SS'.

And a segment of state records in dataset is show in Table 1.

The state records of each taxi are extracted from dataset and sorted by time. Then, we define METER STATE turning from 0 to 1 as a pick-up event and turning from 1 to 0 as a drop-off event. A taxi trace is a series of state records begin with a pick-up event and last until encounter a drop-off event. The METER STATE may be incorrect because it is hard to avoid hardware faults thoroughly and taxi drivers may turn on the taximeter to avoid being interrupted when they have a rest, so a filtering process is necessary to remove these incorrect state records in order to recover taxi's actual traces from raw state records. Here we simply filter out taxi traces with distance less than 300m or travel time less than 2mins.

PATTERN ANALYSIS

To estimate the OD flows, we divide the urban area into locations with size 0.001 degree in longitude and 0.001 degree in latitude. Then we measure the number of taxis' traces that pick up a passenger in location L_i and drop off

him/her in the location L_j. The number of taxis' traces c_{ij} is a good approximation of OD flow from the location L_i to the location L_j. c_{ij} is rather uneven. The frequency f(k) of the *k*th most visited OD flow follows Zipf's law

$$f(k) \sim k^{-\zeta}$$

with $\zeta = 0.4337 \pm 0.0063$, indicating most of human movements in the city occur on some major OD flows. The number of OD flows with $c_{ij} \geq 1000$ is 633 and the number of locations related to those 633 OD flows is

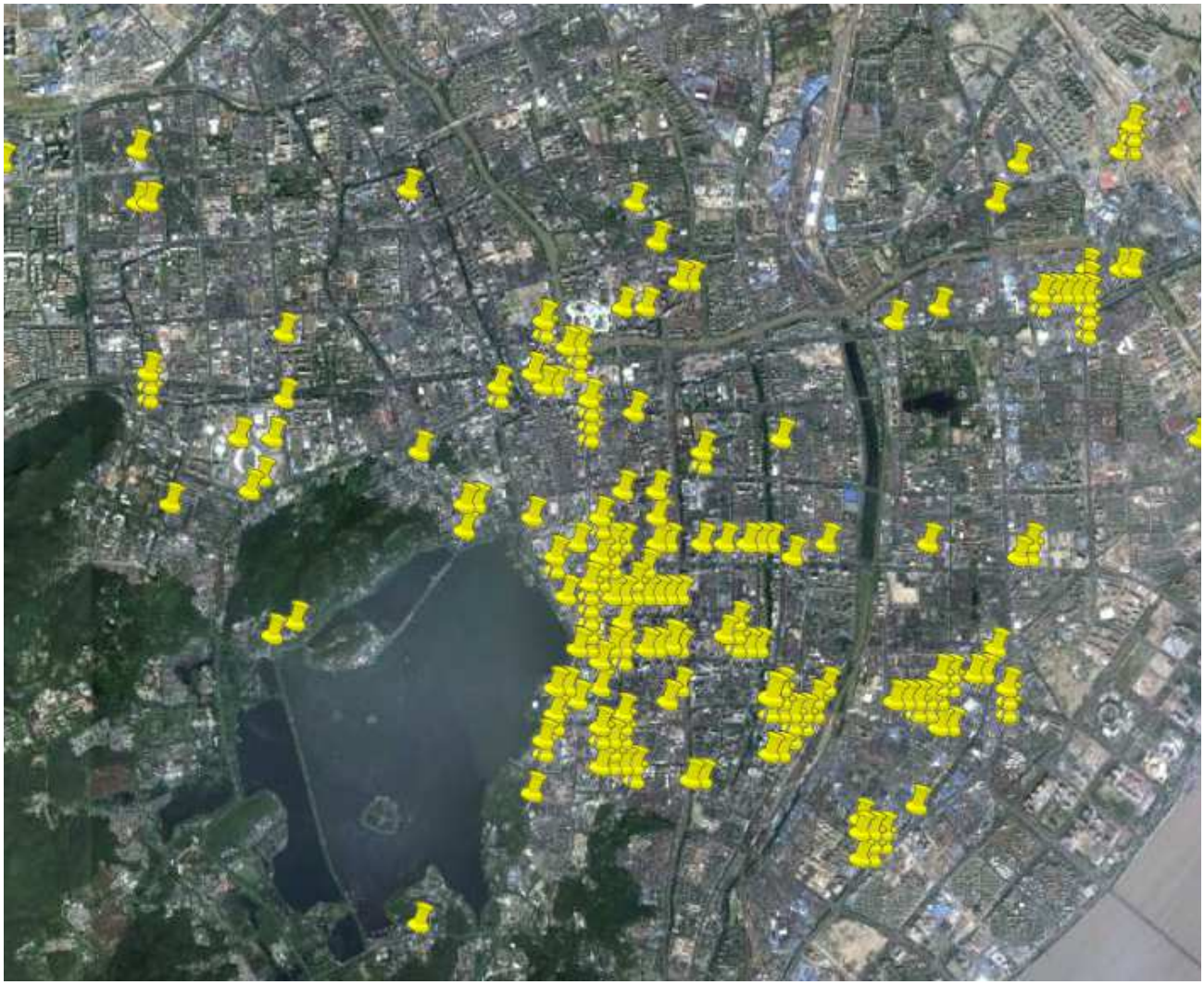


Figure 1. The map of Hangzhou city. Yellow Pins indicate origin/destination locations of OD flows with the number of taxis' traces $c_{ij} \geq 1000$. Note that locations are rather uneven distributed.

TAXI ID	LONGITUDE	LATITUDE	SPEED	ORIENTATION	METER STATE	TIME
1876	120.157295	30.241793	0.00	170.00	1	2009-4-1 00:00:04
14273	120.161180	30.272419	29.63	90.00	1	2009-4-1 00:00:04
2471	120.167820	30.284243	51.86	260.00	0	2009-4-1 00:00:04
14883	120.067444	30.090492	3.70	80.00	0	2009-4-1 00:00:04
18336	120.154850	30.290527	0.74	0.00	1	2009-4-1 00:00:07
10323	120.144110	30.327316	44.45	260.00	0	2009-4-1 00:00:07

Table 1. A segment of state records in dataset.

233(see Figure 1 and 2). Both number are very small compare with the total number of OD flows and locations but they indeed represent main human movements in the city. So we focus on analyzing those 633 OD flows and 233 locations.

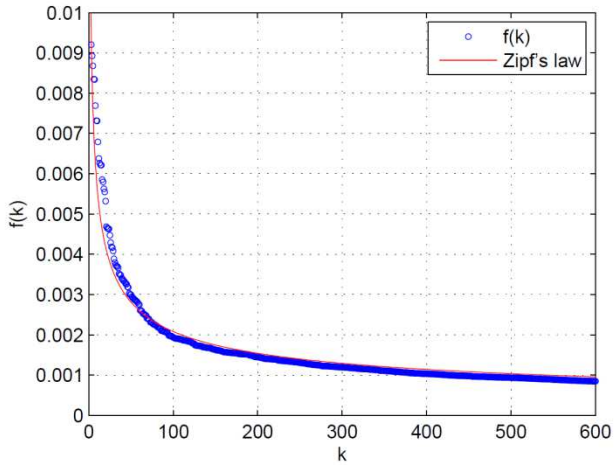


Figure 2. The frequency $f(k)$ of the kth most visited OD flow follows Zipf's law, $f(k) \sim k^{-0.4337 \pm 0.0063}$

To analyze the empirical observations, we measure the change of c_{ij} over time. This fine-grained result shows significant periodic pattern, reflecting the short-term dynamic of city. We define the power spectral density of c_{ij} as

$$S(d) = \frac{1}{N}\left|\sum_{n=1}^{N} c_{ij}(n) e^{-j(2\pi dn)}\right|^2$$

where $c_{ij}(n)$ is the the number of taxis' traces from location L_i to location L_j in time interval n. We find that two major components of its spectrum are 1cyc/day and 1cyc/week, this result is consistent with our daily experience.

After getting the period of c_{ij}, we can now depict each OD flow with a feature vector. Here we define three feature vectors:

- $V_d = \{c_{ij}^1, c_{ij}^2, \ldots, c_{ij}^{24}\}/c_{ij}$: Visit frequency over time of day. c_{ij}^k is the number of traces in the kth hour, c_{ij} is total number of traces.
- $V_w^1 = \{V_d^W, V_d^H\}$: Visit frequency over weekday and weekend. V_d^W is weekday's V_d and V_d^H is weekend's V_d.
- $V_w^2 = \{V_d^{Mo}, V_d^{Tu}, V_d^{We}, V_d^{Th}, V_d^{Fr}, V_d^{Sa}, V_d^{Su}\}$: Visit frequency over time of week. V_d^{Mo} is V_d on Monday, etc.

We find those feature vectors can more or less reflect the characteristic of OD flow. For example, For a OD flow from location L_{15}(a scenic spot) to location L_{32}(a luxury hotel), its V_d have peaks at 11:00AM and 15:00PM and its V_w^2 's weekend components are larger than weekday components(see Figure 3). So we can assume human activity mainly occur on day-time and weekend for this OD flow.

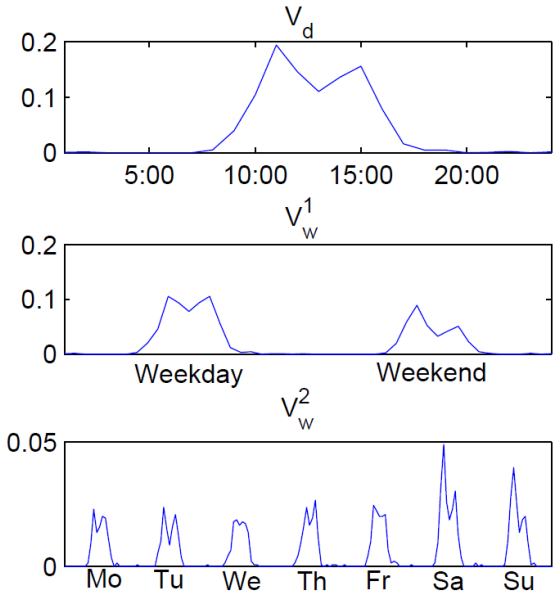


Figure 3. Feature Vectors of OD flow from location L_{15} (a scenic spot) to location L_{32} (a luxury hotel). V_d is the visit frequency over time of day; V_w^1 is the visit frequency over weekday and weekend; V_w^2 is the visit frequency over time of week. For this OD flow, human activity mainly occur on day-time and weekend.

As feature vectors reflect the characteristic the OD flow, we can do cluster to group OD flows with similar character. To compare the performance of those three feature vectors, we do K-means clustering based on them. We define the BSS/TSS factor as

$$\mathrm{BSS/TSS} = \frac{\sum_{S_i \neq S_j} \|V_i - V_j\|^2}{\sum_{i \neq j} \|V_i - V_j\|^2}$$

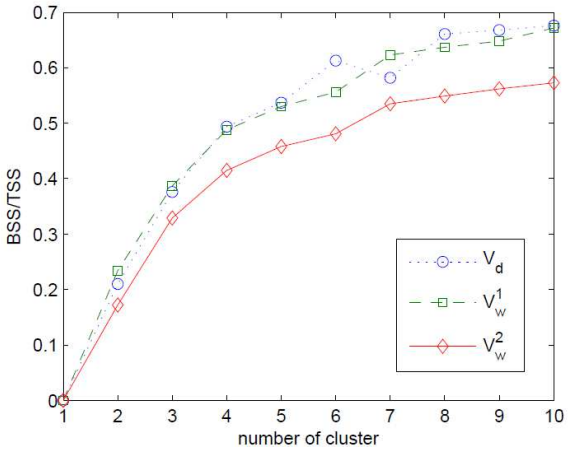


Figure 4. The factor BSS/TSS versus the number of clusters. Higher BSS/TSS indicates larger difference among clusters.

where V_i is a type of feature vector of location L_i and S_i is the cluster of location L_i. Notice that the factor for all three feature vectors increase quickly, indicating there are significant difference among OD flow clusters(see Figure 4). The cluster center can be viewed as principal pattern of OD flows belong to that cluster.

SEMANTICS MINING

To explore the semantics of OD flows, first we label location with semantics of main building in it such as Station or Hospital and investigate the cluster result. We find that most of the origin locations of OD flows in same cluster have same semantics, as well as destination locations. For all clusters, there are a few major semantics of locations: Station, Market, Hotel, Hospital, Mall, Dwelling and Bar. Station includes railway station, coach station, airport and large bus station; Market are places where merchants trade with each other while Mall are places people do shopping. Then we can define the semantics of OD flow by the semantics of its origin location and destination location such as Station to Station or Dwelling to Bar. The number of OD flows belong to each semantics type is also uneven (See Table 2).

Semantics Type	Number
Station to Station	184
Mall to Station	78
Hospital to Station	48
Market to Station	46
Mall to Mall	43
Station to Hospital	42
Other 42 Types	192
Total	633

Table 2. Number of each semantic kind of OD flows.

We compare 4 semantics of OD flows to show their relationship with patterns. The cluster center's V_d of OD flow from Dwelling to Bar has a peak at 21:00 and that of OD flow from Bar to Dwelling has a peak at 4:00, Those patterns are consistent with our daily experience that people go to entertainment place before mid-night and return after mid-night(see Figure 5). For semantics of OD flows with same origin Station, commute flow (Station to Market) aggregate in early morning while transfer flow (Station to Station) is flat distributed in day-time (see Figure 6). Based on the relationship mentioned above, we can now mine the semantics of OD flows by their feature vectors. We use a two-layer feed-forward Neural Network with sigmoid hidden and output neurons to classify the semantics of OD flows. The Neural Network is trained with scaled conjugate gradient back propagation.

To verify the performance of our method, we execute our method on taxi traces data of Hangzhou. The input data is randomly divided into three parts: 70% for training, 15% for validation and 15% for testing. The output is limited in six largest semantics types: Station to Station, Mall to Station, Hospital to Station, Market to Station, Mall to Mall and Station to Hospital. We run the classification process for 10 times and the average of their accurate rates is show in Table 3.

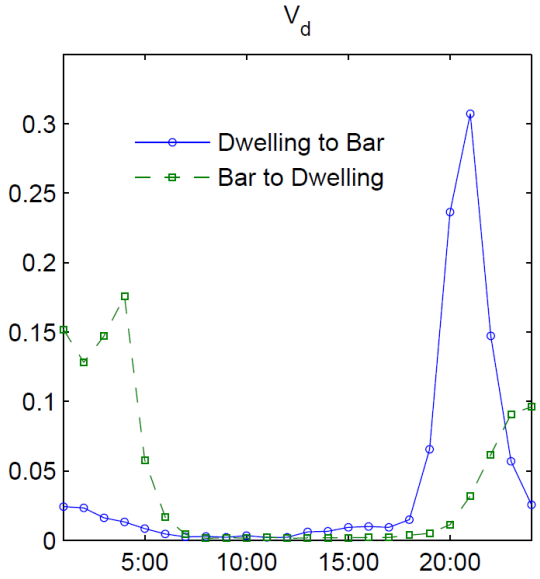


Figure 5. V_d of OD flow from Dwelling to Bar and OD flow from Bar to Dwelling. Note that people go to entertainment place before mid-night and return after mid-night.

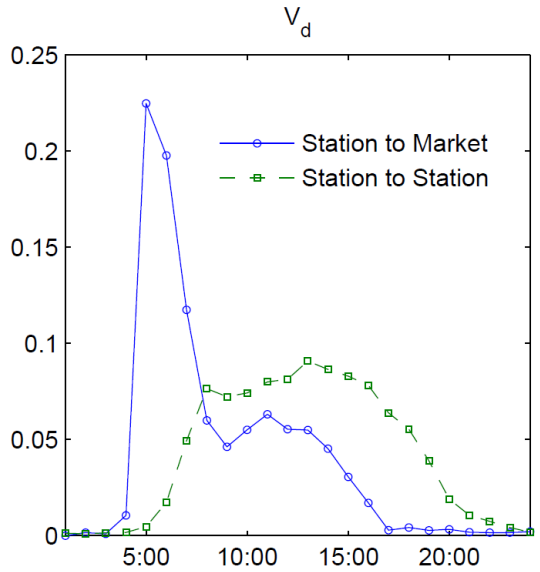


Figure 6. V_d of OD flow from Station to Market and OD flow from Station to Station. Note that commute flow(Station to Market) aggregate in early morning while transfer flow(Station to Station) is flat distributed in day-time.

Feature Vector Type	Average Accuracy
V_d	80.7%
V_w^1	83.4%
V_w^2	83.7%

Table 3. The average accuracy for three types of feature vectors.

Note that the average accurate rates of V_w^1 and V_w^2 are higher than that of V_d, indicating that the information of weekly repeated patterns can help us in mining semantics of OD flows. However, the average accurate rate of V_w^2 is nearly the same as that of V_w^1 while the length of V_w^2 is 3.5 times of that of V_w^1, so the weekday/weekend treatment is enough to represent the weekly repeated pattern.

CONCLUSIONS

In this paper, We estimate the origin-destination(OD) flows from taxis' traces and find that they have significant periodic patterns which closely related with their semantics. We mine the semantics of OD flows based on those patterns and the experiment result achieves a recognition accuracy of 83.7%. Our finding is useful to many LBSN applications and the approach we proposed offers a novel way to explore the human mobility and location characteristic.

Future work includes analyzing the semantics change of OD flow to discover urban events, comparing OD flow's pattern under different conditions such as urban-size or develop-level and detecting communities in city via the semantics of OD flows among them.

ACKNOWLEDGEMENTS

The authors would like to thank anonymous reviewers for the helpful comments. This work is partly supported by High-Tech Program of China (No.2011AA010104) and Qianjiang Talent Program of Zhejiang (2011R10078). The corresponding author is Dr. Gang Pan.

REFERENCES

1. F. Calabrese, G. D. Lorenzo, L. Liu, and C. Ratti. Estimating origin-destination flows using mobile phone location data. *IEEE Pervasive Computing*, 10(4):36–44, 2011.
2. V. Colizza, A. Barrat, M. Barthelemy, A. Valleron, and A. Vespignani. Modeling the worldwide spread of pandemic influenza: baseline case and containment interventions. *PLoS Medicine*, 4(1):95-110, 2007.
3. J. Cranshaw, E. Toch, J. Hong, A. Kittur, and N. Sadeh. Bridging the gap between physical location and online social networks. In *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, pages 119–128. ACM, 2010.
4. N. Eagle, A. Pentland, and D. Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274–15278, 2009.
5. M. Gonzalez, C. Hidalgo, and A. Barab´asi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.
6. M. Horner and M. O'Kelly. Embedding economies of scale concepts for hub network design. *Journal of Transport Geography*, 9(4):255–265, 2001.
7. L. Hufnagel, D. Brockmann, and T. Geisel. Forecast and control of epidemics in a globalized world. *Proceedings of the National Academy of Sciences*, 101(42):15124-15129, 2004.
8. B. Jiang, J. Yin, and S. Zhao. Characterizing the human mobility pattern in a large street network. *Phys. Rev. E*, 80(2):021136, Aug 2009.
9. R. Kitamura, C. Chen, R. Pendyala, and R. Narayanan. Micro-simulation of daily activity-travel patterns for travel demand forecasting. *Transportation*, 27(1):25–51, 2000.
10. M. Li, J. Dai, S. Sahu, and M. Naphade. Trip analyzer through smartphone apps. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 537–540. ACM, 2011.
11. X. Li, G. Pan, Z. Wu, G. Qi, S. Li, D. Zhang, W. Zhang, and Z. Wang. Prediction of urban human mobility using large-scale taxi traces and its applications. *Frontiers of Computer Science*, 6(1):111–121, 2012.
12. G. Pan, G. Qi, Z. Wu, D. Zhang, and S. Li. Land-use classification using taxi GPS traces. *IEEE Transactions on Intelligent Transportation Systems,* , 2012. DOI 10.1109/TITS.2012.2209201
13. G. Pan, Y. Xu, Z. Wu, S. Li, L. Yang, M. Lin, and Z. Liu. Taskshadow: Toward seamless task migration across smart environments. *IEEE Intelligent Systems,* 26(3):50–57, May-June 2011.
14. G. Qi, X. Li, S. Li, G. Pan, Z. Wang, and D. Zhang. Measuring social functions of city regions from large-scale taxi behaviors. In *2011 IEEE International Conference on Pervasive Computing and Communications, Work-in-progress*, pages 384–388. 2011.
15. J. Sun, Z. Wu, and G. Pan. Context-aware smart car: from model to prototype. *Journal of Zhejiang University - Science A*, 10(7):1049–1059, 2009.
16. M. Veloso, S. Phithakkitnukoon, and C. Bento. Urban mobility study using taxi traces. In *Proceedings of the 2011 International Workshop on Trajectory Data Mining and Analysis*, pages 23–30. ACM, 2011.
17. M. Veloso, S. Phithakkitnukoon, C. Bento, N. Fonseca, and P. Olivier. Exploratory study of urban flow using taxi traces. In *The First Workshop on Pervasive Urban*

Applications (PURBA'11), in conjunction with PERVASIVE'11. 2011.

18. Z. Wu, Q. Wu, H. Cheng, G. Pan, M. Zhao, and J. Sun. Scudware: A semantic and adaptive middleware platform for smart vehicle space. *IEEE Transactions on Intelligent Transportation Systems*, 8(1):121–132, March 2007.
19. J. Yuan, Y. Zheng, and X. Xie. Discovering regions of different functions in a city using human mobility and POIs. In *Proceedings of the 18th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2012.
20. N. Yuan, Y. Zheng, and X. Xie. Segmentation of urban areas using road networks. *MSR-TR-2012-65*, 2012.
21. D. Zhang, N. Li, Z. Zhou, C. Chen, L. Sun, and S. Li. ibat: detecting anomalous taxi trajectories from gps traces. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, pages 99–108. ACM, 2011.
22. Y. Zheng. Location-based social networks: Users. *Computing with Spatial Trajectories*, Y. Zheng and X. Zhou, Eds. Springer, 2011.
23. Y. Zheng, Y. Liu, J. Yuan, and X. Xie. Urban computing with taxicabs. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, pages 89–98. ACM, 2011

Towards Reliable Spatial Information in LBSNs

Ke Zhang
University of Pittsburgh
kez11@pitt.edu

Wei Jeng
University of Pittsburgh
wej9@pitt.edu

Francis Fofie
University of Pittsburgh
fof1@pitt.edu

Konstantinos Pelechrinis
University of Pittsburgh
kpele@pitt.edu

Prashant Krishnamurthy
University of Pittsburgh
prashant@sis.pitt.edu

ABSTRACT

The proliferation of Location-based Social Networks (LBSNs) has been rapid during the last year due to the number of novel services they can support. The main interaction between users in an LBSN is location sharing, which builds the spatial component of the system. The majority of the LBSNs make use of the notion of check-in, to enable users to volunteeringly share their whereabouts with their peers and the system. The flow of this spatial information is unidirectional and originates from the users' side. Given that currently there is no infrastructure in place for detecting fake check-ins, the quality of the spatial information plane of an LBSN is solely based on the honesty of the users. In this paper, we seek to raise the awareness of the community for this problem, by identifying and discussing the effects of the presence of fake location information. We further present a preliminary design of a fake check-in detection scheme, based on location-proofs. Our initial simulation results show that if we do not consider the infrastructural constraints, location-proofs can form a viable technical solution.

Author Keywords

Location-based social networks, Location proofs, Security.

ACM Classification Keywords

H.0 Information Systems: General.; K.6.5 Management of Computing and Information Systems: Security and Protection.

General Terms

Design, Reliability, Security

INTRODUCTION

Location-based Social Networks (LBSN) have attracted a lot of attention during the last years. While they exist since the early 2000s (e.g., Dodgeball was founded in 2003), it is only recently that LBSNs have taken off, mainly due to the advancements in mobile handheld devices. The latter allow for a fairly accurate positioning, thus, forming an ideal platform for the realization of advanced location sharing applications.

An LBSN has two distinct components: a social network and a location log for each user. The social part of the system resembles any other existing online social network, where friendships are declared and people can interact with their friends. What differentiates LBSNs for any other digital social network is the type of interaction that are feasible among the users. The main feature of this interaction is location sharing. Users volunteeringly share their location with their friends (or even with everyone in the system depending on the privacy settings). This location information can be either in the form of a trajectory continually tracked by the provider (e.g., systems such as Loopt) or in the form of volunteering sharings of the actual place/venue the user is in through a *check-in* (e.g., systems such as Foursquare). Some systems might also offer both alternatives (e.g., Google Latitude). Clearly, the second approach, where locations are tagged with semantic information (e.g., "I am in the Starbucks") as compared with a geographic trajectory (e.g., specific latitude/longitude), offers richer data that can enable novel services. Hence, this is the model that most popular LBSN utilize. A nice overview of location-based social networks and systems in general can be found in [18] for the interested reader.

In both of the aforementioned models, the flow of the spatial information is unidirectional. In particular, the user provides his location to the system, and as a consequence to the rest of the network. In the above process there is no proof of correctness for any information provided. However, as He *et al.* have shown [8], it is very easy to interfere with the positioning system of a mobile device and alter it in order to report fake coordinates. Moreover, in checkin-based LBSNs the users do not even have to alter the GPS' API to forge their whereabouts; they can simply bypass the automatic localization module[1] and check-in at a different venue than the one they actually are. While some LBSNs offer basic schemes to identify fake check-ins (e.g., the cheater code of foursquare [1]), their scope is limited (e.g., they do not perform well when the dishonest user is located fairly close to the venue he claims to be in).

[1]Since the accuracy and/or the availability of the GPS can be low, especially in urban areas, LBSNs allow users to manually enter the required information if needed.

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.

While in this paper we are not interested into identifying the reasons behind location cheating, the incentives for adopting similar behaviors vary and can be present in a high degree. LBSNs have a strong gaming component and many users are interested in these mobile games [11]. Hence, they might be inclined to cheat simply to gain more virtual rewards (e.g., more points, *badges*/*pins*, etc.). Moreover, while LBSNs were designed mainly with the objectives of connecting people in space, helping them to meet new people in their vicinity, keep track of their own friends and explore new areas, business-related features and interactions related to monetary gains have recently taken-off. For instance, the owner of a venue can offer deals to users that check-in his venue [4]. The majority of these offers (more than 90%) require multiple check-ins [8]. Hence, a user, say Jack, can create a number of fake check-ins for unlocking this offer easier, leading to a monetary loss for the venue owner. In another context, Jack might also be tempted to share a fake location with the system in order to provide some sort of "alibi" or mislead other people with regards to his location.

The contribution of our work is two-fold and can be summarized in the following: **(i)** Raise the awareness of the community for the importance of identifying fake location sharings in LBSNs. We emphasize on the importance of solving this problem by discussing the effects of counterfeit spatial information. **(ii)** Design of a preliminary system, based on the primitives of location proofs, for the detection of fake check-ins. To the best of our knowledge this is the first scheme to tackle this problem.

The rest of the paper is organized as follows. Section discusses studies related with our work. Section analyzes possible effects of the presence of forged check-ins, while Section presents the design and evaluation of our preliminary detection scheme. Finally, Section concludes our work.

RELATED STUDIES

With the increased importance of spatial information for various applications, location-proofs have gained attention in the research community during the last years. Denning and MacDoran [6] describe a location-based authentication system where the position at any time is uniquely identified by a location signature. The signature is created by a location signature sensor (LSS) and it is time varying, hence, making it difficult to be forged. However, this system relies on a dedicated hardware and requires auxiliary equipments to strengthen the weak GPS signal in indoor environment. Saroiu and Wolman [14] design a scheme where location proofs are handed out by WiFi access points (APs). Each mobile device signs the APs' beacons and send them back to APs. The latter upon reception of the signed beacon creates a location signature for the mobile user. Zhang *et al.* [17] also utilize WiFi infrastructure and design a power modulated challenge-response location verification system. This mechanism utilizes RF signal strength from multiple APs to verify whether the claimed location is within the overlapping range of neighbouring APs. Furthermore, Kjærgaard and Wirz [9] present a clustering approach to detect indoor flocks of mobile users (i.e., spatio-temporal clusters).

In the context of LBSN, as aforementioned He *et al.* [8] have identified the problem of fake check-ins, without providing any solution to it. Foursquare has developed the *cheater code* [1] in an effort to minimize fake spatial information. The cheater code imposes additional rules on users' check-in frequency and speed. However, this mitigates potential fake check-ins to some extent only, since cheaters can easily bypass this detection [8]. Furthermore, location learning schemes can potentially used to enhance the detection of fake check-ins. For instance, Lian and Xie [10] propose a scheme to identify the location of a specific check-in based on primitives of location search. The location identified from the system can then be compared with the one claimed from the user. However, this scheme will be able to identify a number only of non-sophisticated fake check-ins (e.g., users that check-in to a remote locale without altering their GPS coordinates).

Our preliminary system design, is based on the primitives of location-proofs and can be complementary to efforts such as the cheater code. The key point is that a cheating user, say Jack, while being able to fake his GPS coordinates, he cannot do the same with the wireless channel's propagation characteristics. In brief, we utilize the notion of location signature using WiFi infrastructure enhancing it with the notion of flocks for identifying users that are not at the location (at the time) they claim to be at.

THE EFFECTS OF FAKE CHECK-INS

Traditionally, social information systems and information quality have followed disjoint paths. However, the presence of low quality of information (QoI) results in a decreased value for the specific platform. For instance, a very representative type of social information systems, vulnerable to low QoI is the Q&A social networks [16] [13]. In these systems, people post questions that can be answered from their peers, and are focused on providing an efficient platform for enabling the crowdsourcing nature of the underlying system. However, no care is taken for the actual quality of the answers provided. In the case of Q&A networks, feedback from users can be roughly used as a metric of the quality of the answer provided.

While similar issues exist related to the location sharing in LBSNs, there are no systems to date that are able to filter fake check-ins to a great extent. However, the existence of forged spatial information in the network can have significant effects in a wide spectrum of the underlying functionalities. In this section we will discuss two representative examples; one related to the effects on participating businesses and one related to the services possibly offered by the LBSN provider.

Monetary Losses: LBSNs have recently evolved into an *inexpensive* marketing channel for local businesses. Users can obtain special offers by checking-in at participating venues. This gives the latter an opportunity to be advertised, in an inexpensive way, only to people that actually have the potential to visit them. A traditional advertisement, which targets the majority of the population is expensive because of the vol-

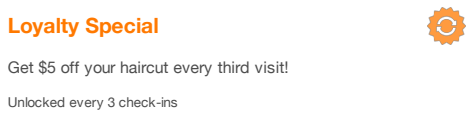


Figure 1. An example of a special offer requiring 3 check-ins.

ume of its target mass. However, only a percentage of the people exposed to it can actually benefit from the advertised good/service (e.g., people that are spatially located nearby).

Special deals lead to a temporary loss for the locale offering it. This is especially true for one-time deals, such as the ones offered in Groupon [5]. The rational behind these offers is that people visiting the venue will come back and hence, this will make up for the temporary loss. Hence, participating venues in LBSN offers might require more than one visit in an attempt to minimize the associated loss. For instance, Figure 1 provides an example of a special offer, which requires 3 check-ins. If the cost of the offer is c (in our example $c = \$5$), the locale's gain would be reduced by c for every visit if the deal was offered to every check-in. By requiring three check-ins the gain is only reduced by $\frac{c}{3}$ for every visit.

Jack who wants to unlock this offer but does not want to have to go three times to the venue and spend on average $s - \frac{c}{3}$ per visit, where s is the average expenditure of a client in the locale, can game the system and create two initial fake check-ins. This will enable him be present in the locale only once and unlock the deal. His expenditure will be only $s - c$, and the venue owner will have a reduced gain (the cost of the offer will again be c per visit). Assuming that $s = \$20$ in our example, the locale's gain is reduced by \$5/client instead of the $\frac{5}{3} = \$1.6$/client that was the target, while the cost per visit for Jack is reduced to \$15 instead of $20 - \frac{5}{3} = \$18.4$. Hence, it is evident that there can be monetary losses for businesses that want to use LBSNs as an advertisement channel. A detection system should be developed in order to filter forged check-ins and establish a secure way for venues offering deals. If the latter is not in place, business owners will have a reduced incentive to participate in similar systems.

Degraded Services: Recently, Foursquare - the largest LBSN to date - launched a novel recommendation engine, which considers check-ins from all users in order to provide recommendations [2] [3]. This engine takes into account user's check-ins, friend's check-ins, venue's check-ins and many other factors in order to provide suggestions. It should be evident that *noisy* data will not yield high quality service. Therefore, not only should LBSNs filter fake check-ins that can harm businesses (as aforementioned), but also identify any kind of fake check-ins (e.g., from gamers that simply want to gain as many virtual rewards as possible by checking-in to venues they have never been). Unless only "True" data are used from the LBSN provider to provide services, the latter will be degraded and of low quality.

Hence, it is evident that fake check-in detection is crucial to the long-run success of the LBSN paradigm. In the following section, we present our initial efforts on this problem.

FAKE CHECK-IN DETECTION

Cheating Model: In our work we consider two types of fake check-ins; (i) users can modify their GPS API and check-in a venue that is located far away and (ii) users that check-in to a locale that is nearby even if they are not physically present in it. Note here, that approaches such as the cheater code would not be able to detect any of them. However, the latter would be able to detect users that do not alter their GPS API and check-in to a far away venue, and thus, we do not consider it in our work. A realistic assumption we make for this study is that the numbers of fake check-ins are less than true check-ins (assumption 1). In addition, true check-ins are spatially contained within the premises of a venue, while fake check-ins are distributed over a larger area outside the latter (assumption 2).

Detection Algorithm: To defend against fake check-ins, every mobile user needs to provide location evidence to the LBSN provider along with his check-in information. For issuing location evidence, the mobile device collects beacon frames sent by nearby WiFi APs and measures the received signal strength (RSS). This provides a vector $RSS = [rss_1\ rss_2\ ...\ rss_n]$, which combined with a vector containing the unique MAC addresses of each AP ($MAC = [mac_1\ mac_2\ ...\ mac_n]$) forms the location proof which is forwarded to the LBSN provider with the check-in.

For location verification of a check-in of user u at locale l, the LBSN provider utilizes the recent k proofs of users claiming presence in l. Then spatial clustering on the RSS space is performed using the density clustering algorithm DBSCAN [7] as described in what follows. Having a set of points (check-ins in our case) DBSCAN first calculates the neighborhood $N(p)$ for each point p. The latter consists of all points within distance ϵ from p (the distance is calculated over the RSS vectors). The algorithm proceeds by examining whether it can merge the neighborhood to an existing cluster. The latter is possible if the neighborhood shares at least one common point to a cluster. Otherwise, if $|N(p)| \geq$ `MinPts` a new cluster is created. However, if $|N(p)| <$ `MinPts`, p with its neighborhood are considered "noise". Figure 2 depicts the high level approach of

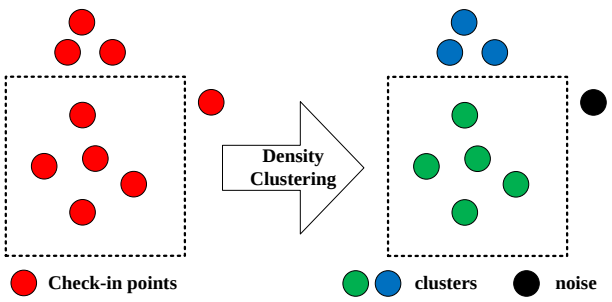


Figure 2. Pictorial representation of spatial clustering (`MinPits`=3).

DBSCAN. Clearly, ϵ and `MinPts` are two parameters that dictate the clusters and "noise" points identified.

In our context, considering the points defined by the RSS vectors of the check-ins claimed in a specific venue, we expect the points originating within the venue to be closer, thus belonging to the same cluster, as compared to those created outside the locale, due to the different wireless signal propagation path. In other words, real and fake check-ins will not be clustered together. Furthermore, we expect the fake check-ins to form clusters of lower cardinality (assumption 1) and/or be "noise" points (assumption 2).

The LBSN provider keeps track of the check-ins to a specific venue l and utilizes the latest k of them. Initially, when there are less than k prior check-ins none of them is classified. Once k of them are obtained spatial clustering is applied. Based on our discussion above, the cluster that includes the most points is flagged as "True" check-ins. The rest points (i.e., check-ins) are classified as "Fake".

Let us now consider Jack claiming to be in locale l. Using a sliding window approach, the k latest classified check-ins (say set S) together with that of Jack form the input to the density clustering. There are two possibilities for Jack's check-in; either classified as "noise" or belong to a cluster. In the former case the check-in is regarded as "Fake" (assumption 2). In the latter case, if the corresponding point belongs to the cluster with the largest cardinality then the check-in is flagged as "True", otherwise as "Fake" (assumption 1).

Note here that, while assumptions 1 and 2 might hold in the long run and over the total check-in set, it might be the case that they do not hold true for a subset of them (i.e., a specific set S). In order to avoid cascades of misclassification, every time a new check-in at l *arrives* we perform a reclustering. However, note that we only decide for the latest check-in in time; there is currently no feedback control for reinforcing previous check-ins classification.

One could use all the check-in history of a locale in order to apply the density clustering. However, evidence and RSS vectors might become stale due to the temporal variations of the wireless channel, as well as changes in the WiFi deployments. Exactly these are the factors that can provide robustness of our approach to replay attacks, where users record the location proofs at time t_1 and provide them with a fake check-in at time t_2.

Simulation and Evaluation: To evaluate our design, we simulate the check-in process over a virtual grid of locales. Venues are grouped into blocks of 6 and arranged in a 2D plane separated by streets. Venues within a block are tangent and separated by walls. 90% of venues are equipped with a WiFi access point. Our simulations include 24 venues (i.e., 4 blocks) and 20 users.

LBSN users follow the RANK model [12] to decide the next destination to check in. According to this model, the probability a user check in venue $v \in U$ from original venue $u \in U$ is defined as:

$$P_{uv} = \frac{rank_u(v)^{-\alpha}}{\sum_{u \in U} rank_u(v)^{-\alpha}} \quad (1)$$

where,

$$rank_u(v) = |\{w \in U : d(u,w) < d(u,v)\}| \quad (2)$$

$d(u,w)$ is the distance between locales u and w. We have also used $\alpha = 0.84$ [12]. For a user who is truthfully checking-in to locale l, his actually position within l is randomly chosen. A user who performs a fake check-in, will be positioned randomly outside the venue, where the probability density of the distance follows an exponential distribution.

We also use a wireless signal propagation model for the RSS values recorded from the users. In particular, we use the Attenuation Factor Model [15]:

$$RSS = P(d_0) + 10n \log\left(\frac{d}{d_0}\right) + n_w \cdot W + v, \quad (3)$$

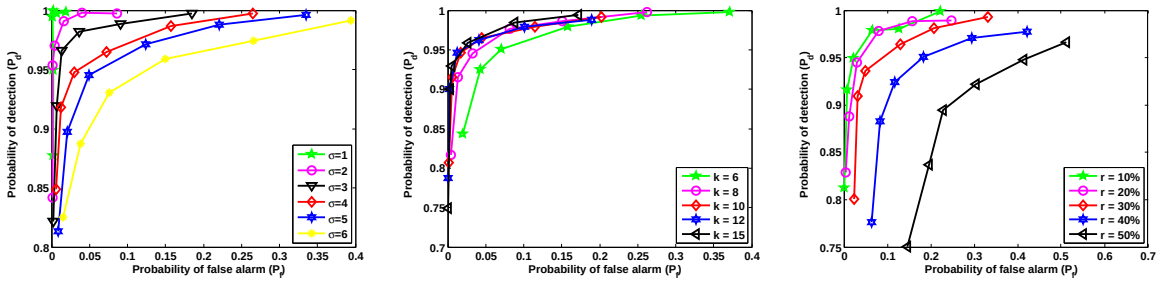
where, $P(d_0)$ (dBm) is the signal strength at distance d_0, n is the path loss exponent, W is an wall attenuation factor and n_w is the number of obstacles along the direct signal propagation path between transmitter and receiver, and v is Gaussian with $N \sim (u, \sigma^2)$. In our simulations, we set $d_0 = 1m$, $P(d_0) = -30dBm$, $n = -2.4$, $W = -15dBm$ and $u = 0$. σ is varied as described below.

We evaluate the performance of our fake check-in detection scheme with regards to the variations of the wireless channel as captured by the deviation σ, the percentage r of actual fake check-ins present in the system and different window sizes k. We set `MinPts`=3. ϵ determines the tradeoff between the probability of detection (a "Fake" check-in correctly classified) and false alarm (a "True" check-in classified as "Fake") and is used as the parameter for obtaining our results.

Figure 3(a) depicts the detection and false alarm probabilities for different σ ($k = 8$, $r = 20\%$). Each point in the curve is obtained for a different ϵ. As we can see the performance is much better when the wireless channel is stable. However, in a stable environment replay attacks can be more successfully since location proofs do not change over time. Nevertheless, even in a highly variable environment ($\sigma = 6$), the algorithms performs efficiently and is also more robust to replay attacks.

Figure 3(b) presents the performance of our scheme for different k ($\sigma = 4$ and $r = 20\%$). Again each point on the curves is obtained for a different threshold ϵ. As one might have expected the more points we input to the clustering algorithm, the more accurate detection it can perform.

Finally, Figure 3(c) presents the results for a varying percentage of fake check-ins r ($\sigma = 4$, $k = 8$). We can see that



(a) Variations in the wireless environment can degrade performance.

(b) The longer the history, the better the performance of our scheme.

(c) Detection works best when the number of fake check-ins is small.

Figure 3. ROC curves for our detection scheme.

the system performs better when r is small. When the latter increases, density clustering performance can be degraded, especially during the initialization phase, where the cluster including the most points is considered as the "True".

In all of the above results as we increase ϵ we move from the top right of the curve to the bottom left. Larger ϵ translates to higher probability that a point can be connected to a cluster. This, decreases the detection probability since it is easier for a fake check-in to fall into a "True" cluster, but also decreases the probability of false alarm.

Future directions: In the above we have presented our initial approach towards identifying fake check-ins. We seek to further extend our approach by examining other clustering algorithms and implementing a prototype system that would enable evaluation in a real environment. In particular, we want to examine: (i) possibilities for feedback control as aforementioned and (ii) the robustness of our approach to replay attacks. Finally, while the assumption of fake users being less than the real users is realistic, we opt to investigate different approaches whose performance is not affected by the proportion of the fake users.

CONCLUSIONS

In this work we have studied the problem of fake check-in information in LBSNs. We have argued for the importance and cruciality of this issue by analyzing possible effects from counterfeit spatial information. We have further designed and evaluate via simulations a detection system combining clustering algorithms and primitives of location proofs. We believe that our study will raise the awareness of the LBSN community and stimulate further research on the topic.

REFERENCES

1. Foursquare's cheater code: http://blog.foursquare.com/2010/04/07/503822143/.
2. Foursquare's recommendation engine: http://engineering.foursquare.com/2011/08/03/foursquares-data-and-the-explore-recommendation-engine/.
3. Foursquare's redesign: http://techcrunch.com/2012/06/03/foursquare-redesign-coming/.
4. Foursquare's special offers: https://foursquare.com/business/merchants/specials.
5. J. Byers, M. Mitzenmacher, and G. Zervas. Daily deals: prediction, social diffusion, and reputational ramifications. In *WSDM*, 2012.
6. D. E. Denning and P. F. Macdoran. Location-Based Authentication : Grounding Cyberspace for Better Security. *Computer Fraud and Security*, (February):12–16, 1996.
7. M. Ester, X. Xu, H.-P. Kriegel, and J. Sander. *Density-based algorithm for discovering clusters in large spatial databases with noise*, pages 226–231. AAAI, 1996.
8. W. He, X. Liu, and M. Ren. Location cheating: A security challenge to location-based social network services. In *IEEE ICDCS*, 2011.
9. M. Kjaergaard, M. Wirz, D. Roggen, and G. Troster. Mobile sensing of pedestrian flocks in indoor environments using wifi signals. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 95 –102, march 2012.
10. D. Lian and X. Xie. Learning location naming from user check-in histories. In *ACM SIGSPATIAL GIS*, 2011.
11. J. Lindqvist, J. Cranshaw, J. Wiese, J. Hong, and J. Zimmerman. I'm the mayor of my house: Examining why people use foursquare - a social-driven location sharing application. In *ACM CHI*, 2011.
12. A. Noulas, S. Scellato, R. Lambiotte, M. Pontil, and C. Mascolo. A tale of many cities: universal patters in human urban mobility. In *PLoS ONE 7(5): e37027. doi:10.1371/journal.pone.0037027*, 2012.
13. K. Pelechrinis, V. Zadorozhny, and V. Oleshchuk. Collaborative assessment of information provider's reliability and expertise using subjective logic. In *CollaborateCom*, 2011.
14. S. Saroiu and A. Wolman. Enabling new mobile applications with location proofs. *Proceedings of the 10th workshop on Mobile Computing Systems and Applications - HotMobile '09*, pages 1–6, 2009.

15. S. Seidel and T. Rappaport. 914 mhz path loss prediction models for indoor wireless communications in multifloored buildings. *Antennas and Propagation, IEEE Transactions on*, 40(2):207–217, 1992.

16. J. Zhang, M. A. Ackerman, and L. Adamic. Expertise networks in online communities: Structure and algorithms. In *WWW*, 2007.

17. Y. Zhang, Z. Li, and W. Trappe. Power-modulated challenge-response schemes for verifying location claims. *IEEE GLOBECOM*, pages 39–43, 2007.

18. Y. Zheng and X. Zhou. *Computing with Spatial Trajectories*. Springer-Verlag New York, LLC, 2011.

Detection, Classification and Visualization of Place-triggerd Geotagged Tweets

Shinya Hiruta[1], **Takuro Yonezawa**[1], **Marko Jurmu**[1,2], **Hideyuki Tokuda**[1]

[1]Keio University
5322 Endo, Fujisawa, Kanagawa, Japan
{hiru, takuro, hxt}@ht.sfc.keio.ac.jp

[2]University of Oulu
Erkki Koiso-Kanttilan katu 3, 90570 Oulu, Finland
marko.jurmu@ee.oulu.fi

ABSTRACT

This paper proposes and evaluates a method to detect and classify tweets that are triggered by places where users locate. Recently, many related works address to detect real world events from social media such as Twitter. However, geotagged tweets often contain noise, which means tweets which are not content-wise related to users' location. This noise is problem for detecting real world events. To address and solve the problem, we define the *Place-Triggered Geotagged Tweet*, meaning tweets which have both geotag and content-based relation to users' location. We designed and implemented a keyword-based matching technique to detect and classify place-triggered geotagged tweets. We evaluated the performance of our method against a ground truth provided by 18 human classifiers, and achieved 82% accuracy. Additionally, we also present two example applications for visualizing place-triggered geotagged tweets.

General Terms

Design, Experimentation

Author Keywords

Microblogs, Location-based Services, Place-triggered Geotagged Tweets

ACM Classification Keywords

H.3.5 Web-based services.

INTRODUCTION

Real world event can be formally structured as a collection of descriptive attributes. Frequently, these attributes are dynamic in nature which means that static *a priori* descriptions cannot be used. Furthermore, a real world event can manifest itself without any a priori static description, in which case dynamic information is the only source for communicating information regarding the event. Example of former is a baseball game that gets postponed because of rain, and an example of the latter is a traffic accident occurring on a motorway and causing significant traffic congestion.

From this framing of real world events, we can conclude that systems which *extract, classify and provide real-time dynamic attributes of the event* are needed. In this paper, we focus on location as a key attribute of both aforementioned event types. This is because location is the most common denominator for a wide variety of events, and in many cases the single most important one. Especially in events describing accident or catastrophe information, location is the first attribute to be resolved.

In designing aforementioned systems, we need to consider the possible data sources and their suitability for this purpose, the actual extraction and classification methods, and finally the APIs that the system can offer for third party applications. Since we classify two types of real world events, those that have a static a priori description and those that don't, we can envision the API providing a subscription-based service for receiving dynamic attributes of an event, given a static unique identifier of the event. More challenging is the API for dynamically occurring events, which can rely on more advanced subscription mechanisms.

Considering the data sources, social networking services are suitable for extraction of dynamic real world data. In this paper, we especially focus on Twitter, due to its public and agile nature as a communication medium. Regarding Twitter, there are several strategies to employ in extraction and analysis. One strategy is to use metadata-only, for example the geotags encoded into tweets. This is however insufficient, since there are a lot of noisy tweets whose contents are not related to the location. Another strategy is to combine the metadata extraction with content-based analysis, meaning first to filter the tweets based on metadata, and subsequently analyze them based on their content. This allows us to identify precise location of tweets using only geotagged data, and to detect meaningful tweets which have a content-based relation to their location.

We define *Place-triggered Geotagged Tweets* as those tweets that have both the geotag metadata as well as content relevant to the associated location. In this paper, we present our first approach towards extraction, analysis and provision of dynamic event attributes by designing, implementing and evaluating a keyword-based classifier for place-triggered

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.

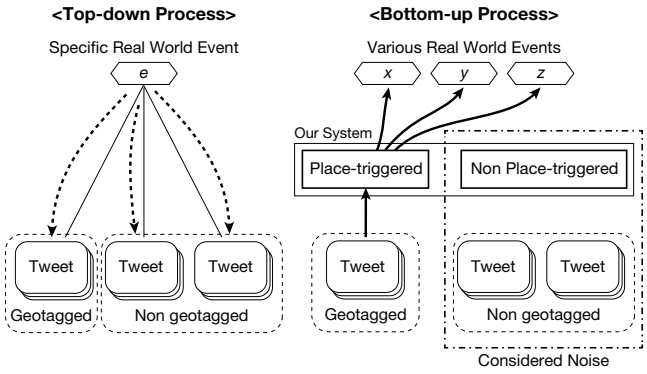


Figure 1. Comparison of approaches to detect events from tweets

geotagged tweets. First, in order to classify the type of place-triggered geotagged tweets, we surveyed geotagged tweets which were collected from Twitter. As a result, we classified the place-triggered geotagged tweets into 5 types: *report of whereabouts*, *food*, *weather*, *back at home* and *earthquake*. Based on these results, we designed filters to classify geotagged tweets into these 5 types by naive keyword matching method. We asked third parties of 18 people to create ground truth. Based on it, we evaluated relevance of our classification and determination accuracy of the filter.

The key contributions of this paper are:

- We survey and classify current usage of geotagged tweets.
- We present the design and implementation of a prototype system which detects and classifies *Place-triggered Geotagged Tweets*.
- We present results from an evaluation study that compares ground truth created by third parties of 18 people with our outcome filtered by the system.

DETECTING EVENTS FROM TWEETS

We define two ways to detect events from social media. One is the top-down process, which first specifies events and then filters the tweets for detection of these events. The other is the bottom-up process, where tweets converge into certain topics of interest, possibly by using certain ground truth as evidence. We take the latter approach, and introduce a new concept of *Place-triggered geotagged tweet*.

Methodology

Top-down Process

We define Top-down Process as a method which is intended for detecting specific events. The objective is to first detect certain events, and then construct a method for detection of these events. The left part of Figure 1 describes an approach of Top-down process. First, a specific event *e* is set, and then tweets are crawled for event detection.

This approach is suitable for events which have obvious characteristics that help to recognize the fact that the event has occurred. For example, forest fire events [3] and live news events [5] are being detected through this process.

Bottom-up Process

In contrast to the needs-oriented process, we define the bottom-up process that classifies the tweets as new point of view. There are two aspects revealing whether a tweet is related to real world events or not. One is the "geotag relation" which means whether a geotag is appended to or not. The other is "content-based relation", meaning whether contents of the tweet refer to the current location or not. These are described in Figure 1.

We define the geotagged tweets whose contents are motivated by events or situations of current locations as "Place-triggered Geotagged Tweets". We consider this bottom-up augmentation of Twitter data as important enabler to detect real world events by mining social media.

Objectives

Our goal is to investigate whether a keyword based classifier, aided by ground truth notion, can effectively classify and augment twitter data for the purpose of real world event detection. We also see this mechanism as a necessary preprocessing and de-noising step for systems mining data from social media.

Our research goal is two-fold: First aim is to *detect* place-triggered geotagged tweets, meaning discovering whether a tweet contains both of the abovementioned location relations. Second aim is to *classify* these place-triggered geotagged tweets by using a method of sequential filtering based on keywords and regular expressions. Our method is based on external classification knowledge, and we focus on measurement of accuracy as well as the relevance of used filters.

Related Work

Recently, many researchers are working for detecting events from social media. When detecting the real world events by leveraging social media, it is important to estimate the place from which the user tweeted.

Sakaki T, et al. detect occurrence of earthquakes in real time by analyzing Twitter stream [8]. In the research, they mainly make use of static locations which are set on user's profile in order to estimate the location where they tweeted. However, users are not necessarily tweeting on that location, because they can tweet from everywhere by their mobile devices. Therefore, inaccurate event locations would be estimated by analyzing inaccurate locations of tweets. In fact, they estimate location error an average of about 300 km from actual source of earthquakes.

Geotagged tweets have the potential to be utilized for real world event detection more accurately. Lee R, et al. detect local events by measuring geographical regularities of geotagged tweets [6]. However, geotagged tweets contain many noises which affect analysis. The research shows low precision rate, in other words, the results contain many unexpected events.

On the other hand, Yin X, et al. propose a method to discover objective information from multiple conflicting data sources

on the web [13]. In a similar fashion, Wang D, et al. estimate the objectiveness of information by using Bayesian and maximum-likelihood methods [10], [11]. Schlieder, C et al. addressed a central problem in the field of social reporting [9]. They proposed an approach to the quality problem that is based on the reciprocal confirmation of reports by other reports. These researches consider only the authenticity of information source.

As illustrated above, when detecting real world events, noise in geotagged tweets affects the accuracy of results negatively. Many previous works take Top-down process for detecting events, but there are also some works proposing the bottom-up approach. Becker H, et al. focus on distinguishing between messages about real world events and non-event messages from the stream of Twitter messages [2]. Muhammad et al. propose a method for automatic tagging of untagged tweets [1]. Both of them use machine learning method to realize classification from large training set of tweets. Similar to these approaches, our research takes the bottom-up process. We attempt to verify whether the geotagged tweets are motivated by events or situations of current locations.

ESTABLISHING THE GROUND TRUTHS

This section describes the method of preliminary survey to classify the geotagged tweets. Then, we classify the type of place-triggered geotagged tweets based on the survey.

Preliminary Survey

In order to investigate how users tweet with geotag, we conducted a survey to classify types of geotagged tweets based on the content of tweet. For this objective, we crawled geotagged tweets in Twitter around Japan from 2011-11-21 00:00:00 to 2011-12-31 23:59:59. We sampled 2,000 tweets from the total amount of 1,977,531, then classified these tweets to certain types based on their content.

The result of classification is shown in Figure 2. We classified 11 place-triggered types based on the content of tweets in total. Most of the tweets (42.5%) were classified as noise, for example, just replying to other follower as "@someone Good morning !". Rest of the tweets were classified as place-triggered tweets. In place-triggered tweets, report of whereabouts type accounted for 74.7%, and the other types were under 8%, each. Report of whereabouts is the content such as "I'm at Keio University now".

Classification of the Place-triggered Geotagged Tweets

From the preliminary survey, many of the place-triggered tweets were confirmed as report of whereabouts. The second most popular type was food, followed by weather information and back at home notifications. The other types were less than 1% of all tweets, so we decided to take account of these four types. Furthermore, we chose the earthquake type in particular, since earthquakes can potentially cause great disaster and thus need to be monitored. Eventually, we classified place-triggered geotagged tweets to five types below:

- Report of whereabouts: A tweet that user refers to his/her current location.

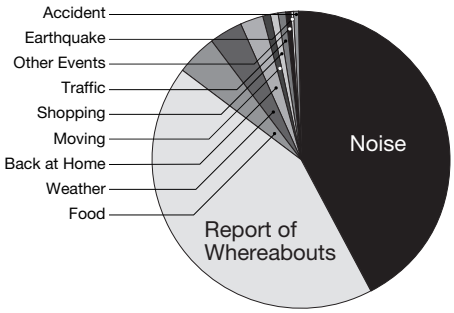


Figure 2. Result of survey of geotagged tweets

- Food: A tweet where user shares information regarding current food or drink.
- Weather: A tweet about weather of the location.
- Back at home: A tweet where user reports the fact that he/she is back at home.
- Earthquake: A tweet in which user reports the feeling of the earthquake.

DETECTION OF PLACE-TRIGGERD GEOTAGGED TWEETS

In this section, we show the approach to detect the place-triggered geotagged tweets. Then, we explain the design and implementation of the proposed system.

Approach

We design filter modules that detect each selected type of place-triggered geotagged tweets. Each filter uses naive keyword matching method to detect the type of tweets, because we assume that people tend to classify tweets mainly by distinctive keywords.

Approaches of each filter module are described below. First, we consider that the tweets from check-in services are the report of whereabouts. From the result of preliminary survey, most of the tweets which classified as the report of whereabouts were made using the check-in services. Many users link their check-in service account to Twitter's, so that we can acquire check-in activity from crawling Twitter. The filter estimates tweets to be a report of whereabouts, if the *source* information of tweets contain Foursquare, Loctouch [7], or Imakoko-now [4] site URL.

The other filters: *food*, *weather*, *back at home* and *earthquake* use a keyword matching method. As an example of keyword matching, we describe the *food* filter. First, we created a list of synonyms of the keyword "food" using Weblio english thesaurus [12]. If text of a tweet contains words in the synonyms list of food, the tweet is considered to be *food* type. In the same way, we make lists of synonyms of the other keywords. For the first step, we prepare synonyms of verb, noun and adjective for each filter.

The results of classification are returned after the tweets are filtered by above filters. Each filter is independent of each other, one tweet may be classified as more than one type.

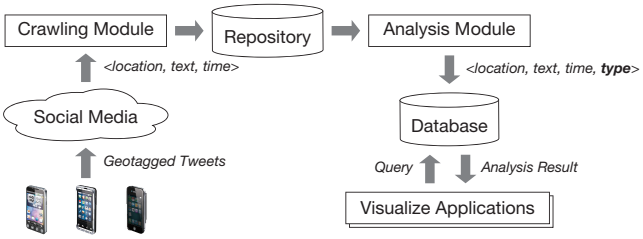


Figure 3. Module configuration

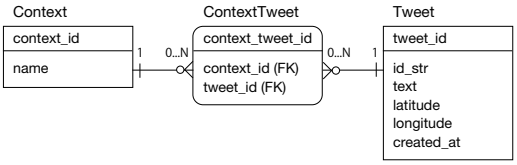


Figure 4. Database structures

As a method of keyword matching, filters use regular expressions. We also examined the possibility of morphological analysis, but decided to apply simple keyword matching method to avoid performance degradation due to object analysis. By increasing the pattern of keywords in the list of synonyms, we have confirmed the accuracy of determination equal to or better than if we use the morphological analysis. Visualization applications are intended to assist in the detection of real world events with place-triggered geotagged tweets. We describe details of implemented applications in Section .

Design and Implementation

We design the system that analyzes and visualizes crawled geotagged tweets. The proposed system is composed of three modules: crawling module, tweets analysis module, and visualization applications. Figure 3 shows the module architecture.

The crawling module collects tweets whose location information is set nearby Japan. Tweets are crawled from Twitter Streaming API in real time. We acquire about 50,000 to 70,000 geotagged tweets per day. In tweets analysis module, tweets are classified into different place-triggered types by using the method described in Section . The classified tweets are saved to the database. Database structure is shown at Figure 4. A tweet can have multiple place-triggered types that are described as Context and ContextTweet tables in the figure. The tweet record has five fields, id_str is unique ID provided by Twitter api, text is contents of tweet, latitude and longitude is location coordnate of the tweet and created_at is the timestamp when user tweeted. The visualization applications are intended to assist in the detection of real world events with place-triggered geotagged tweets. We describe detail of the applications at Section . The system is implemented using Ruby, PHP and MySQL.

API

We define an application programming interface to access the analyzed tweet data stored in the database. The API is currently used for two visualization applications described in Section . Applications invoke the API through an HTTP request, followed by server response containing the result as a JSON dataset. The result contains the tweet list as well as classification information of each included tweet.

We show detail of the API below. Resource URL is:
`http://example.com/api/getTweet`
The abovementioned domain name is tentative, and should be replaced in application implementations. Applications use HTTP GET method to acquire tweets. There are 3 parameters to specify query conditions.

- timestamp
  Returns tweets generated after the given date and duration. Date should be formatted as:
  YYYYMMDDhhmmss:(durationSec)
  Example values: 20120905120000:600

- bounds
  Returns tweets located within a given latitude/longitude pairs of NorthEast and SouthWest rectangle. The parameter is specified by:
  NElatitude,NElongitude,SWlatitude,SWlongitude.
  Example values:
  35.604094,139.585396,35.753852,139.844261

- context (optional)
  Returns tweets which are filterd by the given place-triggered types. Each place-triggered type is identified by integer, which is set in advance by the system. The parameter is specified by comma splitted integer. If more than one ID is specified, filter works through the OR grouping condition. Example values: 1,2,3

Example request is shown below:
`http://example.com/api/getTweet?context=&timestamp=20120905120000:600&bounds=35.604094,139.585396,35.753852,139.844261`
The API returns tweets from 2012-09-05 12:00:00 to 2012-09-05 12:10:00 tweeted on location of the given coordinates:

```
[
    {
        "context": [
            "checkin"
        ],
        "created_at": "Tue Sep 05 12:00:10 +0000 2012",
        "id": 123456789012345678,
        "id_str": "123456789012345678",
        "source":
        "<a href="http://foursquare.com">foursquare</a>",
        "text": "I'm at Tokyo Sta. http://example.com/foo"
    }
]
```

Each hash represents a tweet and multiple tweets can be included in a array. In a hash, the "context" attribute represents place-triggered type, and others are raw tweet data acquired from Twitter API.

VISUALIZATIONS

In this section, we describe details of two application prototypes using the place-triggered geotagged tweets. First, we mention the animation visualizer which is intended to discover the real world events. Then, we describe the web

based interactive interface which is more focused on interactive browsing of analyzed tweets.

Animation Visualizer

We implemented the animation visualizer which plots the place-triggered geotagged tweets on a map. Any range of regions can be visualized, animating images in any time interval. Figure 5 shows the plot of tweet at intervals of 4 hours in April 29, 2012. Column A represents all the geotagged tweets, column B separates place-triggered geotagged tweets, which is shown by o sign, from noise. Column C shows the each type of place-triggered geotagged tweets. Each type of plot shows the report of whereabouts, food, weather, back at home, earthquake and noise.

We discovered some interesting cases by visualizing place-triggered geotagged tweets. On column A, it was detected that cluster surrounded by red circle, A-1 to A-5, is moving northeast at a constant rate. It cannot be estimated whether it is meaningful or not by column A, but it can be determined as noise by column B. It was found to be a Bot that follows a user who tweets with location information. The tweets are not related to the location, but generated randomly. Furthermore, we found that the tweets about the earthquake emerged rapidly in the plot C-1. In fact, a big earthquake occurred at May 28, 2012, 19:28. As explained above, by performing visualization plots for place-triggered geotagged tweets, we have been able to remove noise and detect real world events more easily.

Web-based Interactive Interface

We also implemented the other application which plots place-triggered geotagged tweets on Google Maps. Figure 6 shows a screenshot of the visualization interface. Place-triggered geotagged tweets are plotted on the map, colored by types. By clicking a pin, details of the tweet like texts and pictures can be displayed. Right part of the display shows control panels. You can adjust time windows by sliders, and the tweets are retrieved from the database each time you scroll the map, giving a sense of interactivity. The interface helps users to detect more precise events by displaying detail of each tweets. For example, by enabling filtering by food and picture, popular restaurants often remarked by users can be discovered, and by enabling filtering by weather and picture, guerrilla heavy rain can be visualized.

There are limitations that tweets can only be plotted up to 200 at the same time, from the problem on the performance of the API on Google Maps. Our de-noising method can make it possible to plot more important tweets. We aim to launch this service as a web application, which helps to overcome these limitations.

EVALUATION

In this section, we describe the detail of evaluation study, which was conducted from two perspectives. First, in order to verify whether the classification of place-triggered geotagged tweets is reasonable, we asked 18 people to classify geotagged tweets crawled from Twitter. Next, the results are

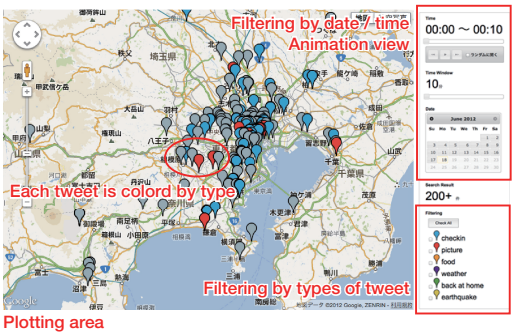


Figure 6. Interactive visualization interface for place-triggered geotagged tweets

compared with the implemented analysis module, in order to evaluate the performance of the module.

Classification of Geotagged Tweets by Third Parties

Methodology

In order to evaluate whether the preliminary classification is reasonable, we asked 18 third party people to classify tweets. Participants were 12 men in their 20s, 5 women in their 20s and 2 men in their 30s. We used different tweets pool from the preliminary classification described at Section . The evaluation targets are geotagged tweet which were crawled from 2012-01-01 00:00:00 to 2012-03-31 23:59:59, total amount is 4,254,257. Each participant reviewed 500 tweets which were randomly sampled from the dataset.

For reduction of effort to review many tweets, we created a web based evaluation tool. Figure 7 shows screenshot of the tool. Before the conduction of evaluation study, we explained the concept of place-triggered geotagged tweets to the participants. The participants see details of each tweets, which contains text, tweeted time and location, and are asked whether the tweet can be classified to any given type. If the tweet cannot be classified to these types, but it is clearly considered to be a place-triggered tweet, the *other* type is selected. On the other hand, if the tweet cannot be considered to be a place-triggered tweet, no types are selected. Multiple selection of types is also allowed.

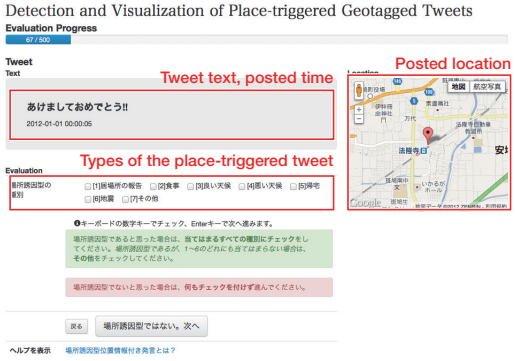


Figure 7. Evaluation tool for tweet classification

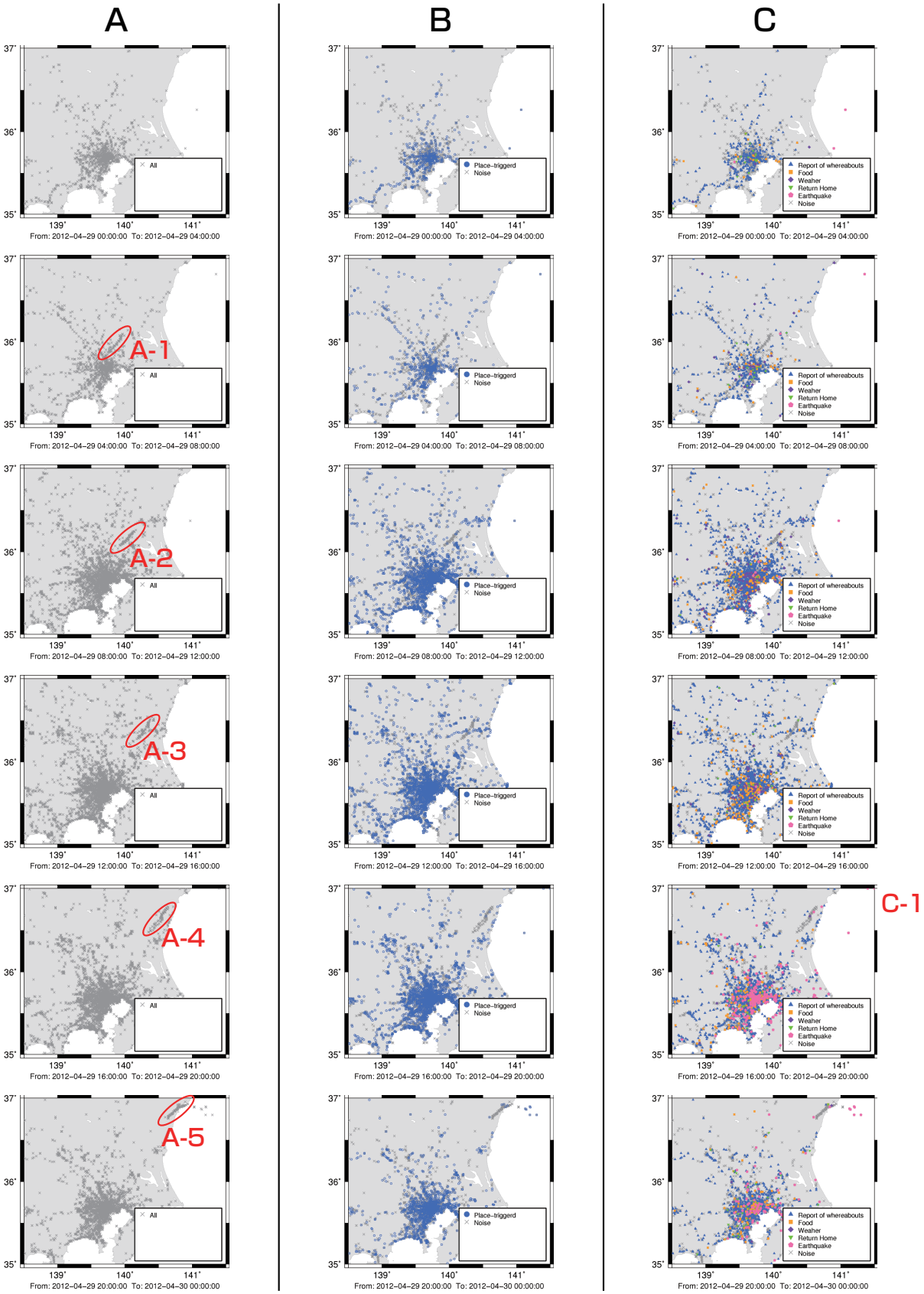


Figure 5. Plotted result of place-triggered geotagged tweets

Table 1. Result of place-triggered geotagged tweets classification by third parties

Type of Tweets	Number	Percentage
Report of whereabouts	4,300	76 %
Food	664	12 %
Weather	255	4.5 %
Back at home	61	1.1 %
Earthquake	44	0.77 %
Other	354	6.2 %
Total	5,678	100 %

Results and Discussions

We collected 8,988 tweets for the ground truths in this evaluation study. Twelve results are missed due to an application or network error. The breakdown of the result is as follows: 5,027 tweets are place-triggered (55.93%) , 3,961 tweets are non place-triggered (44.07%). Further breakdown of the place-triggered types is described in Table 1. Total amount of classification is 5,678, since multiple tagging is allowed.

We consider that the five types of preliminary classifications are mostly appropriate, since most of tweets were classified to these types by the third party people. However, 354 tweets were classified to the other type, which was place-triggered but cannot be classified to the preliminary types, so we decided to investigate these results more closely. For example, "Going out to buy something ..." and "I'm attending the event of ... ." are classified to place-triggered tweets by third parties. Unfortunately, since the standard of classification differs very much by each person, it was difficult to find any tendency within the type *other* to classify any further types.

Performance of the Tweets Classification Module

Methodology

To evaluate the performance of our tweet analysis module, we compared the module output with the ground truth collected in the previous subsection. We used the same tweets pools of that was used in the classification by third parties.

The evaluation is conducted in two aspects. First we examined the accuracy rate, that whether the system correctly detected place-triggered tweets or not. Second, we inspected the accuracy of each place-triggered type. For each type, we define the number of total tweets that is classified as the type by participants as C, number of tweets that is detected as the type by the system as N, and number of correct answer which the system detected as R. Then, we calculated the following measures: $Precision = \frac{R}{N}$, $Recall = \frac{R}{C}$ and $F\text{-}measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$

Results and Discussions

First, we show the overall accuracy of the module. The false-positive and true-negative rate is shown in Table 2. As we show in Table 2, the place-triggered geotagged tweets could be detected with an accuracy of 82%. The false-positive rate is relatively low at 2.18%, meaning the system can get rid of most noise. Whereas, the false-negative rate is still 15.84%, meaning that significant amount of geotagged tweets are missed.

Table 2. Accuracy rate of detecting place-triggered geotagged tweets

	Positive	Negative
True	40.09 %	15.84 %
False	2.18 %	41.89 %

Table 3. Result of place-triggered geotagged tweets classification by system

Type of Tweets	Precision	Recall	F-measure
Report of whereabouts	93.18 %	77.16 %	84.42 %
Food	53.6 %	17.8 %	26.7 %
Weather	57 %	21 %	30 %
Back at Home	54 %	23 %	32 %
Earthquake	76 %	66 %	71 %

Second, we describe accuracies of each place-triggered type. The breakdown of detection accuracy in each type of place-triggered tweets is shown in Table 3.

- Report of Whereabouts
  The report of whereabouts has been detected with high accuracy, as the F-measure shows 84.42%. There are still some missed tweets, because there are tweets reporting whereabouts without using check-in service, thus they cannot be detected by the system for now.

- Food, Weather and Back at Home
  On the other hand, food, weather and back at home types could not be detected well. We consider that there are many tweets which mention the types of food, weather and back at home, without containing words within our dictionary. To improve recall, we should apply not only simple keyword matching method, but more detail method such as language analysis.

- Earthquake
  Earthquake type is detected with a relatively high accuracy. We conceive that keyword matching method is suitable for earthquake, since the report of earthquake occasion tend to be short words.

OPEN ISSUES AND FUTURE WORK

We introduced a new concept of place-triggered geotagged tweets, and presented the result of classification accuracy by using simple keyword matching algorithm. However, several limitations and room for improvement still remain. Here, we briefly discuss these remaining issues and provide implications for future research.

Expanding the classification

Although we classified place-triggered geotagged tweets into five categories based on their relative frequencies, other types of tweets that fulfill the criteria of place-triggered geotagged tweets still remain (see Figure 2). Additionally, since we only focused on Japanese tweets for defining the categories, it is necessary to investigate tweets in other countries. More complete categories of place-triggered geotagged tweets can be useful for future systems aiming to reliably detect real world events.

Improving detection accuracy

We leveraged a simple keyword-matching algorithm to detect place-triggered geotagged tweets. Although we presented reasonable accuracy with the proposed method, more efficient detection methods should be investigated. One promising direction is to utilize results from linguistic analysis research (e.g., unsupervised classification, allowing the system to converge to a finite number of types with significant relative frequencies). Furthermore, as real world places and events are often associated with dedicated terminologies and language constructs, the use of slang should be analyzed.

Discovering real events

We presented two applications for visualizing and interacting with place-triggered geotagged tweets. We consider that users can discover real world events with more ease by analyzing place-triggered geotagged tweets, compared with analyzing all tweets including noise. However, more automatic or efficient methods, such as temporal-spatial analysis of place-triggered geotagged tweets, should be investigated.

Security & spoofing location

When treating location information, we should consider whether the attached location is genuine or not. If a user spoofs his/her location, system might detect non-real events. Additionally, collaborative spoofing by groups of users can severely harm the detection process. In future work, we plan to take these types of attacks into account in the system design.

CONCLUSIONS

Detecting real world events from geotagged tweets whose location have no association with their actual contents significantly restricts overall performance. In this research, we defined *Place-triggered Geotagged Tweets*: Tweets containing both geotag and content-based relation to your location, and designed a system for their detection and classification. We classified the place-triggered geotagged tweets as 5 types: *report of whereabouts*, *food*, *weather*, *back at home* and *earthquake*, based on ground truth established by a survey as well as a study featuring 18 human classifiers. We conducted evaluation study and showed that the system can detect place-triggered geotagged tweets with an overall accuracy of 82%. Furthermore, we implemented visualization applications to detect real world events from place-triggered geotagged tweets. Our work contributes to current state-of-the-art through a pre-survey, design and implementation of a prototype system, as well as an evaluation against a ground truth notion.

ACKNOWLEDGMENTS

This research is partly supported by National Institute of Information and Communications Technology and Research Laboratories NTT Docomo Inc. Third author would also like to thank the Academy of Finland for financial support.

REFERENCES

1. M. Asif Hossain Khan, M. Iwai, and K. Sezaki. Towards urban phenomenon sensing by automatic tagging of tweets. In *Proceedings of the Ninth International Conference on Networked Sensing Systems*, 2011.
2. H. Becker, M. Naaman, and L. Gravano. Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, pages 438–441, 2011.
3. B. De Longueville, R. S. Smith, and G. Luraschi. ”omg, from here, i can see the flames!”: a use case of mining location based social networks to acquire spatio-temporal data on forest fires. In *Proceedings of the 2009 International Workshop on Location Based Social Networks*, pages 73–80, 2009.
4. fujita-lab.com. Imakoko-now. `http://imakoko-gps.appspot.com/`, 2012.
5. A. Jackoway, H. Samet, and J. Sankaranarayanan. Identification of live news events using twitter. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks*, pages 25–32, 2011.
6. R. Lee and K. Sumiya. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, pages 1–10, 2010.
7. NHN Japan Corp. Loctouch. `http://tou.ch/`, 2012.
8. T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, pages 851–860, 2010.
9. C. Schlieder and O. Yanenko. Spatio-temporal proximity and social distance: a confirmation framework for social reporting. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, pages 60–67, 2010.
10. D. Wang, T. Abdelzaher, H. Ahmadi, J. Pasternack, D. Roth, M. Gupta, J. Han, O. Fatemieh, H. Le, and C. Aggarwal. On bayesian interpretation of fact-finding in information networks. In *Proceedings of the 14th International Conference on Information Fusion*, pages 1–8, 2011.
11. D. Wang, L. Kaplan, H. Le, and T. Abdelzaher. On truth discovery in social sensing: a maximum likelihood estimation approach. In *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, pages 233–244, 2012.
12. Weblio, Inc. Weblio thesaurus. `http://thesaurus.weblio.jp/`, 2012.
13. X. Yin, J. Han, and P. Yu. Truth discovery with multiple conflicting information providers on the web. *IEEE Transactions on Knowledge and Data Engineering*, 20(6):796–808, June 2008.

Users Sleeping Time Analysis based on Micro-blogging Data

Haoran Yu
Key Lab. on High Performance Computing, Anhui Province
School of Computer Science
University of Science and Technology of China
haoran_yu@hotmail.com

Guangzhong Sun
Key Lab. on High Performance Computing, Anhui Province
School of Computer Science
University of Science and Technology of China
gzsun@ustc.edu.cn

Min Lv
Key Lab. on High Performance Computing, Anhui Province
School of Computer Science
University of Science and Technology of China
lvmin05@ustc.edu.cn

ABSTRACT
The emergence of new social network services, often labeled as Web 2.0, has permitted an amazingly increase of user generated content. In particular, Sina Weibo, a popular Chinese micro-blogging service is designed as platforms allowing users to generate contents that open to the public. From analyzing activates of submitting posts to Sina Weibo, some features of users can be estimated. This paper aims to contribute to this growing body of literature by studying how users' frequent activities reflect their sleeping time and living time zones. By mining a large set of users' activates data from Sina Weibo, we demonstrate its possible role to detect the sleeping time of users and find a new method for judging users' time zone.

Author Keywords
social networking services, micro-blogging, time series, sleeping time, time zone.

ACM Classification Keywords
H.2.8 [Database Management]: Database Applications—Data mining; H.3.3 [Information Storage and Retrieval]:Information Search and Retrieval—Relevance feedback.

General Terms
Measurement, Documentation, Experimentation, Human Factors, Verification.

1. INTRODUCTION

With the development of social network and related technologies, users, as the core part of the service, communicate with each other and generate a lot of content. Twitter, Facebook, Sina Weibo and other services became the most popular social communication platforms. A growing number of researchers are focusing on related topics. Since considerable features of users are reflected by their activities, we can estimate unknown features of users by analyzing the data of users. Within all the features, the time zone and the corresponding location are both important features, which are related with users' sleeping time.

UbiComp'12, Sep 5 – Sep 8, 2012, Pittsburgh, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

This paper aims to contribute to this growing body of literature by studying how users' frequent activities reflect their sleeping time and living time zones. By mining a large set of users' activates data on Sina Weibo, we demonstrate its possible role to detect the sleeping time of users and find a new method to judge users' time zone. Specifically, the analysis not only estimates the sleeping time pattern of Weibo users but also gives out the authenticity of user's location in their profile and the result of movement between time zones (e.g. from China to U. S.) detection. Although our study involves only simple methods and the general conclusion, it presents a kind of interesting orientation for analyzing users' sleeping activities and time zone detecting.

The main contributions of this paper are as below:

- A new open problem related to the relationship between sleeping time and micro-blogging activities is presented. Besides, this paper associates the pattern of sleep and the time zone users live in.
- For such a problem above, preliminary simple solutions are presented in this paper. Based on experiment results on real data set, the proposed solutions are proved both efficient and effective.

- A data set of users' time series of activities on Weibo is collected and open to the public. The URL is: http://www.haoranyu.com/research/weibosleep

Meanwhile, there are some weak points in this study. Some impact factors, including the job of users, the scale of the city and the individual habits, are not considered. Plus, the technical depth is not very deep so far. Text analysis on micro-blogging data, interactions between friends and geo-tagged sources can be used to further improve the accuracy of prediction in the future.

The rest of this paper is organized as follows. In Section 2, we introduce the Sina Weibo platform and review related works. Section 3 motivates our research and provides our method and results for detecting sleeping time of users. Later in Section 4 we analyzed the relationship between time zone and users' sleeping time. Meanwhile we show some interesting cases. Finally, we conclude the paper in Section 5.

2. RELATED WORK

2.1. Sina Weibo

Sina Weibo[6] (http://weibo.com/), as a micro-blogging platform, akin to a hybrid of Twitter and Facebook; it is used by well over 30% of 513 million users (Up to the end of year 2011) in China. It has a similar market penetration that Twitter has established out of China. Sina Weibo has a verification program for famous people and special organizations. Once an account is verified, a 'V' badge and a verification description will be added next to the account name.

There are several conventions used by Weibo users to convey information within the limit of 140 characters.

- Hashtags (#) userd in the "#Topic#" format are used to add discussion topic and group related posts together.
- Weibo users may talk to or quote other people by using an @username in post.
- Weibo users can re-post with "//@UserName". Similar to Twitter's retweet function.
- URLs posted by user are automatically shortened using the domain name 't.cn'.
- Comments to a post can be shown as a list right below the post.

2.2. Previous Research

There are more and more published studies related user activities in micro-blogging and social network services. The authors of [3] collected data from digg and reddit and get an understanding of how content is generated and how the popularity of a post evolves overtime. Bertrand De Longueville, improve the understanding on how LBSN can be used as a reliable source of spatio-temporal information, by analysing the temporal, spatial and social dynamics of Twitter activity during a major forest fire event in the South of France in July 2009. [2] Kate Ehrlich and N. Sadat Shami (2010) studied users of BlueTwit and Twitter over the same time period. Deepen the understanding of the workplace benefit of micro-blogging and examined how people appropriate social technologies for public and private use.[4] Tony S.M. Tse and Elaine Yulan Zhang (2012) analyzes blog and microblog contents created by mainland Chinese visitors sharing their Hong Kong experiences and find out that a generally positive image of Hong Kong as a destination among the mainland Chinese bloggers.[7] Alan Mislove, Bimal Viswanath, ect discovered that certain user attributes can be inferred with high accuracy when given information on as little as 20% of the users[1].

SHI Xuemin and ZHANG Chuang (2011) studied users' activities on Sina Weibo and found that the peak time that users update posts is different in every time zone. [5] The data set of their research is not properly pretreated. Besides, in their experiment the locations labeled by users are 100% trusted. This study, however, may just be scratching the surface of how time series of micro-blogging can be used for research.

3. DETECTING SLEEPING TIME

Although, more and more researchers are now paying attention to micro-blogging and other similar social network services, most of them have focused on their social dimension by studying users' motivations, interactions or collaboration. This paper pays attention to the posts time series of on Sina Weibo.

3.1. When do users sleep?

According to the common sense, excluding few people who works on the night shift, people are active at day time and inactive while sleeping time at night. However, it is not an easy task to accurately figure out when do users of Sina Weibo sleep. This question will be discussed in this paper.

Sina Weibo is obviously a reliable network service that bundles time and location together. Each of the posts on Sina Weibo has been published with a level of accuracy of 1 minute and the list of posts is organized in reverse order by a natural time. Thus, it is necessary to obtain a sequence of data points of time from Sina Weibo.

3.2. Find the longest inactive time span to discover sleeping time from dense data

This method is used to estimate the span of users' sleeping time under ideal conditions. Under this condition, people sleep for a long time once a day and keep active in the rest of time. Namely, we can't find out two or more long inactive span, equal or greater than 5 hours, within any 1440-minutes-period (as long as a day, that is 24 hours $\times$ 60 minutes /hour). Plus, any of the users must stays in the same time zone and keeps stable activity habits

As depicted in Figure 1, time series data is a sequence of time points, $T = \{t_1, t_2, \ldots, t_n\}$. Each time point $t_i \in T$ corresponds to a time of a post on Sina Weibo.

Figure 1: Time Series

From the time series data, the lengths of spans $D = \{d_1, d_2, \ldots, d_{n-1}\}$ determined by every two points can be defined as:

$$d_i.length = t_{i+1} - t_i$$

All the long inactive spans ($d_i.length \geq 5$h) can be easily detected from a time series data. Each of the result spans is marked as $d'_j = [t_{j0}, t_{j1})$, $j \in (1, k)$. Therefore, a final span $\overline{(d')}$ representing the average sleeping time of a user can be figured out by calculating the average lower bound and upper bound of all result spans.

$$\overline{d'} = \left[\frac{\sum_{j=1}^{k} t_{j0}}{k}, \frac{\sum_{j=1}^{k} t_{j1}}{k}\right)$$

3.3. Statistics method in discovering sleeping time pattern from sparse data

As a matter of fact, the majority of Sina Weibo users are not always keeping active during every whole day period. Namely, the method in section 3.2 is not practical or useful in real application. In the real world, the users' data is much sparse. Less than 1% of Weibo users can satisfy the requirement. In this section, we will present a statistics method in discovering sleeping time pattern from sparse data. In this new method, we still assume that users keep a daily routine for their life, going to bed and waking up on time.

Within any 1440-minutes-period, we may able to find two or more long inactive span, equal or greater than 5 hours. Thus, the statistics method is supposed to be an effective way. Daily data of time series are united together by a section as it is showed in Figure 2.

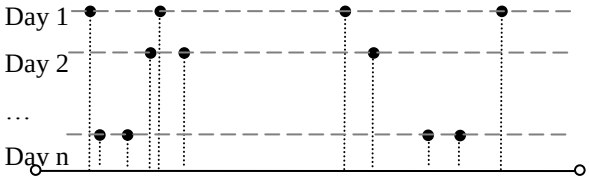


Figure 2: United Time Series

3.3.1. United time series by month

In order to solve this problem in this case, daily data of the time series are united by month for example, as new sequences Tm = {tm1, tm2 … tmn}. Each time point tmi∈Tm corresponds to a time point of a Sina Weibo post with the month m. (See Figure 3)

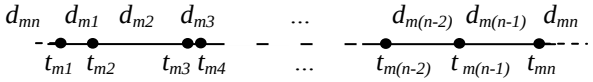


Figure 3: Time Series by Month

In each of the time series data of month m, the lengths of spans Dm = {dm1, dm2, … ,dmn} determined by every two points can be defined as:

$$d_{mi}.length = \begin{cases} t_{m(i+1)} - t_{mi} & (1 \leq i < n) \\ t_{m1} - t_{mn} + 1440 & (i = n) \end{cases}$$

Each of the long inactive spans can be easily obtained from the time series data of the corresponding month. Accordingly, the lower bound and upper bound of user sleeping time in a specific month will be approximately showed.

3.3.2. Subsequence statistics

Yet the method in section 3.3.1 also has defects in practical use. The sparsity of time series data is different from different users. The chosen length of the section for doing statistics cannot match all the circumstances. Some users may frequently submit posts on Sina Weibo and the length of the section for statistics may better be set as a month. At the same time, some other users seldom have a post on Sina Weibo and the length may better be set as long as a quarter (or even a year).

In addition, time points that users start traveling are not always at the first date or last date of the section we chose. In other words, people may have different daily sleeping time during the section of time chosen, which will lead to a meaningless result.

Using the algorithm shown in the following figure, these spans of sleeping time can be estimated automatically from a time series data $T = \{t_1, t_2, \ldots, t_n\}$ of a user.

Algorithm SleepTime_estimation

Input: A time series data of a user *T*
Output: A set of sleeping time *ST*

```
1.  S=∅, i=1, pointNum = |T| // the number of time
2.  point
3.  S.insert(t_i);
4.  while i <= pointNum do,
5.  |  j := i + 1;
6.  |  while !inactSp(S) do,//find no inactive span
7.  |  in S
8.  |  |  S.insert(t_j);
9.  |  |  j:=j+1;
10. |  ST.insert(inactSp(S));
11. |  i:=i+1;
    |  S=∅;
    return ST;
```

The reasons for why we estimate sleeping time span in such a way lie in two aspects. On the one hand, it automatically determines the start point and the shortest length of section needed for finding out an inactive span of time. On the other hand, the sections are different, but overlapped. If we need to figure out the sleeping time with a time span, we can average the result from all sections covered by this time span. Moreover, an empty set or a set with a few elements will be returned if the input data series is too sparse.

However, with the increasing of computing accuracy, the computation efficiency of this method decreases significantly. It will take excessive time for process thousands of time point of each user. Thus, we will not use it for experiment.

3.4. Dataset and Experiment Result

In order to gain information for the present study, a spider program is written to crawl related information from Sina Weibo, obtaining users' location and all the posts they submitted.

844 users are included in the dataset. These users are randomly selected from all the verified users. All individuals of them have more than 500 fans and 100 posts. This requirement ensures that the users are active for a long time and the time series of them will not be too sparse for statistic.

About 1, 579, 623 posts of these users are grasped (started on the 14th of August, 2009 and ended in April, 2012). From each post, a time stamp can be extracted from it. Time series of a user consists of all the time stamps of this user. (The data set mentioned above can be obtained from section 1)

By using the method we introduced in section 3.3.1, the lower bound and upper bound of user sleeping time in a specific month are obtained. Two sample graphs of result are shown in Figure 5 and Figure 6.

4. DETECT TIME ZONES BY SLEEPING TIME SPAN

Location information of users is crawled from their profiles. We first map the time zone of these users from their labeled location. Then we can find a relationship between the detected sleeping time and users' time zone.

4.1. Detect time zone of Sina Weibo users

Users in the same time zone share the similar sleeping time pattern. Namely, each sleeping time pattern can correspond to a time zone that users are live in. Therefore the pattern of each time zone is discovered.

According to the real dataset, most of Sina Weibo users submit less than 10 posts a day. It is obviously sparse. The following experiments are based on the method in 3.3.1. Sleeping time patterns of GMT/ GMT+1/ GMT+6/ GMT+7/ GMT+8/ GMT+9 time zones are as follows. (Not all the data are included. Some users label their location as a country and can't map to a time zone, for example the United States and Australia).

Time zone	Main City	Start sleeping time (GMT+8)	End sleeping time (GMT+8)
GMT	London	7:00 ~8:00	14:00~15:00
GMT +1	Paris	6:00 ~7:00	13:00~14:00
GMT +6	Dhaka	1:00~2:00	9:00~10:00
GMT +7	Hanoi	0:00~1:00	8:00~9:00
GMT +8	Beijing	23:00~0:00	8:00~9:00
GMT +9	Tokyo	23:00~0:00	7:00~8:00

Table 1. Relationship between time zone and sleeping time.

4.1.1. User with fake or unclear location label

Sina Weibo users label their location in the profile without verification. Some of them label the location as 'unknown' or as fake locations.

Fake location labels do considerable harm to the social network services. For example, recommendation algorithm with location factor will be apparently affected.

In our result, 12 users are judged to have obvious fake location labeled in their profiles. (Fake location within the real time zone can't been detected)

For example, Xiaosong Gao (weibo.com/u/1191220232), a famous musician of mainland China, labels his location as 'The United States, Oversea' (GMT-8~GMT-5). But due to our result, his location is judged to be not true. His sleeping time pattern is shown in Figure 4. Nage

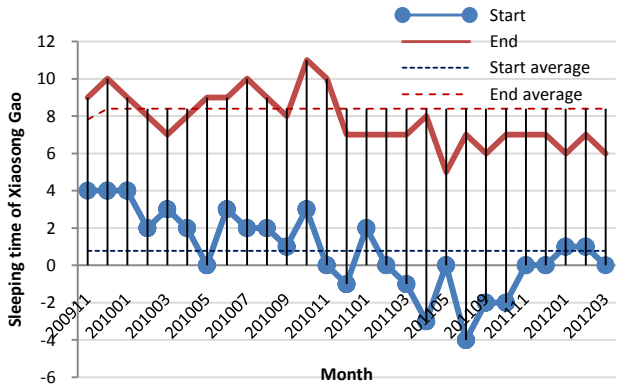


Figure 4: Sleeping time of Xiaosong Gao

The result shows that he sleeps at about 0 a.m. (GMT+8) and wakes up at about 8 a.m. (GMT+8). The corresponding time zone is GMT+7. Therefore, the location Mr. Gao labels himself is fake according to the difference between the location in his profile and the detected time zone.

4.1.2. Detection on movement between different time zones

People are not always located in an area without a movement. There are users who study abroad or have business overseas. They travel far, but they won't change the label of location in the profile page frequently.

Detect and understand the obvious location movements are beneficial. Customized advertising related to locations, such as advertising for local business can be accurately sent to users.

In our result, 33 users have obvious movement (move from locations in different time zones). A user, named Xiaoqian Liu (weibo.com/u/1693775877) describes himself as a reporter of CCTV Brazil and labels his location as 'Brazil, Oversea'. In the result, he is judged to have obvious movement. His sleeping time pattern is shown in Figure 5.

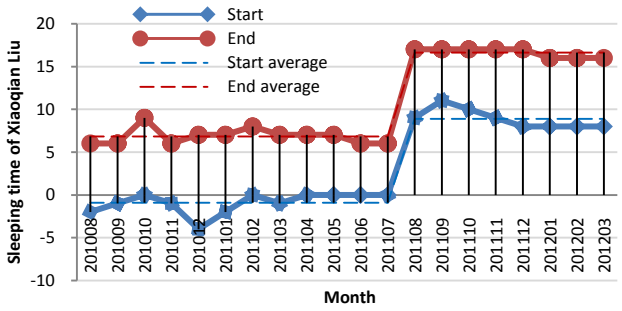


Figure 5: Sleeping time of Xiaoqian Liu

The result indicates that, before Sept. 2011, he goes to bed at about 0 a.m. (GMT+8) and wakes up at about 6 a.m. (GMT+8). The corresponding time zone is GMT+8~GMT+9. After Sept. 2011, the time he starts sleeping change to about 9 a.m. (GMT+8) and the sleep period end at 4 p.m. (GMT+8). The corresponding time zone is GMT-3.5 ~ GMT-2.5. Thus, the obvious movement is detected and the time of this change can be estimated.

5. CONCLUSION

In this paper, three methods for estimating the sleeping time of users are expounded. Broad locations are detected by the linear relationship between sleeping time and time zone of users, which is benefit to study on location-based social networks (LBSN).[8][9] Besides, the study can be extended to other data sets. For example, by mining the web logs on the web server, the real users and robots from specific area can be clearly distinguished. In the future work, some relevant research can be developed to compensating the shortage of this study.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China under the grant No. 61033009 and No. 61103228. It is also supported by the Anhui Natural Science Foundation under the grant No. 1208085QF106.

REFERENCES

1. Alan Mislove, Bimal Viswanath, P. Krishna Gummadi, Peter Druschel. You are who you know: inferring user profiles in online social networks. *WSDM 2010*, (2010), 251-260
2. Bertrand De Longueville, Robin S. Smith, Gianluca Luraschi. "OMG, from here, I can see the flames!": a use case of mining location based social networks to acquire spatio-temporal data on forest fires. *GIS-LBSN 2009*, (2009), 73-80
3. Christian Wallenta and Mohamed Ahmed and Ian Brown and Stephen Hailes and Felipe Huici. Analysing and Modelling Traffic of Systems with Highly Dynamic User Generated Content. *UCL Research Note RN_08_10*, (2008), http://web4.cs.ucl.ac.uk/staff/C.Wallenta/research/wallenta_RN_08_10.pdf.
4. Kate Ehrlich, N. Sadat Shami. Microblogging Inside and Outside the Workplace. *ICWSM 2010*, (2010)
5. SHI Xuemin, ZHANG Chuang. Time Zone Prediction Based on User Behavior of Microblogging. *Science paper Online*, (2011), (in Chinese) . http://www.paper.edu.cn/index.php/default/releasepaper/content/201112-467
6. Sina Weibo. http://en.wikipedia.org/wiki/Sina_Weibo
7. Tony S.M. Tse, Elaine Yulan Zhang. Analysis of Blogs and Microblogs: A Case Study of Chinese Bloggers Sharing Their Hong Kong Travel Experiences, *Asia Pacific Journal of Tourism Research*, (2012), DOI:10.1080/10941665.2012.658413
8. Yu Zheng. Location-based social networks: Users. Computing with Spatial Trajectories, Yu Zheng and Xiaofang Zhou Eds. Springer (2011). ISBN: 978-1-4614-1628-9
9. Yu Zheng. Tutorial on Location-Based Social Networks. *WWW2012*, (2012).

Spatial Dissemination Metrics for Location-Based Social Networks

Antonio Lima
School of Computer Science
University of Birmingham
United Kingdom
a.lima@cs.bham.ac.uk

Mirco Musolesi
School of Computer Science
University of Birmingham
United Kingdom
m.musolesi@cs.bham.ac.uk

ABSTRACT

The importance of spatial information in Online Social Networks is increasing at a fast pace. The number of users regularly accessing services from their phones is rising and, therefore, local information is becoming more and more important, for example in targeted marketing and personalized services. In particular, news, from gossips to security alerts, are daily spread across cities through social networks. Content produced by users is consumed by their friends or followers, whose locations can be known or inferred. The spatial location of users' social connections strongly affects the areas where such information will be disseminated. As a consequence, some users can deliver content to a certain geographic area more easily and efficiently than others, for example because they have a larger number of friends in that area.

In this paper we present a set of metrics that quantitatively capture the effects of social links on the spreading of information in a given area. We discuss possible application scenarios and we present an initial critical evaluation by means of two datasets from Twitter and Foursquare by discussing a series of case studies.

1. INTRODUCTION

Location information is assuming increasing importance in Online Social Networks (OSNs). In particular, a very large number of users is now accessing these services using mobile devices [20, 14]. Certain social networks, such as Foursquare, are built around the very same concept of location [23]. Geo-tagging of posts and photos is becoming popular in Facebook and geographic information is often provided in user profiles and in the generated contents in Twitter. Through these services, information is disseminated across cities, regions, states and the entire planet. Contents of different types are propagated and are consumed by millions of people dispersed around the globe. Understanding the dynamics of the dissemination process is critical for a variety of purposes and to answer a set of fundamental questions that might have important implications for the design of the location-based online network services themselves. For example, to what extent does the geographic distribution of friendships in the network affect where the content will be potentially propagated? Are we able to determine which users are structurally central in delivering information to a specific spatial region? By identifying these users it will be possible to exploit them to deliver information efficiently to specific regions.

General structural properties of large-scale OSNs have already been explored in great detail in the past (see for example [16]). More recently, several works have focussed on geo-social properties of OSNs, focussing for example on the correlation between geography and social topology [19]. Others have investigated co-location and friendship [2, 18] and the possibility of predicting location using friendship information [18]. Indeed, these networks can be considered as a particular class of spatial networks [4]. In the context of complex networks a significant effort has been made to try to give an answer to the question "Which are the most important (i.e., the most central) nodes in a network?". Finding an answer is important because it has strong implications on the processes taking places in networks, such as information diffusion in a social network. The problem has been answered by defining various centrality metrics [17]. All these centrality measures are defined in different ways, by taking into account only social ties (i.e., *topological* information). However, the problem of finding the most important nodes person with respect to the people that are in a specific location (i.e. by using the *spatial* information of the social links) remains open and largely unexplored.

In this paper, we propose information diffusion metrics that capture and quantify geographic importance and centrality of users in geo-social networks. We evaluate these metrics by associating users to one or more locations, using datasets extracted from Twitter and Foursquare. Our metrics focus on the structural properties of the geo-social networks and not on the processes happening over them, such as information cascading and retweeting. Moreover, by separating structure and dynamics, they can be used as quantitative generic tools for evaluating the *potential* role of each node in disseminating information in the geographic space.

The need for modeling spatial social networks and finding measures for quantifying geographic centrality and influence comes not only from the ambition to study the complex interactions between the social and spatial dimensions more comprehensively, but also from a variety of potential

practical applications which could benefit from this analysis. These include:

Targeted information spreading. Being able to measure geographic centrality allows us to rank users according to the number of contacts they have in a certain area. Consequently, they can be used to select individuals to be targeted for spreading information. Applications include not only support for advertisement campaigns of certain products or promotions restricted to given areas, but also the design of systems for dissemination of emergency alerts in natural or man-made disaster situations, where information should be disseminated in a spatially-limited area (for example in case of security alerts in parts of a city or for weather alerts in a certain region).

Models of cultural influence. OSNs are an invaluable source of data for studies in social sciences that were simply not possible in the past [10, 12]. In particular, estimating social and political influence can be very important and relevant for analyzing and interpreting several cultural phenomena. For example, a person tweeting in London might have influence also outside it, for example in his/her hometowns, and in case of recent immigrants, in his/her country of origin. Other possible fields include health studies [7] and economics [21]: until now research in these fields has focussed mainly on the structure of the social networks without considering geographic aspects.

The main contributions of this paper can be summarized as follows:

- Starting from some well-known metrics of centrality and clustering in location-agnostic networks, we define new measures of centrality for quantifying spatial influence, spatial closeness, and spatial efficiency for geo-social networks. We also propose a definition of spatial local clustering coefficient to quantify the presence of *social triangles* in a given location.

- We present a preliminary evaluation of the effectiveness of these metrics by means of two datasets obtained from real world OSNs, namely Twitter and Foursquare, and we discuss the application of these metrics to some realistic application scenarios.

This paper is organized as follows: in Section 2 we introduce the influence metrics; then, in Section 3, we evaluate these metrics by means of the two datasets. We discuss the potential use of these geo-structural metrics for studying dynamic processes in Section 4. Finally, in Section 5 we conclude the paper by discussing future work.

2. SPATIAL INFORMATION DISSEMINATION METRICS

We can represent a social network as a graph $\mathcal{G} = (V, E)$ with N nodes and K links, where nodes are users and links are the social connections between them[1]. We define a *spatial social network* as a social network where each user i is assigned a set of n_i points on Earth $\mathcal{P}_i = \{p_0^{(i)}, p_1^{(i)}, \dots p_{n_i}^{(i)}\}$ including locations that are significant for him/her (e.g., hometown, workplace, favorite restaurant, etc.)[2]. We will firstly introduce a set of accessory definitions that will be used in the remainder of this paper.

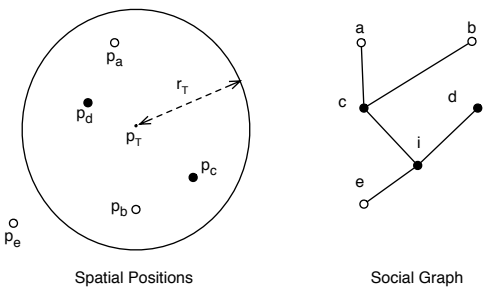


Figure 1: Example of social graph (on the right) and spatial dimension (on the left). In this example the social neighborhood $\mathcal{N}_i$ of i is composed by the nodes c, d, and e; the points of interest of a, b, c and d are inside the spatial neighborhood $\mathcal{S}$, which is the circle of center p_T and radius r_T. As a consequence, the socio-spatial neighborhood $\mathcal{N}_{i,\mathcal{S}}$ indicated with full black dots, does not include node e because it falls outside the spatial neighborhood and also excludes nodes a and b because their positions are located outside the social neighborhood.

- As far the social graph is concerned, we define the neighbors (or connections) of node i as the set of nodes j that are reachable from i through the out-link e_{ij} (content can flow from i to j). The *social neighborhood* $\mathcal{N}_i$ of a node i is the set of all the k_i neighboring nodes of i (e.g., all the followers of user i in Twitter); k_i is often referred to as the degree of node i. This set is defined only on social ties and does not take into consideration any geographic information.

- As far as the spatial dimension is concerned, we use the notation $d_G(p_1, p_2)$ to indicate the geodesic distance between two points on Earth p_1 and p_2. We then define the *spatial neighborhood* $\mathcal{S}$ as an arbitrarily shaped part of the geographic surface; this is a continuous set of geographic points. For simplicity, in this work we will often consider circular regions specified by their center and radius but the definitions presented here can be applied to regions of any shape.

- Given a node j and a geographic region $\mathcal{S}$, the intersection $\mathcal{P}_j \cap \mathcal{S}$ contains all the significant points of j falling inside the region. We define the *socio-spatial neighborhood* $\mathcal{N}_{i,\mathcal{S}}$ of the node i with respect to $\mathcal{S}$ as the set of neighbors j who have at least one significant point inside $\mathcal{S}$:

$$\mathcal{N}_{i,\mathcal{S}} = \{j \in \mathcal{N}_i : \mathcal{P}_j \cap \mathcal{S} \neq \emptyset\}. \quad (1)$$

With $k_{i,\mathcal{S}}$ we denote the number of users in this set. An example is presented in Figure 1.

[1]This representation can be considered as a snapshot of the graph at a given time t. A treatment considering the time-varying nature of the social graphs is outside the scope of this work.

[2]In the simpler case each user can be assigned a single significant location. In the evaluation section we will present two examples covering both cases.

2.1 Spatial Degree Centrality

In general, in a social graph degree centrality is used to rank users according to the number of ties they have within the network [17]; its value is a simple indicator of *influence* and prestige [22]. Methods based on degree centrality are generally used to select the best nodes for spreading information [9]. We extend the concept of degree centrality to spatial social networks with respect to a given spatial neighborhood $\mathcal{S}$ by introducing the concept of *spatial degree centrality*:

$$C_{i,\mathcal{S}} = \sum_{j \in \mathcal{N}_i} |\mathcal{P}_j \cap \mathcal{S}| \,. \tag{2}$$

This value indicates how many significant points the social neighborhood of user i has got inside the considered spatial neighborhood $\mathcal{S}$. If every user is associated only one significant point, this value indicates the size of the audience of user i in the region. In the general case of many significant points for each user, this also takes into account the strength of the potential audience in the region (i.e. social connections with many significant places inside the region give a larger contribution than those with fewer).

The size of the considered region $\mathcal{S}$ affects the calculation of the values of the metrics. For this reason, the size should be set according to the characteristics of the dataset (measurement granularity and precision) and the goal of the analysis itself (for example, researchers might be interested in an analysis at city level). Since the degree of each node also affects this value, a normalization of this metric might also be necessary. The normalization is particularly convenient when comparing users who have a number of followers that differs by orders of magnitude. This might be the case that happens when comparing accounts of news agencies and celebrities, often followed by hundreds of thousands users, with users who have dozens or hundreds of followers. We call the normalized version *spatial degree ratio*, formally defined as:

$$\rho_{i,\mathcal{S}} = \frac{1}{\sum_{j \in \mathcal{N}_i} n_i} C_{i,\mathcal{S}} \tag{3}$$

where n_i is the number of significant places of the user i; this is equivalent, for the one-place case, to:

$$\rho_{i,\mathcal{S}} = \frac{1}{k_i} C_{i,\mathcal{S}}. \tag{4}$$

This metric has values in the range $[0, 1]$. It represents the ratio of connections of i that are inside the area $\mathcal{S}$, therefore it allows to compare nodes that have different degrees in the graph.

These centralities might be considered as simple measures of spatial influence, which can be used in the selection of a user for spreading information to a certain geographic region. However, as they are based on the concepts of geographic membership and social membership, they might not be entirely sufficient to describe the geographic distribution of the neighbors of users. For this reason, in the next subsection we will introduce metrics that take also into account geographic distances.

2.2 Spatial Closeness Centrality

We have defined spatial degree centrality relating to a region. Now we will define a measure of centrality concerning a *punctual* location. Given a target point $p^\star$ on Earth, we define the *spatial closeness centrality* for a user i towards this point as its average geographic distance from all the significant places of his/her connections, formally:

$$C^C_{i,p^\star} = \frac{1}{\sum_{j \in \mathcal{N}_i} n_i} \sum_{j \in \mathcal{N}_i} d_G(p_j, p^\star). \tag{5}$$

This definition is an indicator of how the influenced audience of a user is geographically close to the target point. It can be considered as the spatial counterpart of closeness centrality, which for complex networks is defined as the average distance of the shortest path from the node to all the other nodes [17] it is used as a heuristic when selecting nodes in information diffusion processes [9]. However, this metric might have some drawbacks in specific scenarios given the fact it is calculated as an average of all the distances. This metric can be generalized to the case of multiple locations.

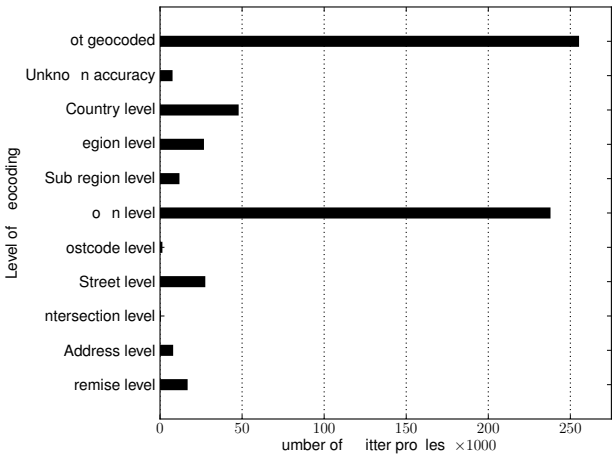


Figure 2: Accuracy of geocoding for the Twitter dataset.

2.3 Spatial Efficiency

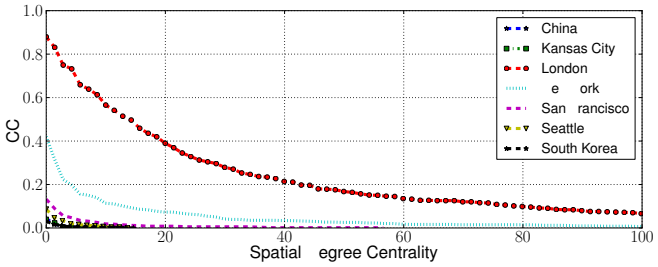
In order to deal with the problem of very large distances which might skew the value of spatial closeness centrality, we define *spatial efficiency* of user i with respect to a point $p^\star$ as follows:

$$C^E_{i,p^\star} = \frac{1}{k_i} \sum_{j \in \mathcal{N}_i} \frac{1}{d_G(p_j, p^\star)}. \tag{6}$$

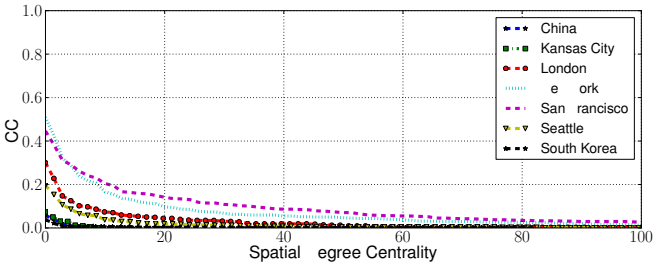
This measure can be thought of as a spatial version of efficiency of traditional graphs [11]. However, this definition has also a potential drawback: if the the neighbor location p_j coincides with $p^\star$ this formula is not defined. For this reason, we modify the above formula by introducing a smoothing decay term as follows:

$$C^E_i(p) = \frac{1}{k_i} \sum_{j \in \mathcal{N}_i} e^{-d_G(p_j, p^\star)/\gamma)} \tag{7}$$

where γ is a scaling factor that can be used to give different weights to the distance $d_G(p_j, p^\star)$. In this formula, the contribution for every neighbor j is at most 1. It is equal to 1 if the neighbor location p_j coincides with $p^\star$, whereas it is negligible if the point is very distant (asymptotically zero if the distance is infinite). This definition can be generalized to multiple locations in a similar way to the formulae presented above.

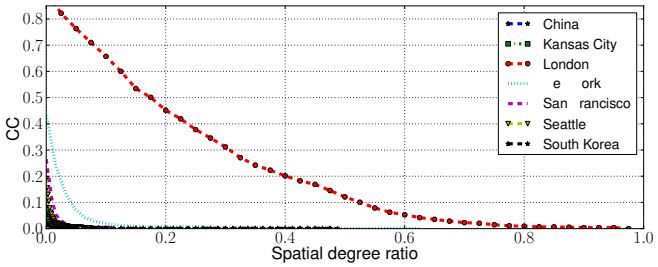


(a) Spatial degree centrality from London.

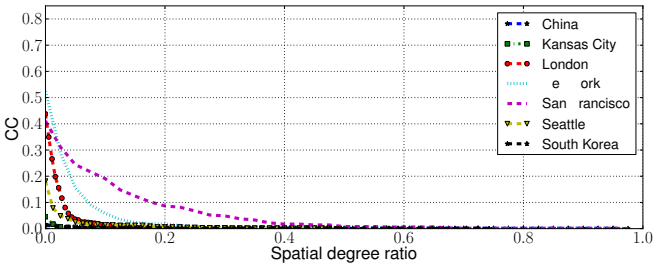


(b) Spatial degree centrality from San Francisco.

Figure 3: Spatial degree centrality of Twitter users in London and San Francisco.



(a) Spatial degree ratio from London.



(b) Spatial degree ratio from San Francisco.

Figure 4: Spatial degree ratio of Twitter users in London and San Francisco.

2.4 Local Spatial Clustering Coefficient

All the definitions presented up to here concern links between pairs of nodes. An interesting measure often used in social network analysis, which deals with triplets of people, is the local clustering coefficient, also called transitivity [22]. It is a local measure quantifying the fraction of triangles among a node and its neighbors. Its spatial version, the geographic clustering coefficient, weights every triangle depending on the geographic distances between the nodes of the triangle [19]. However, this geographic version does not give insights on how neighbors of neighbors might interact in a *specific* geographic region. For this reason, we define the *local spatial clustering coefficient* as the number of triangles present in the socio-spatial neighborhood taken into analysis, formally:

$$C_{i,\mathcal{S}} = \frac{|\{e_{jk} \in E : j,k \in \mathcal{N}_{i,\mathcal{S}}\}|}{k_{i,\mathcal{S}}(k_{i,\mathcal{S}}-1)} \quad (8)$$

where the numerator counts how many links in the social graph are present between users in the socio-spatial neighborhood and the denominator counts how many there could be at most, if they were all connected between each other. The local clustering coefficient measures to which extent neighbors of a node are connected to each other. This metric acquires a special meaning in its spatial version. Nodes scoring high values are part of "social circles", making them potentially highly influential[3]. Social circles defined in this way can be considered a simple example of spatial network motifs, i.e., patterns of interactions on which the network is built [15]. The investigation of the role of spatial network motifs in information dissemination is outside the scope of this work.

[3]At the same time, it is worth noting that, according to some existing theories such as Burt's structural holes [5], nodes scoring low values might also be considered very influential but in a different way as they are able to bring information to users who are not connected between each other, therefore controlling information flows for these users.

3. EVALUATION

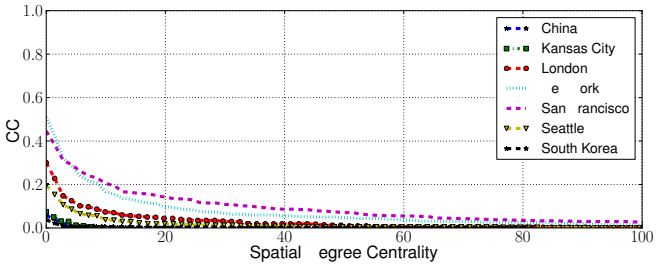
In this section we provide a preliminary evaluation of the proposed metrics. We first present the datasets and we analyze the results deriving from the application of the metrics to different case studies.

3.1 Description of the Datasets

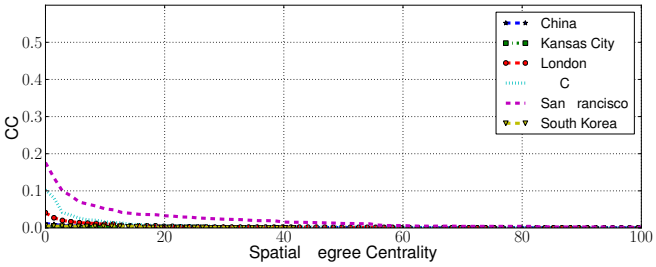
In order to evaluate our metrics, we analyze two popular real-world OSNs, Twitter and Foursquare. In general, datasets were acquired using 2-hop snowball sampling, seeded with random users chosen in well-defined geographic areas. Due to different properties of the two social networking services taken into consideration, the two datasets were obtained following different methodologies, as explained below.

With respect to Twitter, we crawled a dataset containing information about 657,777 users, starting from two evenly distributed sets of 1375 seed users. These were chosen randomly among users that were tweeting from two urban areas, London, UK and San Francisco, California[4]. This location bias was necessary given the nature of our investigation, which requires to have a statistically significant sample of users in the area. It is also worth noting that this can be

[4]The locations for the "seeds" were retrieved from geotags, i.e., spatial tags which are associated to tweets either by automatic geographic sensors as GPS or manually by the user.

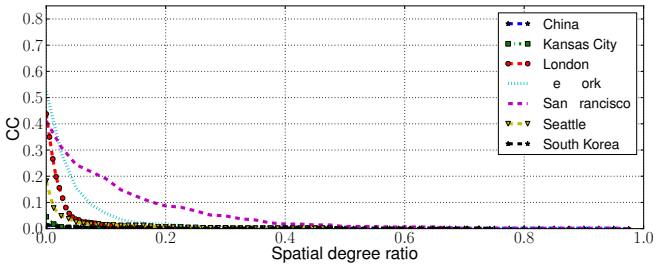


(a) Spatial degree centrality from London.

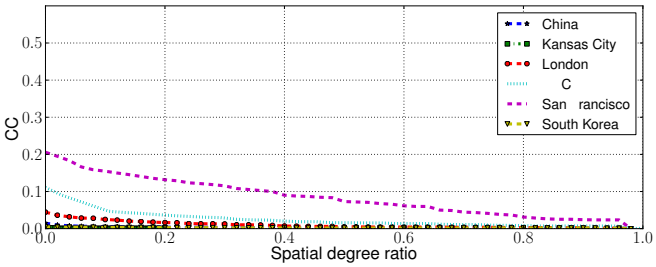


(b) Spatial degree centrality from San Francisco.

Figure 5: Spatial degree centrality of Foursquare users in London and San Francisco.



(a) Spatial degree ratio from London.



(b) Spatial degree ratio from San Francisco.

Figure 6: Spatial degree ratio of Foursquare users in London and San Francisco.

considered as a practical way of retrieving these users for a potential deployment of the algorithms for the calculation of the proposed metrics. We assigned a single significant place to each user, by fetching the information in the "location" field of their personal profile, and converting it to geographic coordinates through the Google Geocoding API. The geocoder was able to identify location for 378,829 users, with different levels of precision; the majority of the identified locations were at town level, according to the distribution shown in Fig. 2, similar to that shown in [8]. We are indeed aware of the fact that locations are not precise and the data are noisy[5].

In Foursquare, a location-based online social network, users "check-in" at venues to let their friends know about their whereabouts, to keep track of their habits and to explore places related to the interests they have in common with other people. The user with the highest number of check-ins over the last 60 days is called the "mayor" of the venue in Foursquare jargon. For this reason, mayorship provides information of potentially strong spatial significance of a certain place for that user. This is also a fine-grained information, as venues are commonly specified at premises level. For this reason, we used the collection of mayorships locations to build the set of significant places. We crawled a dataset of 177,809 users. Since the number of connections in Foursquare is typically smaller than the number of followers in Twitter and the former tend to link with spatially close people [19], our sampling strategy followed a different approach in order to avoid geographically sparse data. We selected a group of interesting urban areas and we crawled venues in the area using the Foursquare API. It is worth noting that these considerations are of great importance for a practical implementation of systems for calculating these metrics in (quasi) real-time, also considering the crawling limitations of the APIs[6].

Finally, we make the simplifying assumption that the rate of change of the network topology is negligible with respect to the information dissemination process taking place over it. This assumption seems reasonable in networks such as Twitter or Foursquare where the rate of change of links is usually very low at the scale of 1 day for example. In fact, the number of new added and removed followers and friends is quite low for a given user after an initial period where a large number of users is added.

3.2 Results

In this section we will present a selection of measurements for each metric. More specifically, in this preliminary study, we choose to compare areas that are heterogeneous from a cultural point of view and different in size.

We also consider two practical case studies. The first is related to the London riots that took place in August 2011: we measure the centrality of Londoners on Croydon, which was one of the theaters of the most violent acts in the British capital. This scenario is an example of usage of this technique in case of emergency. In other words, we are able to answer the following question: *what is the best set of people to target in order to have localized influence through social media in case of natural and man-made emergency and disasters?*

[5]The problem of dealing with noisy data is part of ongoing work.

[6]The Foursquare API returns at most 50 venues per call and does not allow to paginate over all venues in a given large area. Therefore, we queried for venues in categories in small-radius areas (i.e., with a 50 m radius) randomly selected inside the larger areas considered.

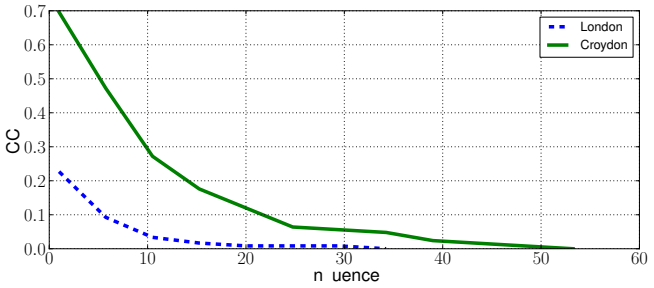


Figure 7: Spatial degree centrality of Foursquare users in Croydon and London towards Croydon.

The second consists in quantifying the centrality of both San Franciscans on people living in Chinatown, and people living in Chinatown on people living in China. It can be seen as an application to the area of geo-demographics [1], aimed at quantifying the potential cultural influence of the inhabitants of certain areas of the city over other areas.

Spatial Degree Centrality and Ratio In Figure 3 we report the Complementary Cumulative Distribution Function (CCDF) of spatial degree centrality of the users located in London and in San Francisco towards four cities (New York, Kansas City, London and Seattle) and two countries (China and South Korea) using the Twitter dataset. We selected the countries by considering the presence of a non-negligible percentage of their population belonging to these ethnic groups. It is possible to observe that both cities have a high degree centrality with respect to themselves, as expected. It is surprising, though, that the degree centrality of Londoners on themselves is very high; in comparison, San Franciscans are not significantly central with respect to their fellow-citizens, and have a self-centrality similar to the centrality shown towards New Yorkers. In our opinion, a possible cause might be that many people who spend most of the day in San Francisco (for example, because their workplace is based there), actually live in the neighboring areas and commute everyday. While 9 users out of 10 in London have at least 1 followers from their own city, only 1 user out of 2 in San Francisco has at least a fellow-citizen reading his content. San Franciscans have some limited potential influence on Londoners, though not as much as on New Yorkers. Users from London and Seattle are also potentially influenced in a substantial way, though not as much as New Yorkers. The countries, China and South Korea, score very low centrality measures in both scenarios, and their curves overlap with those related to Kansas City, the city on which both Londoners and San Franciscans influence the least.

It is worth noting that these results could be influenced by a culture-related tendency to include location information: users from some locations might be keener to include the real personal location, compared to users from other places, due to a different sensibility about privacy issues. Unfortunately, we do not have hard evidence about this fact.

Similar observations can be made for the CCDF for Spatial Degree Ratio in Figure 4. We can notice how the high degree centrality of London with respect to itself is actually connected to a low spatial heterogeneity of followers: nearly one Londoners out of two has *at least* 20 followers living in the same city, while in San Francisco only one out of ten satisfies this property. This peculiar characteristic might be explained both with the tendency of Londoners to follow people from London and with a low interest shown by non-Londoners for the content shared by Londoners. We can also observe how the two highest curves of ratio show a more linear progress, compared to their spatial degree centrality counterparts.

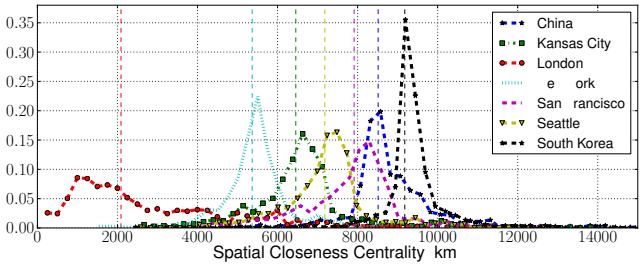
We also perform a similar analysis using the Foursquare dataset. The lower penetration of Foursquare leads to a lower average degree (i.e., on average users in Foursquare have a smaller number of connections than in Twitter) and consequently to smaller centrality values, which include many zeros. However, results shown in Fig. 5 and Fig. 6 are still in accordance with those observed for Twitter. Considering the characteristics of the users in the city, London is again a place of high centrality with respect to itself.

While this city-level analysis can be carried out on the Twitter dataset, we cannot use it for a meaningful analysis at a finer scale, given the nature and quality of the data. Therefore, we use instead the Foursquare dataset, in particular to study the potential influence of Chinatown towards San Francisco and China. The metric is able to identify 8% of users that have a non-null centrality on China and to *rank* them according to their centrality (which quantifies their potential influence over China). When analyzing the average values of the measure, it is interesting to note that the centrality of San Francisco towards Chinatown and the centrality of Chinatown towards itself are comparable (3.2 vs 3.06). This might support the hypothesis that the district is considerably influenced by people living in other parts of the city and that choosing to deliver information to people in Chinatown instead of San Francisco might not have a significant impact on how the information is spread in Chinatown itself. Moreover, given its history and ethnic composition it is not surprising to discover that the average centrality of Chinatown towards China is almost 3 times bigger than the average centrality of the city of San Francisco on China (32.24 vs 11.87).

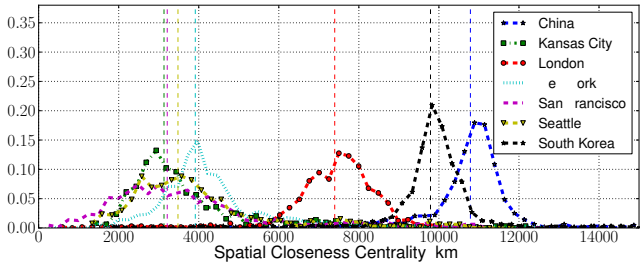
For the Croydon scenario, in Figure 7 we report the centrality of Croydon and London on the Croydon area itself. Users in Croydon appear to have substantially higher values of degree centrality than users of London. This suggests that when disseminating information, targeting people in the area of Croydon, instead of the whole London, might give an advantage in reaching the area of Croydon itself.

Spatial Closeness Centrality In Figure 8 we represent the probability distribution function for the seven distributions of spatial closeness centrality. For each curve, a dashed vertical line represents the median. We can firstly notice that for both cities taken into consideration, London and San Francisco, the closeness centrality curves are more spread out compared to the other curves, which are generally narrower and characterized by a series of peaks. London shows this behavior with stronger emphasis; this can be another evidence of the the high locality of London followers. By definition, geographic constraints have a strong impact on this metric; therefore, we would expect that the peak and the median are very close to the physical distance between the considered points. Indeed, this is the case for all the pairs we report in the figure.

Spatial Efficiency In order to characterize spatial efficiency, we set the value of γ equal to the maximum radius of the geographic area taken into consideration. Figure 9 shows the CCDF for the Twitter users in London and San

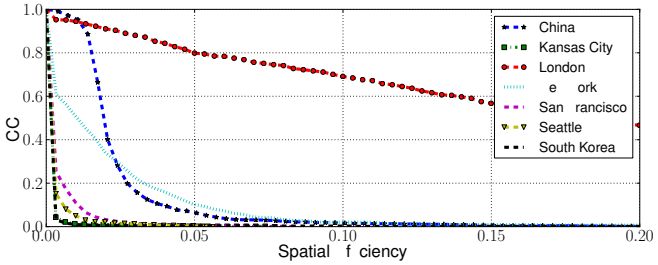


(a) Spatial closeness centrality from London.

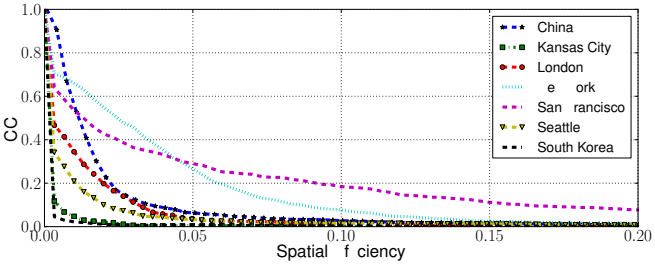


(b) Spatial closeness centrality from San Francisco.

Figure 8: Spatial closeness centrality of Twitter users in London and San Francisco.



(a) Spatial efficiency centrality from London.



(b) Spatial efficiency centrality from San Francisco.

Figure 9: Spatial efficiency centrality of Twitter users in London and San Francisco.

Francisco with respect to the areas considered for the other metrics. As this measure emphasizes the role of neighbors which are close to the location taken into consideration, we can see how the efficiency of London with respect to itself stands out from all the other curves.

Local Spatial Clustering Coefficient The local spatial clustering coefficient allows to identify how many "social triangles" are present in a specified area. As an example, we compute this metric for neighbors of Twitter users which indicated their location in the London area. The percentage of null values is quite high (88%) indicating that a small number of Londoners have social circles in their own city. In Figure 10 we show the CDF for the non-null values.

4. DISCUSSION

When defining metrics to determine how users are influential in a social network, or equivalently how central they are in the process of information diffusion, it seems natural to consider quantities related to the level of actual engagement of users (e.g., how many elements they share, how many reactions/retweets they receive in turn from their friends and so on) and about the semantics of the shared content (e.g., whether it is multimedia content, news links, games, etc.). Such measures can give information about the role of the user in the network and also about his/her topics of interests. For each topic he could either be a pure provider of content or a pure consumer of content or, as it happens more commonly, a combination of the two. The goal of this work is to explore spatio-social centralities relying only on structural properties of social networks, without considering data derived from processes taking place in the network, such as information diffusion [13].

This might be considered a limitation of the current metrics. However, our goal is to propose a generic set of metrics that can be used as the basis for an analysis of the dynamic processes happening over them [3].

Popular OSNs are used by millions of users and handle massive amounts of data. Given the fact that the proposed metrics are calculated on very large datasets, computational complexity is a key issue. First of all, we observe that the metrics we have defined are *local*: in order to perform the calculation we do not need global information about the entire graph. Given a specific location, described by a geographic point or surface, in order to determine the measures defined above for a set of n users, we need to know the coordinates of the neighbors' significant places. Spatial degree centrality, spatial closeness centrality and efficiency measures scale as $\mathcal{O}(nkt)$ where k is the expected number of neighbors of each node and t is the expected number of significant points for each neighbor. In order to determine local spatial clustering coefficient we also need to retrieve the neighbors of neighbors of the starting node (so that we can determine if two of his/her neighbors are neighbors in turn); the complexity of the calculation of this metric scales as $\mathcal{O}(nk^2t^2)$.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented metrics for quantifying potential information dissemination in social networks where geographic information is associated to each user. We have evaluated these metrics by means of two datasets extracted from Twitter and Foursquare by analyzing different realistic case studies, which might be relevant for emergency communications and social sciences. The applications of these metrics are many, including targeted location-aware mar-

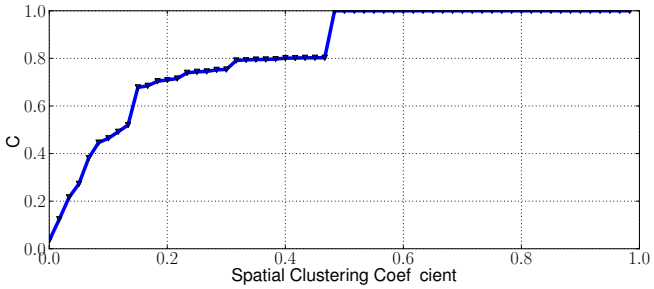


Figure 10: Spatial clustering coefficient of Twitter users in London towards London itself.

keting and efficient information spreading during emergency events.

We plan to extend this analysis by taking into consideration explicit actions (such as retweets or mentions, in Twitter [13, 6]). We also plan to explore the aspects related to the implementation of these algorithms to extract these indicators in real-time.

Acknowledgements

The authors would like to thank Manlio De Domenico for his insightful comments about an earlier draft of this paper. This work was supported through the EPSRC Grant "The Uncertainty of Identity: Linking Spatiotemporal Information Between Virtual and Real Worlds" (EP/J005266/1).

6. REFERENCES

[1] D. I. Ashby and P. A. Longley. Geocomputation, geodemographics and resource allocation for local policing. *Transactions in GIS*, 9(1):53–72, 2005.

[2] L. Backstrom, E. Sun, and C. Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of WWW'10*, pages 61–70, New York, NY, USA, 2010. ACM.

[3] A. Barrat, M. Barthélemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, 2008.

[4] M. Barthélemy. Spatial networks. *Physics Reports*, 499:1–101, 2011.

[5] R. Burt. *Structural Holes: The Social Structure of Competition*. Harvard University Press, 1994.

[6] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring user influence in Twitter: The million follower fallacy. In *Proceedings of ICWSM'10*. AAAI, 2010.

[7] N. A. Christakis and J. H. Fowler. The Spread of Obesity in a Large Social Network over 32 Years. *New England Journal of Medicine*, 357(4):370–379, 2007.

[8] B. Hecht, L. Hong, B. Suh, and E. H. Chi. Tweets from Justin Bieber's heart: the dynamics of the location field in user profiles. In *Proceedings of CHI'11*, pages 237–246, New York, NY, USA, 2011. ACM.

[9] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the Spread of Influence in a Social Network. In *Proceedings of KDD'03*, pages 137–146. ACM, 2003.

[10] J. Kleinberg. The convergence of social and technological networks. *Communications of the ACM*, 51:66–72, Nov. 2008.

[11] V. Latora and M. Marchiori. Efficient behavior of small-world networks. *Phys. Rev. Lett.*, 87:198701, Oct 2001.

[12] D. Lazer et al. Computational Social Science. *Science*, 323:721–723, February 2009.

[13] K. Lerman and R. Gosh. Information contagion: An empirical study of the spread of news on Digg and Twitter social networks. In *Proceedings of ICWSM'10*. AAAI, 2010.

[14] R. MacManus. Facebook mobile usage set to explode, October 2011. ReadWriteWeb.

[15] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[16] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of IMC'07*, pages 29–42, New York, NY, USA, 2007. ACM.

[17] M. Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[18] A. Sadilek, H. Kautz, and J. P. Bigham. Finding your friends and following them to where you are. In *Proceedings of WSDM'12*, pages 723–732, New York, NY, USA, 2012. ACM.

[19] S. Scellato, C. Mascolo, M. Musolesi, and V. Latora. Distance Matters: Geo-social Metrics for Online Social Networks. In *Proceedings of WOSN'10*, Boston, MA, USA, June 2010.

[20] E. Schonfield. Meeker Says Majority of Pandora's and Twitter's Traffic is Mobile, October 2011. TechCrunch.

[21] O. Sorenson. Social networks and industrial geography. *Journal of Evolutionary Economics*, 13:513–527, 2003.

[22] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[23] Y. Zheng. Location-based social networks: Users. In *Computing with Spatial Trajectories*. Eds. Springer, 2011.

LBSNRank: Personalized PageRank on Location-based Social Networks *

Zhaoyan Jin
National University of Defense Technology
Changsha, China
jinzhaoyan@163.com

Dianxi Shi
National University of Defense Technology
Changsha, China
dxshi@nudt.edu.cn

Quanyuan Wu
National University of Defense Technology
Changsha, China
wqy.nudt@gmail.com

Huining Yan
National University of Defense Technology
Changsha, China
yhnbj@qq.com

Hua Fan
National University of Defense Technology
Changsha, China
fh.nudt@gmail.com

ABSTRACT

Different from traditional social networks, the location-based social networks allow people to share their locations according to location-tagged user-generated contents, such as check-ins, trajectories, text, photos, etc. In location-based social networks, which are based on users' checkins, people could share his or her location according to checkin while visiting around. However, people's locations change frequently and the rankings of people change dynamically too, which makes ranking on graphs a challenging work. To address this challenge, we propose the LBSNRank algorithm on graphs with nodes whose contents change dynamically. To validate our algorithm on real datasets, we have crawled and analyzed a dataset from the Dianping website. Experiments on this real dataset show that our LBSNRank algorithm performs better than traditional personalized PageRank in efficiency.

Author Keywords

location-based social networks, personalized PageRank, Monte Carlo method, random walk, MapReduce.

ACM Classification Keywords

H.2.8 Database Applications: Spatial databases and GIS.

General Terms

Algorithms, Experimentation, Performance

INTRODUCTION

With the rapid development of mobile devices and wireless broadband access, the market of mobile internet grows up quickly and lots of new services come up. As two of the most important mobile internet services, location-based services and mobile social network services developed separately in the past. But nowadays, we have been seeing a convergence of the two, and a new kind of social networks, called location-based social network (LBSN for short), is becoming increasingly popular and hundreds of millions of users are active on a daily basis. In traditional social networks, users maintain their relationships mainly among friends in the virtual world, while in LBSN, users can also know new friends online and enhance their relationships according to the offline activities. Obviously, the LBSN offers us a better experience of communication. For more about LBSN, one can refer to Zheng [28] for details.

The LBSN services not only allow users to tweet, reply or retweet, but also allow users to post their locations while tweeting. For example, Foursquare[1] and Dianping[2] are two popular web sites of this kind, which allow users to checkin while tweeting. As time goes on, each user has a history of locations which reflects his or her interests, hobbies, habits, etc. A simple LBSN can be seen in figure 1. Scellato et al. [22, 23] find that, most geographical distances among friends are short-distance links and users like to make friends with nearby people in a specific location. So how to choose popular locations and how to ascertain influential people in some specific location are two important topics in LBSN. Consider the following two scenarios.

(1) As a traveler, you are prepared to visit the Great Wall in Beijing. You can find some influential people in Beijing especially in the area of the Great Wall in advance, and then you can browse their comments, talk with them, and even make friends and live with them in the physical world.
(2) After visiting the Great Wall, you also want to visit some other popular places in Beijing.

*This work was supported in part by the National Significant Science and Technology Special Project of China (Nos. 2011ZX03002-004-01 and 2009ZX01043-002-004) and the National Natural Science Foundation of China (No. 90818028.)

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.

[1] www.foursquare.com
[2] www.dianping.com

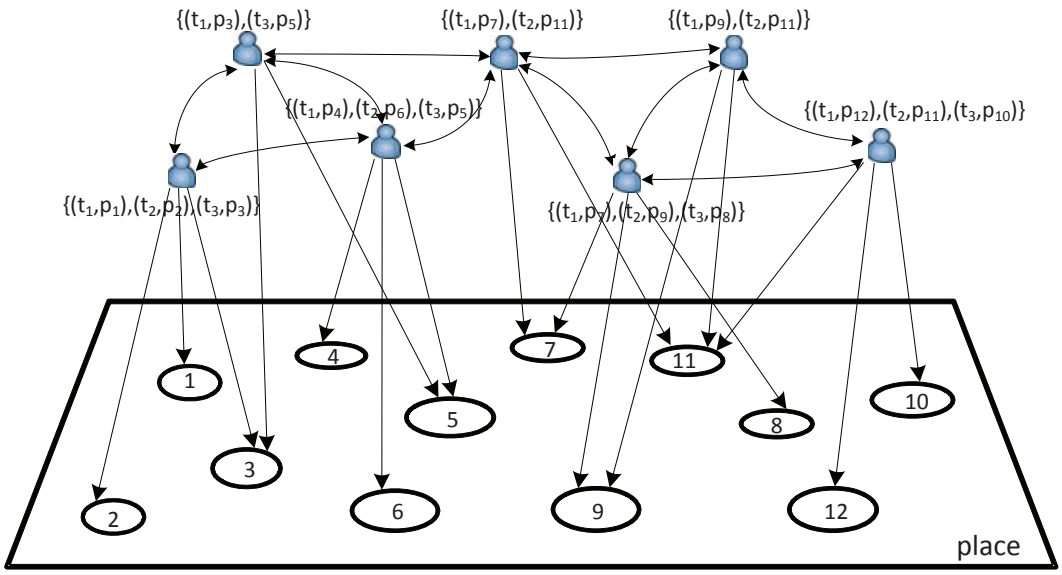


Figure 1. A simple LBSN

In order to find popular places and influential people in specific location, ranking algorithms, such as PageRank [17], HITS [14], Monte Carlo method [16, 10] and the algorithms derived from them, can be used to compute the rankings of people in a LBSN. These algorithms deal with graphs with static nodes, which assume that the contents of the nodes in a graph never change. However, in LBSN, people's locations change frequently and the rankings of people and locations change dynamically too, which makes ranking on such graphs a challenging work.

In this paper, we aim to rank users and locations with respect to some specific location in a LBSN, which means ranking on a graph based on users' relationships and location histories. As there are many users and locations in a LBSN, ObjectRank [4] suggests precomputing all rankings offline in order to answer users' queries quickly. In order to save computing and storage resources, HubRank [6] suggests precomputing some fixed important rankings offline. However, in a LBSN, the checkin records of users change with time frequently, and it isn't advisable to preprocess a fixed location set, and thus we adapt the offline location set dynamically according to locations' popularity. We propose the LBSNRank algorithm for graphs with nodes whose contents change dynamically, have crawled and analyzed a real dataset from the Dianping website, and validate the efficiency of the proposed LBSNRank algorithm.

The rest of the paper is organized as follows. In section 2, we introduce some background for the PageRank and the Monte Carlo method based PageRank and review some related work. Our LBSNRank algorithm is given in section 3. In section 4, we analyze some characteristics of our dataset, and experiments are given in section 5. Conclusion and future work is given in section 6.

BACKGROUND AND RELATED WORK

In this section, we first introduce some background for the PageRank algorithm and the Monte Carlo method of PageRank computing, and then review some work related to ranking on graphs.

Background

We assume to have a weighted graph $G = (V, E)$ with n nodes and m edges, and the weight on an edge (u, v) ($(u, v) \in E$) is denoted with $w_{u,v}$. For the sake of simplifying the presentation of the formulae, we assume that the weights on the outgoing edges of each node sum up to 1.

PageRank

The PageRank method computes the stationary distribution of a random walk. Starting from a source node, we walk randomly in a graph. At each step, we jump to a personalized node with the probability ϵ, and walk along the current node's outgoing edges with the probability $1 - \epsilon$. When the number of steps is big enough, no matter which source node we choose, the stationary probability of each node tends to be a constant, and is called its PageRank score.

With respect to a source node u, the PageRank score $\pi_u(v)$ of node v satisfies:

$$\pi_u(v) = \epsilon\delta_u(v) + (1 - \epsilon) \sum_{\{w|(w,v)\in E\}} \pi_u(w)w_{w,v} \quad (1)$$

At each step, we jump to a personalized node with the probability ϵ, so how to choose the personalized node decides the classification of PageRank. The naive PageRank is that we choose the personalized node fully randomly, which mean that $\delta_u(v) = \frac{1}{n}$ for all $v \in V$, the topic-sensitive PageRank is that we choose different nodes for different probabilities, which is $\delta_u(v) = p_v$ and $\sum p_v = 1$ for all $v \in V$, and the

personalized PageRank is the same as PageRank except that we jump to the source node at all jumps, where $\delta_u(v) = 1$ if $u = v$, and 0 otherwise.

Monte Carlo Method of PageRank Computing

In order to compute the personalized PageRank Score of each node, we can either use linear algebraic techniques, such as Power Iteration [17], or approximate the personalized PageRank score using the Monte Carlo method, which simulates several real random walks and then estimates the stationary distribution with the empirical distribution of the performed random walks. Based on this idea, Litvak [16] and Fogaras et al. [9, 10] propose the following approximation method for personalized PageRank: Starting from each node $u \in V$, do a number, R, of random walks and at each random step, stop with the probability of ϵ, and do the random walk with a probability of $1 - \epsilon$. When it stops, The nodes visited are called "fingerprints", and the length of a fingerprint conforms to the geometrical distribution Geom(ϵ). Then, the frequency of visits to each node in all fingerprints could approximate the PageRank score.

Related Work

Webpage Ranking: Webpage ranking is a hot-spot research in the World Wide Web. Classical methods, such as HITS [14] and PageRank [17] are both link analysis algorithms. HITS assumes that the ranking of a page contains authority and hub, where authority means that a site receives many citations and the citation from important sites weights more than less important sites, and hub means that a site links to many authoritative sites. Because the computation process is carried out online and needs lots of computation resources, it can not satisfy users' queries timely. In contrast, PageRank computes only one measure of ranking for each page offline. The heuristic underlying this is that the ranking of a page is proportional to its parents' ranking, and inversely proportional to the number of its parents' outgoing edges. The PageRank method has a better efficiency, which is the reason why google successes. Following the success of PageRank, a lot of link analysis algorithms, such as Block-level PageRank [5], HostRank[6], hierarchical rank [26], etc., are proposed. However, there is a common disadvantage in link analysis algorithms, that is, the ranking of a page depends on the link structure of webpages and they ignore the contents of the query at hand.

Personalized PageRank: Chakrabarti et al. [7], Pennock et al. [18] and Richardson et al. [19] demonstrate that the properties of web graphs are sensitive to page topics, and Haveliwala et al. [12, 11] propose the TS-PageRank (Topic-Sensitive PageRank) and the personalized PageRank. Instead of using a single PageRank score to represent the importance of a page, TS-PageRank calculates a vector of PageRank scores for every page according to the 16 topics in ODP[3]. The TS-PageRank contains two processing steps, offline and online. In the offline step, the algorithm computes a PageRank score for each topic based on different transition matrix, while in the online step, it computes the linear combination of the vector for every page according to the query context and returns the ordered pages. The TS-PageRank method deals with queries that are relevant to a limited number of topics. For fully personalized PageRank, HubRank [6] and BinRank [13] propose approaches that generate a vector for every possible query term. Since the query terms are so great that they can not be preprocessed completely, so they classify the query terms, such as fixed classification and classification based on similarity, and preprocess the classified ones.

Monte Carlo Method: The Monte Carlo method proposed by Litvak [16] and Fogaras et al. [9, 10] is an efficient method to compute personalized PageRank. It simulates several real random walks with fingerprints and then approximates the personalized PageRank with the empirical distribution of the performed walks. However, some of the fingerprints maybe very long, and hence limit the efficiency of parallel algorithms. Sarma et al. [21] propose the idea of doing random walks of fixed length starting from each node, which improves the parallel efficiency greatly. The Monte Carlo method is not only efficient, but also allows to perform continuous update of the PageRank as the structure of the graph changes [3]. Avrachenkov et al. [1] demonstrate that the Monte Carlo method provides good estimation of PageRank for relatively important pages already after one iteration. As the MapReduce programming model becomes more and more popular, the optimal implementation of Monte Carlo approximation of personalized PageRank vectors of all the nodes in a graph is studied by Bahmani et al. [2].

Social Entity Ranking: Besides webpage ranking, social entity ranking in social networks also considers ranking on large-scale social graphs. TunkRank [24] and TwitterRank [25], both of which are variants of PageRank, measure the rankings of users based on a graph constructed by the "following" relationships in Twitter. The difference is that TwitterRank considers both the topic similarity between users and the following relationships, whereas TunkRank measures rankings on Twitter based on how much attention one's followers can actually give him. In addition, the IP-influence algorithm [20], similar to HITS, takes influence and passivity into consideration while it calculates rankings in social graphs. In social networks, users can tweet, retweet and reply about a topic. These actions can also construct graphs, such as the user-tweet graph [15] and the action-based user influence graph [27]. Based on these graphs, we can also rank users according to their corresponding definitions.

Travel Recommendation in LBSN: Location histories of people in LBSN ont only reflect where they have gone, but also imply the location correlation in people's daily lives [29]. With this correlation, a lot of valuable services, such as travel recommendation, sales promotion and bus route planning, could be enabled. Zheng et al. [30, 31] study the problem of locations and travel sequences recommendation from user-generated GPS trajectories. They extract stay points from these trajectories, construct a Tree-Based Hierarchical Graph, and then mine points of interest (POIs) and classic travel sequences. In order to compute the score of

[3]http://www.opendirectoryproject.com/

each POI, they construct a HITS-based inference model. In that model, locations and users have a mutual reinforcement relationship. But in this paper, we rank users according to their relationships and sort locations according to their popularity. As the links of graph and contents of nodes change with time, we adjust the offline location set dynamically in order to shorten the response time to users' queries.

LBSNRANK

In this section, we introduce some notations, state the problem of ranking on LBSN, and describe the proposed LBSNRank algorithm.

Notations and Problem Definition

In a LBSN, we view users as nodes, and the following / followed relations among users as directed edges, thus we can get a weighted directed graph $G = (V, E)$, where $|V| = n$ and $|edge| = m$. We denote the weight on an edge (u, v) ($(u, v) \in E$) with $w_{u,v}$ ($w_{u,v} = \frac{1}{out(u)}$, where $out(u)$ means the number of out-links of u). We also have a location history L_G, where each node $u \in V$ has a list l_u ($l_u \in L_G$) of location-timestamp pair (t, p), i.e., $l_u = \{(t_i, p_i)\}$, which means that the user u visited the location p_i at time t_i.

We begin by defining the **ranking of a user** with respect to some location in some period:

Definition 1. Given a social graph G and its corresponding location history L_G, the ranking score of a node u, in location p between t_1 and t_2 with the place-timestamp pairs $l_{u,p}(t_1, t_2) = \{(t_i, p)|(t_1 \leq t_i \leq t_2)\}$, is decided by its personalized PageRank:

$$\pi_{p,t_1,t_2}(v) = \epsilon\delta_{p,t_1,t_2}(v) + (1-\epsilon) \sum_{\{o|(o,v)\in E\}} \pi_{p,t_1,t_2}(o) w_{o,v} \tag{2}$$

Where $\delta_{p,t_1,t_2}(v) = \frac{|l_{v,p}(t_1,t_2)|}{\sum_{o\in V} |l_{o,p}(t_1,t_2)|}$.

Next, we define the **ranking of a location** in some period:

Definition 2. The ranking of a location p in the period between t_1 and t_2 is proportional to the number of visitors, the ranking and visited times of each visitor, that is:

$$r_p(t_1, t_2) = \sum_{u\in V} \pi_{p,t_1,t_2}(u) \times |l_{u,p}(t_1, t_2)| \tag{3}$$

LBSNRank Algorithm

In this section, we describe our LBSNRank algorithm for fully personalized PageRank. In LBSN, there are such many locations that people have visited that it is impossible to preprocess rankings with respect to all locations. As a matter of fact, most people have visited only a few of locations, so we need only to preprocess rankings of people for those popular locations offline, and compute rankings of people for less popular locations online. Because the popularity of locations changes frequently, the location set that we preprocess offline changes frequently, too. In the following algorithm, we adjust the location set that will be preprocessed offline at every iteration.

The LBSNRank algorithm:
1. determine the location set Sp that will be preprocessed;
2. compute the ranking of the location $p \in Sp$;
3. compute the ranking of people for each location $p \in Sp$, and go to step 1 for next iteration;

Determination of Location Set

There are two reasons that we can't compute all personalized PageRank scores offline. One is that the set of locations is so big that computing and storing all personalized PageRank scores would need lots of computing and storage resources, and even a small dataset would take several days in our experiments. The other reason is that people update their locations so frequently that if we don't compute the required personalized PageRank timely, then the results returned would be meaningless. In order to answer queries timely, we choose a set of popular locations (details is in Section), for each of which we compute its personalized PageRank offline. This not only answers users' queries timely with preprossed results but also reduces the computing and storage resources by an order of magnitude or more.

Ranking for Locations

As time goes on, the popularity or ranking for each location changes with time. For example, an important sport activity may make some location popular at that time, whereas the change of seasons may make other locations comfortable or attractive in fixed seasons. Initially, we assume all rankings of people are equal, that is $\pi_p(u) = \frac{1}{n}$, for all $p \in Sp$ and $u \in V$, and the ranking of location p is $r_p = \frac{1}{n}\sum_{u\in V} |l_{u,p}|$. Since the second iteration, the rankings of locations are computed according to Definition 2.

Ranking for Users

To compute the rankings of people with respect to some location, we employ the fixed length random walks ([21]) on the reduced subgraph. To reduce the graph in some period about a given location, we select all users who visit that location in that period, and expand their neighbours, including inlinks and outlinks. Algorithm 1 describes the details of our random walks. In algorithm 1, the personalized vector $V_p = \delta_{p,t_1,t_2}$, which can be calculated according to equation 2, and the choosing of the timestamp pairs t_1 and t_2 between two consecutive iterations depends on your actual requirements and the ability of your platform.

Query Processing

For a given query, if it requires some locations, the LBSNRank algorithm returns high ranking locations from the location set, and if it requires some people in a location, then the LBSNRank algorithm returns top-k people relevant to that location. As the LBSNRank algorithm computes rankings with respect to only a few of locations, the query may require both preprocessed and unprocessed locations. If all the required locations is preprocessed, the LBSNRank algorithm returns a linear composition of them. Otherwise,

Algorithm 1 MonteCarloK(G_r, Vp, ϵ, k)

Input: A reduced subgraph $G_r = (V, E)$, the personalized vector V_p, the possibility ϵ for restart and the length k for each fingerprint;
Output: A list l of $(v, frequency)$, for all $v \in V$ in G_r;

```
let l = {(v,0)|∀v ∈ V};
for all v ∈ V do
    let current = v;
    for i = 1 to k do
        if Random.nextDouble() < ε then
            //next random walk;
            next = current.neighbours.Random();
            (next, frequency) = (next, frequency + 1);
            current = next;
        else
            //restart;
            for i = 1 to |Vp| do
                if Random.nextDouble() <= Vp_i then
                    next = i;
                    (next, frequency) =
                    (next, frequency + 1);
                    current = next;
                end if
            end for
        end if
    end for
end for;
```

if it requires unprocessed locations, then the LBSNRank algorithm computes unprocessed locations online, and returns a linear composition of them. By default, queries are accompanied by the latest timestamp pairs, but if a query requires older ranking, the LBSNRank algorithm deals with it online.

DATASET

To evaluate our method in a real LBSN, we have crawled the Dianping website to collect a dataset of users, their social links and checkin histories. We have crawled about 200,000 users from the Dianping website. The dataset are crawled from Dec. 19, 2011 to Feb. 26, 2012, and table 1 lists some basic statistics of this dataset.

Table 1. Statistics of the Dianping Dataset

# users	# links	# checkins
204,074	926,720	2,730,072
# cities	# districts	# POIs
347	1,691	313,565

Time-Checkin Distribution

The Dianping website prevents us from crawling the whole checkin records, so we can't have a complete view of users' checkin histories. In our dataset, the checkins are from Jan. 2011 to Feb. 2012, and the distribution of which is in figure 2. As can be seen from the figure that, the number of checkins grows up quickly from Jan. 2011 to Aug, 2011, and after that, it grows smoothly. The reason is that the Dianping website has been developing quickly since its introduction of location-based services. Moreover, the crawling is begin with the homepage, so most of the dataset are up-to-date.

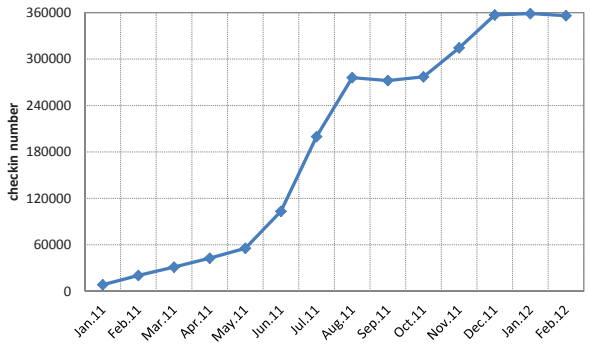


Figure 2. the Time-Checkin Distribution

Location-Checkin Distribution

The 80/20 rule is probably one of the most powerful ideas, which is universally applicable in the daily activities of our society. The underlying idea is that only a small part, about 20%, is important in a group, whereas the other 80% is trivial. In this paper, wo show that only a small number of locations, less than 20%, are popular among over 80% people, and many less popular locations have been visited by only a few people.

In this paper, we study the problem of ranking on LBSN in some location, and the locations in our checkin records are the name of the concrete geographic locations, such as POIs, districts and cities, instead of geographic coordinates. A POI is a small location such as restaurant, cinema or square, a district is bigger location that contains many small POIs, and a city is the same as our intuition. We analyze the location-checkin distribution in our dataset, and the details can be seen in figure 3. As can be seen from the figure that, about 20% POIs have been visited by nearly 80% people. As for districts or cities, about 10% of them have been visited by more than 98% people. This is because only a number of districts or cities are prosperous, and they attract more people. Table 2 lists the top-10 cities that people usually checkin in China.

Table 2. Top-10 Cities

1	2	3	4	5
Shanghai	Beijing	Guangzhou	Tianjin	Nanjing
6	7	8	9	10
Shenzhen	Hangzhou	Shuzhou	Wuhan	Xi'an

Degree-Checkin Distribution

The checkins in our dataset are also related to the degrees of people. Figure 4 illustrates the distribution of checkins according to people' degrees. The in-degree of a people is the number of people that follow him or her, the out-degree means the number of people he or she follows, and the degree is the sum of in-degree and out-degree. As can be seen from figure 4 that, the distribution of checkins is mainly decided by the in-degree. The reason is that when people have

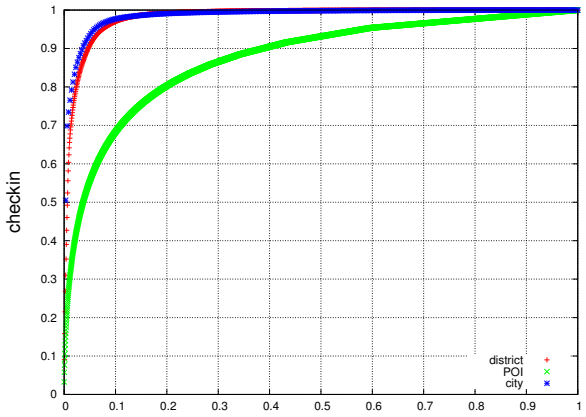


Figure 3. the Location-Checkin Distribution

more checkin records and comment more about those corresponding locations, they will attract more followers.

EXPERIMENTS

In this section, we present the results of the experiments that we have done to test the performance of our algorithm.

Experimental Setup

MapReduce [8], proposed by Google, is a programming model for processing huge amounts of data in parallel using a large number of commodity machines, and its open-source implementation is Hadoop[4]. By automatically handling the lower level issues, such as job distribution, data storage and fault tolerance, it allows programmers without any experience in parallel and distributed systems to easily utilize the resources of a large distributed system.

In this paper, we implement the LBSNRank algorithm in Java on top of the Hadoop platform. Our experiments are executed on a cluster of 20 nodes, where each node is a commodity machine with a 2.16GHz Intel Core 2 Duo CPU and 1GB of RAM, running CentOS v6.0. In order to demonstrate the robustness of our algorithm and to show its performance on realistic data, we present experiments with the Dianping dataset that we have crawled. Details of the dataset can be seen in section .

Experimental Results

In this section, several experiments are performed to compare the proposed LBSNRank algorithm with the traditional personalized PageRank.

Efficiency Evaluation

It needs several days to compute personalized PageRank with respect to all districts. Even if this is tolerable, then returning popular POIs or influential persons several months ago would be meaningless. There are two methods that can reduce the execution time, i.e., preprocessing a number of PageRank scores offline and compressing the graph with smaller subgraph. The LBSNRank algorithm is a composition of the two, figure 5 illustrates the comparison of execution time.

[4]hadoop.apache.org/common/docs/r0.16.4/hdfs_design.html

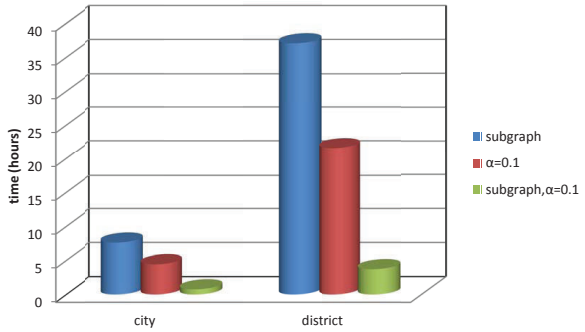


Figure 5. Comparison of Execution Time, where $\alpha = 0.1$ is the location ratio that we preprocessed offline

Though we already computed some popular personalized PageRank scores offline, there are still many personalized PageRank scores to compute online when queries are not hit. If the online computation takes much time, then a lot of queries would be blocked, and the throughput of the system would decrease. But as the locations become less popular, the people that visit them become fewer and fewer, so the subgraph of a LBSN becomes smaller and smaller, and we need less time to compute personalized PageRank scores online. Details can be seen in figure 6. Because the hadoop platform need some time to start-up the virtual machines and to process the input, the execution time cannot be small enough, and it tends to be a constant.

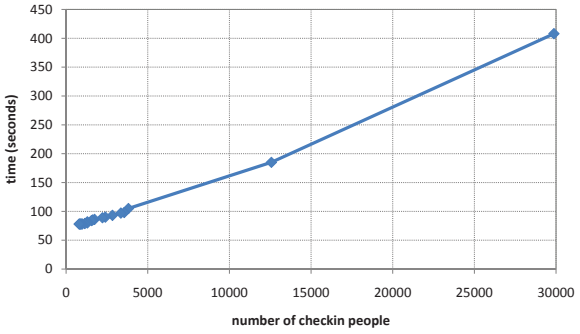
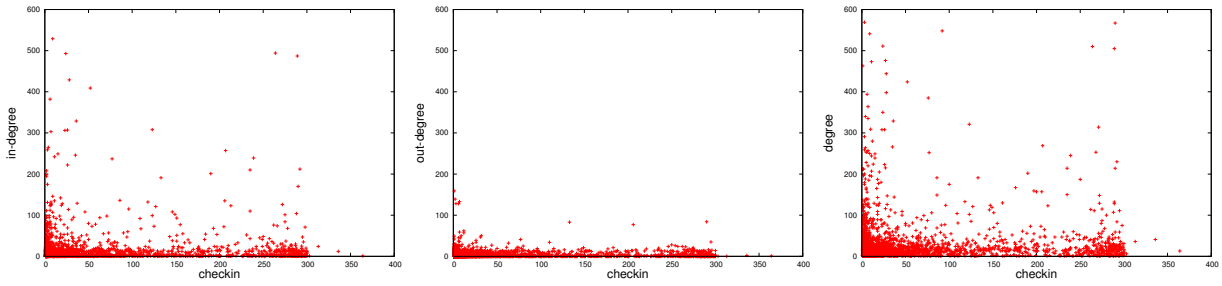


Figure 6. the Subgraph-Time Distribution

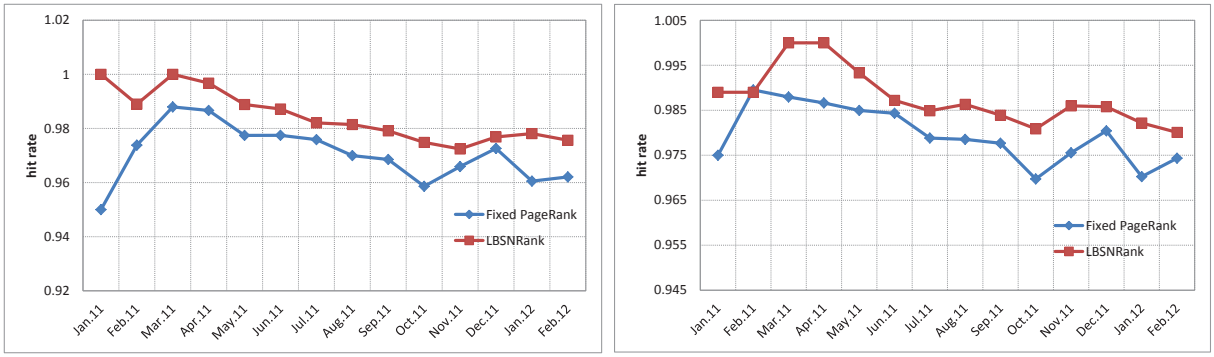
Hit Rate Evaluation

The hit rate of query is the ratio of queries that require results that have been preprocessed. When a query is issued by a user, the system seeks and returns the highest ranking results for that query. Because we only preprocess a number of personalized PageRank, there are some queries that require personalized PageRank that we haven't preprocessed yet. If we require a location that hasn't been preprocessed, then it will take more time to compute it online, and this will degrade the efficiency of the system. Therefore, if we improve the hit rate, we would save more time and resources, and hence improve the efficiency of the system. Though HubRank [6] computes some fixed personalized PageRank carefully, the



(a) in-degree (b) out-degree (c) degree

Figure 4. the Degree-Checkin Distribution



(a) city, $\alpha = 0.1$ (b) district, $\alpha = 0.1$

Figure 7. Hit Rate Comparison, where $\alpha = 0.1$ is the location ratio that we preprocessed offline

popularity of location changes quickly, which makes the hit rate of query even lower. In this experiment, we choose 10% ($\alpha = 0.1$) locations according to their popularity in the whole dataset to preprocess offline. In contrast to HubRank's fixed set, the proposed LBSNRank algorithm adjusts the location set that will be computed offline. In our experiments, we choose 35 ($\alpha = 0.1$) most popular locations as preprocessed location set in each month. In order to testify the accuracy of the LBSNRank algorithm, we choose 1% records of the following moth as test queries, and study the hit rate of query, details can be seen in figure 7. From the figure we can see that the LBSNRank algorithm has a better hit rate than the fixed personalized PageRank.

CONCLUSION AND FUTURE WORK

We study the problem of LBSNRank, i.e., the personalized PageRank on location-based social networks which are based on users' checkin histories. We rank locations based on their popularity, and for a specific location, we rank users based on their relationships and checkin records. As users in location-based social networks change their locations frequently, the rankings of locations and personalized PageRank scores of users change frequently as well. As MapReduce programming model becomes increasingly popular, we evaluate our experiments on its open-source implementation Hadoop. In order to validate our LBSNRank algorithm on real dataset, we have crawled a dataset from the Dianping website, and also analyzed some characteristics of users' checkin records. Experiments on this real dataset show that our LBSNRank algorithm is not only efficient, but also improves the hit rate of query.

In this paper, we precompute rankings with respect to popular locations, and thus we can answer users' queries timely with these results. However, it takes several days to rank all the districts in our experiments, and as for POIs, the execution time would be unacceptable. Though we precompute only a few of them offline, the computation still needs much time. Therefore, how to answer users' queries timely with up-to-date results would be our future work.

REFERENCES

1. K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova. Monte carlo methods in pagerank computation: When one iteration is sufficient. *SIAM Journal on Numerical Analysis*, 45(2), 2007.

2. B. Bahmani, K. Chakrabarti, and D. Xin. Fast personalized pagerank on mapreduce. In *Proceedings of the 2011 international conference on Management of data*, pages 973–984. ACM, 2011.

3. B. Bahmani, A. Chowdhury, and A. Goel. Fast incremental and personalized pagerank. *Proceedings of the VLDB Endowment*, 4(3):173–184, 2010.

4. A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *Proceedings of the Thirtieth international*

conference on Very large data bases-Volume 30, pages 564–575. VLDB Endowment, 2004.

5. D. Cai, X. He, J.-R. Wen, and W.-Y. Ma. Block-level link analysis. In *The 27th ACM/SIGIR International Symposium on Information Retrieval*, pages 440–447, New York, NY, USA, 2004. ACM.

6. S. Chakrabarti. Dynamic personalized pagerank in entity-relation graphs. In *Proceedings of the 16th international conference on World Wide Web*, pages 571–580. ACM, 2007.

7. S. Chakrabarti, M. Joshi, K. Punera, and D. Pennock. The structure of broad topics on the web. In *Proceedings of the 11th international conference on World Wide Web*, pages 251–262. ACM, 2002.

8. J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

9. D. Fogaras and B. Rácz. Towards scaling fully personalized pagerank. *Algorithms and Models for the Web-Graph*, pages 105–117, 2004.

10. D. Fogaras, B. Rácz, K. Csalogány, and T. Sarlós. Towards scaling fully personalized pagerank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, 2(3):333–358, 2005.

11. T. Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526. ACM, 2002.

12. T. Haveliwala, S. Kamvar, and G. Jeh. An analytical comparison of approaches to personalizing pagerank. Technical Report 1999-31, Stanford University, 2003.

13. H. Hwang, A. Balmin, B. Reinwald, and E. Nijkamp. Binrank: Scaling dynamic authority-based search using materialized subgraphs. *Knowledge and Data Engineering, IEEE Transactions on*, 22(8):1176–1190, 2010.

14. J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

15. S. Kong and L. Feng. A tweet-centric approach for topic-specific author ranking in micro-blog. In *Proceedings of the 7th international conference on Advanced Data Mining and Applications - Volume Part I*, ADMA'11, pages 138–151, Berlin, Heidelberg, 2011. Springer-Verlag.

16. N. Litvak. Monte carlo methods of pagerank computation. *Department of Applied Mathematics, University of Twente*, 2004.

17. L. Page, S. Brin, R. Motwani, and Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.

18. D. Pennock, G. Flake, S. Lawrence, E. Glover, and C. Giles. Winners don't take all: Characterizing the competition for links on the web. *Proceedings of the National Academy of Sciences*, 99(8):5207, 2002.

19. M. Richardson and P. Domingos. The intelligent surfer: Probabilistic combination of link and content information in pagerank. *Advances in neural information processing systems*, 14:1441–1448, 2002.

20. D. Romero, W. Galuba, S. Asur, and B. Huberman. Influence and passivity in social media. *Machine Learning and Knowledge Discovery in Databases*, 6913:18–33, 2011.

21. A. Sarma, S. Gollapudi, and R. Panigrahy. Estimating pagerank on graph streams. *Journal of the ACM (JACM)*, 58(3):13, 2011.

22. S. Scellato, C. Mascolo, M. Musolesi, and V. Latora. Distance matters: geo-social metrics for online social networks. In *WOSN'10*, Berkeley, CA, USA, 2010. USENIX Association.

23. S. Scellato, A. Noulas, R. Lambiotte, and C. Mascolo. Socio-spatial properties of online location-based social networks. *Proceedings of ICWSM*, 11:329–336, 2011.

24. D. Tunkelang. A twitter analog to pagerank. *The Noisy Channel*, 2009.

25. J. Weng, E. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *In Proceedings of the Third ACM International Conference on Web Search and Data Mining*, 2010.

26. G. Xue, Q. Yang, H. Zeng, Y. Yu, and Z. Chen. Exploiting the hierarchical structure for link analysis. In *The 28th ACM/SIGIR International Symposium on Information Retrieval*, pages 186–193, New York, NY, USA, 2005. ACM.

27. M. Zhang, C. Sun, and W. Liu. Identifying influential users of micro-blogging services: A dynamic action-based network approach. In *PACIS Proceedings*, 2011.

28. Y. Zheng. Location-based social networks: Users. In Y. Zheng and X. Zhou, editors, *Computing with Spatial Trajectories*, pages 243–276. Springer New York, 2011.

29. Y. Zheng and X. Xie. Learning location correlation from gps trajectories. In *Mobile Data Management (MDM), 2010 Eleventh International Conference on*, pages 27–32. Ieee, 2010.

30. Y. Zheng and X. Xie. Learning travel recommendations from user-generated gps traces. *ACM Transaction on Intelligent Systems and Technology (ACM TIST)*, 2(1):2:1–2:29, Jan. 2011.

31. Y. Zheng, L. Zhang, X. Xie, and W. Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th international conference on World wide web*, pages 791–800. ACM, 2009.

Followee Recommendation in Asymmetrical Location-Based Social Networks

Josh Jia-Ching Ying, Eric Hsueh-Chan Lu, and Vincent S. Tseng*
Institute of Computer Science and Information Engineering
National Cheng Kung University
No.1, University Road, Tainan City 701, Taiwan (R.O.C.)
{jashying, ericlu416 }@gmail.com, *Correspondence: tsengsm@mail.ncku.edu.tw

ABSTRACT
Researches on recommending followees in social networks have attracted a lot of attentions in recent years. Existing studies on this topic mostly treat this kind of recommendation as just a type of friend recommendation. However, apart from making friends, the reason of a user to follow someone in social networks is inherently to satisfy his/her information needs in asymmetrical manner. In this paper, we propose a novel mining-based recommendation approach named Geographic-Textual-Social Based Followee Recommendation (GTS-FR), which takes into account the user movements, online texting and social properties to discover the relationship between users' information needs and provided information for followee recommendation. The core idea of our proposal is to discover users' similarity in terms of all the three properties of information which are provided by the users in a Location-Based Social Network (LBSN). To achieve this goal, we define three kinds of features to capture the key properties of users' interestingness from their provided information. In GTS-FR approach, we propose a series of novel similarity measurements to calculate similarity of each pair of users based on various properties. Based on the similarity, we make on-line recommendation for the followee a user might be interested in following. To our best knowledge, this is the first work on followee recommendation in LBSNs by exploring the geographic, textual and social properties simultaneously. Through a comprehensive evaluation using a real LBSN dataset, we show that the proposed GTS-FR approach delivers excellent performance and outperforms existing stat-of-the-art friend recommendation methods significantly.

Author Keywords
Location-Based Social Network (LBSN), Followee Recommendation, Semantic Similarity, Data mining.

ACM Classification Keywords
H.2.8 [Database Management]: Database Applications – *Data Mining, Spatial Databases and GIS*

General Terms
Performance, Design, Experimentation.

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

INTRODUCTION
With the rapid growth and fierce competition in the market of social networking services, many service providers have deployed various recommendation services, such as friend recommender, to promote users to understand each other in order to grow the underlying social networks. For example, several well known social networking systems, such as Facebook, Twitter, and FriendFeed, they have provided various services on friend search and recommendation. These services are very useful for users to find people who have similar interests, learn and share information/experiences with others, and make friends. Based on our observations, we could categorize these social networks into two classes:

- *Symmetrical Social Networks* (*SSNs*) that correspond to the general social relationship of users. This kind of social network is always represented as undirected graph, such as Facebook, Gowalla and Foursquare.
- *Asymmetrical Social Networks* (*ASNs*) that are likely represented as directed graph, such as Tweeter and Everytrail. In this kind of social network, users can follow other users whom they are interested in. If users follow somebody, they will receive notifications when their followees upload new trips or do something special on the social network website.

As contrasted with *SSNs*, the concept of social activity on *ASNs* is more complicated. Because people may not only want to make friend when they follow someone, they probably are more interested in the information which is provided by someone [12]. In other words, if some people have information needs, they will try to search and follow the persons who have the information. Here, we call this kind of asymmetric relationship "information-need relationship". As shown in Figure 1, user *A* and user *B* are friends each other if they have link in a symmetrical social network (see Figure 1(a)), but the reason user *A* follows user *B* may be user *A* and user *B* have the information-need relationship (i.e., user *B* provides some interesting

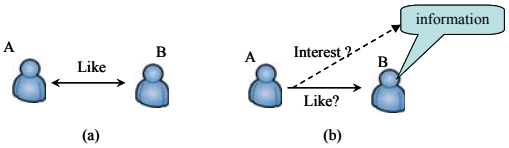


Figure 1. Two Types of Social Networks.

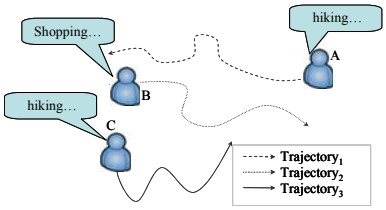


Figure 2. A scenario of Information Needs.

information for user *A*). As the result, *ASNs* always contain these two kinds of relationship, i.e., social relationship and information-need relationship. Thus we argue that these two kinds of relationship must be considered for followee recommendation.

However, most of the followee recommendation engines (called followee recommenders) just directly adopt friend recommendation techniques for recommending followees on *ASNs*. In other words, they only use the concept of social relationship to make recommendations (e.g., some systems often recommend followees' followees to their users) instead of capturing the information-need relationship. As the result, the existing works focus only on analysis of social properties, like followee of followee link, common followee, etc., to make recommendation. We argue that this recommending strategy could not work well on information-need relationship. The reason is that the social properties could not illustrate complete information-need relationship. For example, suppose that two users follow a lot of hikers, but the reasons of that the two users follow these hikers may be totally different. One of the two users may just like hiking and another may likes the pictures which are provided by these hikers. Accordingly, it is necessary to involve more information to make followee recommendation.

Although there are several previous studies [9, 10, 11]on Location-Based Social Networks (LBSNs) involve the information of user movements for potential friend recommendation, these existing techniques mostly focus on analyzing the similarity of moving sequences (i.e., geographical or semantic trajectories). Due to the experience binding of users' movements, these recommendation techniques only recommend people for the users who have geographical or semantic common movements. Take Figure 2 as an example, there are three trajectories provided by the three different users. There is no user who is similar to user *C* since there is no trajectory which is similar $Trajectory_3$. Thus, the traditional recommendation techniques would suffer for the problem of experience-limitation (i.e., only recommending the users who have similar movements). However, the textual information, such as travelogues and comments of trips, did not be involved in the existing work. Actually, the textual information could represent the intension of users' preference or fancy. Take Figure 2 as an example again, we can see that the user *A* and user *C* often talk about "hiking" in their provided textual information. Thus, we may recommend them to each other as their followee.

To address the above-mentioned problem, we propose a novel approach named *Geographic-Textual-Social Based Followee Recommendation (GTS-FR)* for recommending users the followees based on not only social factors but also users generated data. As shown in (1), given a set of users *U*, the problem of followee recommendation can be formulated as classifying the relation of a given ordered user pair, *u* and *v,* into the binary class, *1* and *0*. Here, class *1* means that user *u* follows user *v*, and class *0* means that user *u* does NOT follow user *v*.

$$f(u \mid v) \rightarrow \{0,1\}, \text{where } u \in U \text{ and } v \in U \quad (1)$$

Note that $f(u|v) \neq f(v|u)$ because the "follow" is asymmetric relation. Hence, followee recommendation in LBSN can be addressed as the problem of binary-class classification for each individual user (i.e., to classify all other users into "followee" class and "non- followee" class). While binary-class classification techniques have been developed for many applications, such as protein function classification [4], music categorization [6] and semantic scene classification [2], the problem has not been explored previously under the context of asymmetrical LBSN. Furthermore, the geographical and textual information changes quickly especially in LBSNs. How to extract appropriate features to support the recommendation from such heterogeneous data is also a critical and challenge issue. To support followee recommendation based on user-generated data and social properties, we address this problem by learning a SVM classifier for each individual user. To do so, a fundamental issue is to identify and extract a number of descriptive features for each user in the system. Selecting the right features is important because those features have a directed impact on the effectiveness of the prediction task. As mentioned earlier, only considering the common movements and social properties did not work well. Therefore, we explore the users' textual information and seek unique features of users captured in their own information and information need for followee classification.

By dealing with the observations prompted in the above examples, we extract features of user pair in three different but complementary aspects: 1) *Social Property* (*SP*), 2) *Geographical Property* (*GP*), and 3) *Textual Property* (*TP*). The features extracted from *Social Property*, corresponding to a given user pair, can be derived from the intersection among their followees and followers based on statistical analysis. To consider the factor of users' provided information, we extract the features from *Geographical Property* to capture the relevance between users' provided trips by a HITS-Based random walk model [3]2. To involve the factor of users' information need, we extract the features from *Textual Property* to capture the relevance between users' provided textual information and information needs by exploiting the representative keywords of their travelogues and comments of trips. To facilitate feature extraction from *Textual Property*, we propose a family of graph representations that capture the

user-keyword and location-keyword relationships from the users' textual information. We develop an algorithm to build a captures the information needs of each user by exploring above-mentioned two graphs. Accordingly, for each ordered user pair (*u, v*), we derive the probability to evaluate the closeness between the information provided by *v* and *u*'s information needs. This following probability is thus treated as a feature of *Textual Property*, along with the features derived from *Social Property* and *Geographical Property*, to feed the binary SVM in our *GTS-FR* model.

This research work has made a number of significant contributions, as summarized below:

- We propose to tackle the problem of user textual information mining in users' relations, which is a crucial prerequisite for effective followee recommendation in an asymmetrical LBSN.
- We propose *Geographic-Textual-Social Based Followee Recommendation* (*GTS-FR*), a new approach for users' similarity mining and followee recommendation on an asymmetrical LBSN. The problems and ideas in *GTS-FR* have not been explored previously in the research community.
- We formulate the problem of followee recommendation in an asymmetrical LBSN as the problem of binary class classification and propose *GTS-FR* to learn a SVM for each user to estimate possibilities of other users. In the proposed *GTS-FR*, we explore 1) *Social Property* (*SP*), 2) *Geographical Property* (*GP*), and 3) *Textual Property* (*TP*) by exploiting the LBSN data to extract descriptive features.
- We use a real dataset, which was crawled from *EveryTrail* [1], to evaluate the effectiveness of our proposed *GTS-FR* in a series of experiments. The results show *GTS-FR* delivers superior effectiveness over other recommendation strategies in terms of the popular measures precision, recall and F-measure.

The rest of this paper is organized as follows. We briefly review the related work in 2nd Section and provide our followee recommendation approach GTS-FR in 3rd Section. Finally, we present the evaluation result of our empirical performance study in 4th Section and discuss our conclusions and future work in 5th Section.

RELATED WORK

Actually, apart from friend-of-friend strategy, most existing friend recommendations on LBSN focus on dealing with users' similarity measurement for making recommendations. Many studies [5, 6, 8, 11] have proposed to discuss the problem of similarity measurement in the field of data mining. Trajectory similarity measurement [5] and user similarity measurement [6, 8, 11] are two hot topics in this problem. In [5], Lee *et al.* proposed a Partition-and-Group method to calculate the similarity between two trajectories. For all trajectories, they first find the characteristic points to form line segments and then apply three kinds of distance measures, i.e., perpendicular distance, parallel distance and angle distance, on these segments to group the trajectories. However, these distance measures are only applicable to geographic information and thus can not be used to measure user similarity based on semantic trajectories.

The main idea of trajectory-based user similarity measurement is to derive the user similarity by analyzing the movement behaviors of mobile users. In [11], Zheng *et al.* proposed a personalized friend and location recommendation system which is called *HGSM-based recommender*. To explore users' similarities, the system considers users' movement behaviors in various location granularities. Based on the definition of stay point which is the geographic region where mobile users usually stay for over a time threshold, the system discovers all of the stay points in trajectories and then employ a density-based clustering algorithm to organize these stay points as a hierarchical framework. Such cluster is named stay region (or stay location). As such, a personal hierarchical graph is formed for each user. For each level of hierarchical graph, a user's trajectory can be transformed as a sequence of stay regions. To measure the similarity of two users, some common sequences, named similar sequence, are discovered by matching their stay region sequences in each level of hierarchical graph. Then, for each stay region, the TFIDF value for a similar sequence is calculated, where TF value represents the minimum frequency of the two users accessed this stay region within the similar sequence, while the IDF value indicates the number of users who have visited this stay region. Finally, the similarity between two users is derived by the summation of the TFIDF values of all stay regions within the similar sequences. However, this approach treats every stay region in the similar sequence independently, i.e., without considering the sequential property of stay regions in the similar sequence. In [8], the *LBS-Alignment* method was proposed to calculate the similarity of two mobile users. The LBS-Alignment method calculates the similarity of two users by using the longest common sequence within their Mobile Sequential Patterns. By analyzing such longest common sequences, the ratio of common part in the Mobile Sequential Patterns are taken as the similarity. Although all these approaches have considered temporal information and location hierarchy, they do not take into account the semantics of locations.

GTS BASED FOLLOWEE RECOMMENDATION

The proposed *GTS-FR* approach is designed a two-phase algorithm, as shown in Figure 3, to address the problem of users' similarity mining for followee recommendation. The first phase deals with the feature extraction (lines 1 to 5), while the second phase explains the followee recommendation (lines 7 to 11). The task of feature extraction explores three aspects that are discussed in Introduction. For a user pair, we explore the *Social Property* (*SP*) as population features which abstract the aggregated number of followee-of-followee links of two users. On the other hand, we explore the *Geographical*

Input: Social Links Set L
Users' Trips T
Users' Textual Information I
Users U
Output: relation between each pair of users
1 *Phase 1. Feature Extraction*
2 Feature Set $F \leftarrow \varnothing$
3 $F \leftarrow F \cup SP(L, I)$
4 $F \leftarrow F \cup GP(T, I)$
5 $F \leftarrow F \cup TP(I)$
6
7 *Phase 2. Feature Extraction*
8 Training Set $T \leftarrow F \cup L$
9 Classifier $C \leftarrow$ SVM(T)
10 Classification Result $R \leftarrow C(U \times U)$
11 Return R

Figure 3. GTS-FR algorithm.

Property (*GP*) between two users to formulate descriptive features of a specific user pair. Moreover, to overcome the experience-limitation problem, *Textual Property* (*TP*) is considered as a feature to represent information needs of users in our recommendation model. The features derived from *Social Property*, *Geographical Property* and *Textual Property* are used to learn a SVM model for each user to classify whether other users could be followed in the phase of followee recommendation. For a user, other users are classified into followee and non-followee classes by the individual SVM model of the user. After checking all users, we obtain all qualified potential followees for the user under examination.

Features from Social Property

As discussed earlier, the traditional social-based friend recommendations could not work well. The reason is that the traditional social-based friend recommendations always make recommendation by friend-of-friend links. The concept of reformation by using friend-of-friend links is that if an user B is a friend of user A's friends, B is likely to be a friend of A. If we directly adopt such recommendation concept, we should modify it by using followee-of-followee link. In other words, such followee-of-followee recommendation strategy is based on the concept that if user X is followed by user Y's followees, X may be followed by Y. Take Figure 4 as an example, we may recommend user *b* to user *k* because user *k* follows user *j* and user *j* follows user *b*. However, recommending followee's followee can not reflect the relation of information need and offered information. In other words, the reason of user *k* following user *j* is different with the reason of user *j* following user *b*. We argue that the "transitivity" of followee-of-followee link should be considered.

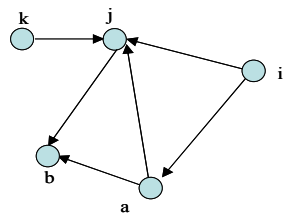


Figure 4. transitivity and followee-of-followee link

Definition 1. Transition-Setter. Given a followee-of-followee link, denoted ($u \rightarrow v$, w), is a directed path in an asymmetrical social network from user u to user v via user w. The middle user w is called *Transition-Setter*.

Accordingly, given a user-user order pair (u, v), here (u, v) $\neq$(v, u), the features extracted from *Social Property* could be generally formulated as (2).

$$SP(u,v) = \sum_{t \in T(u,v)} Transitivity(t) \quad (2)$$

where $T(u, v)$ indicates the set of Transition-Setters of all followee-of-followee links from u to v.

As mentioned above, we can significantly observe that measuring transitivity of two users' Transition-Setters is the key of *Social Property* features. Intuitively, population of common followees of an user and his followers could be utilized for measuring the transitivity of the user because their information needs are satisfied with the information offered by their common followees. As a result, different transitivity, naturally formed in aggregated relations of followers to followees, is embedded in the followers' following behaviors. In an asymmetrical social network data, the most important information is user's following behaviors among users for user transitivity measurement. In the following, we propose to extract two population features to depict users' Transition-Setter as below.

- **Transitivity by Links between Followees and Followers (LinkTran)** - As discussed above, some people will follow another people by cooperating of followee-of-followee link and Transition-Setters' transitivity. Based on this idea, the design idea of LinkTran focuses on the proportion of pairs of follower and followee are linked from follower to followee. Accordingly, we formulate the LinkTran of a Transition-Setter as (3).

$$LinkTran(i) = \frac{1}{|P(i)| \times |S(i)|} \times \sum_{\forall j \in P(i)} \sum_{\forall k \in S(i)} I(j,k) \quad (3)$$

where $P(i)$ indicates the set of followers of user i, $S(i)$ indicates the set of followees of user i, $I(j, k)$ is an indicator function which indicates whether user j follows user k. Take Figure 4 as an example. The followers of user j are user a user i, and user k. The followee of user j is user b. Thus, the LinkTran of user i and is (1+0+0)/(3×1) ≒ 0.33

- **Transitivity by Communications between Followees and Followers (CTran)** - We employ the χ^2 test for testing relation of texting behaviors of EveryTrail users and their followee. If the test shows significant, it means that the user always comments his followees' trips. Based on the

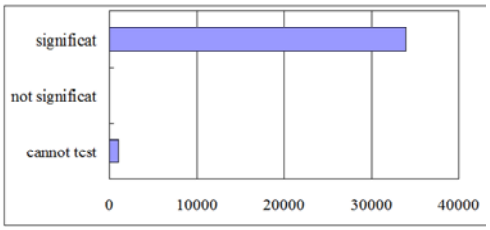


Figure 5. result of χ^2 test.

observations from the EveryTrail dataset, shown in Figure 5, we find most of users will comment their followees' trips. Hence the number of comments is a good index for measuring users' transitivity. Based on the observations, we replace the indicator function, i.e., $I(j, k)$, of formula (3) by following (4).

$$CTran(j,k)=\begin{cases}1-\dfrac{Comment(j,k)}{\max\limits_{f\in S(j)}\{Comment(j,f)\}} & \text{, if user } j \text{ follows user } k\\ 0 & \text{, otherwise}\end{cases} \quad (4)$$

where $Comment(j, k)$ indicates the number of comments offered by user j to user k and $S(j)$ indicates the set of followees of user j.

Features from Geographical Property

As mentioned above, the reason of a user follows other users in the asymmetrical LBSN is either the information need or making friends. To make a complete recommendation, users' interest should be considered because people always make friend who have similar interest. In EveryTrail website, there are two kinds of user-generated data could reflect their interest, i.e., trips and tags of trips, as shown in Figure 6. The trip is also called trajectory typically consists of a sequence of geographic points (represented as <latitude, longitude>). The trajectory could reflect the detail of user's activity. On the other hand, the tag of trajectory could reflect the high level concept of user's activity. Each trajectory just have only one tag, e.g., hiking, biking, etc.

Accordingly, given an ordered user pair (u, v), the features extracted form Geographical Property could be generally formulated as (7).

$$GP(u,v)=\frac{1}{|Tr(u)|\times|Tr(v)|}\times\sum_{\forall p\in Tr(u)}\sum_{\forall q\in Tr(v)}Similarity(p,q) \quad (7)$$

where $Tr(u)$ indicates the set of trajectories of user u.

We can significantly observe that measuring similarity of two trajectories is the key of Geographical Property features. Intuitively, the regions which user stays in could reflect the user's preference. As a result, for each user, we adopt the notion of *stay locations* [11] to represent the users' movement behavior as shown in Figure 7. To discover stay locations, we first detect the regions, called stay points, where a user stayed in, i.e., *s1* and *s2* in Figure 7. Then we cluster all detected stay points to form stay locations, i.e., *location2* and *location5* in Figure 7. As shown in Figure 7, the trajectory could be transformed as the sequence <*location2*, *location5*>. As the result, the similarity measurement could be modeled as the sequences matching problem.

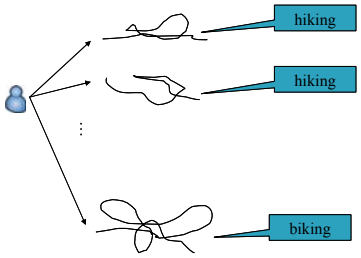


Figure 6. trips and tags of trips.

Given two sequences, we argue that they are more similar when they have more common parts. Thus, we use the *Longest Common Sequence* (*LCS*) of these each pair of sequences to represent their longest common part. For example, given a sequence $P = <A, B, C, D>$ and a pattern $Q = <A, D, C>$, their longest common sequence is $LCS(P, Q) = <A, C>$. Accordingly, we define the *participation ratio* of the common part to a pattern P as follows.

$$ratio(LCS(P,Q),P)=\frac{|LCS(P,Q)|}{|P|} \quad (8)$$

Intuitively, the tags of two sequences could reflect basic concepts of them. Therefore, the similarity of two sequences will be evaluated as 0 if their tags are different. Thus, we calculate the similarity of two sequences by averaging the participation ratios of their common part to them. Given sequences P and Q, a simple approach is to directly compute the average of the two ratios to P and Q, as shown in Equation (9). Thus, we call this approach *Equal Average (EA)*. On the other hand, as shown in Equation (10), we can compute the *Weighted Average (WA),* in proportion to the lengths of the two sequences. The argument is that a longer pattern provides more information about user behaviors than a shorter pattern. Therefore, the longer pattern gives more weight than the shorter one in measuring the similarity between two sequences.

$$Similarity_{EA}(P,Q)=I_T(P,Q)\times\frac{ratio(LCS(P,Q),P)+ratio(LCS(P,Q),Q)}{2} \quad (9)$$

$$Similarity_{WA}(P,Q)= I_T(P,Q)\times\frac{|P|\times ratio(LCS(P,Q),P)+|Q|\times ratio(LCS(P,Q),Q)}{|P|+|Q|} \quad (10)$$

where $I_T(P, Q)$ is an indicator function which indicates whether the tags of P and Q are the same. Note that we could extract two features from Geographical Property, namely *EA* and *WA*.

Features from Textual Property

As discussed earlier, we intend to exploit the users' information needs in LBSN for matching other users' provided information by a HITS-Based random walk model [3]. We believe that users comment other users' trip or write travelogue within their trips can represent their information needs. Therefore, we build a *User-Keyword* (*UK*) graph, which consists of users and keywords connected in accordance with the textual records. Let $t(u_i, w_j, l_s) \in TI$ denotes a textual record describing that user u_i has provided textual information which contains the keyword w_j and associate the location l_s, where *TI* indicates

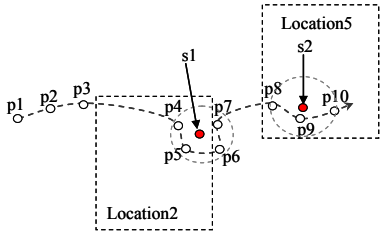


Figure 7. transitivity and followee-of-followee link

the collection of all textual records. Here, the keywords are extracted from all textual information with high TFIDF value. Definition 2 gives the formal definition of the *UK* graph.

Definition 2. User-Keyword (UK) Graph, denoted by $G_u(V_u,E_u)$, is an undirected bipartite graph (as illustrated in Figure 8(a)). Here $V_u = U \cup K$, where U and K are the sets of all users and keywords, respectively, and $E_u = \{e_{i,j} \mid t(u_i, w_j, \cdot) \in TI\}$, where $t(u_i, w_j, \cdot)$ denotes that user u_i has texted keyword w_j in some textual information. In this graph, each edge $e_{i,j} \in E_u$ is weighted by the number of keyword w_j has been texted by user u_i.

Given m users and n keywords, we build an $m \times n$ adjacency matrix M for *UK* Graph. Formally, $M = [c_{ij}]$, $0 \leqq i < m$; $0 \leqq j < n$, where c_{ij} represents how many times the i_{th} user has texted the j_{th} keyword. Formally, the random walk model applied to UK Graph can be described as follows:

$$\begin{aligned} x_{keyword}^{k+1} &= (\varepsilon M_{col}^{T} + (1-\varepsilon)\delta_1)x_{user}^{k} \\ x_{user}^{k+1} &= (\varepsilon M_{row} + (1-\varepsilon)\delta_2)x_{keyword}^{k+1} \end{aligned} \quad (11)$$

where k is the number of iterations, M_{col} is the column stochastic matrix of M (M_{col} is computed by normalizing each column in M), M_{row} is the row stochastic matrix of M (M_{row} is computed by normalizing each row in M), δ_1 is a matrix with all elements equal to $1/m$, δ_2 is a matrix with all elements equal to $1/n$, and ε is the "teleport probability," which represents the probability of a random surfer teleporting from a keyword node to a user node (respectively from a user node to a keyword node) instead of following the links in *UK* Graph.

As the above-mentioned random walk model, the users' relevance can be obtained. However, such random walk model do not consider the relationship among keywords. We argue that the similar keywords could represent similar information needs. Intuitively, similar keywords could be texted with the same locations. Therefore, we build a *Location-Keyword* (*LK*) graph, where the locations are the same as stay locations which is extracted in the *Geographical Property* feature extraction step. Definition 3 gives the formal definition of the *LK* graph.

Definition 3. Location-Keyword (LK) Graph, denoted by $G_l(V_l, E_l)$, is an undirected bipartite graph (as illustrated in Figure 7(b)). Here $V_l = L \cup K$, where L and K indicate the sets of all locations and keywords, respectively, and $E_l = \{e_{j,s} \mid t(\cdot, w_j, l_s) \in TI\}$, where $t(\cdot, w_j, l_s)$ denotes that location l_s has been texted with keyword w_j in comments or travelogues. In this graph, each edge $e_{j,s} \in E_l$ is weighted by the proportion of keyword w_j that has been texted in the comments or travelogues of user location l_s.

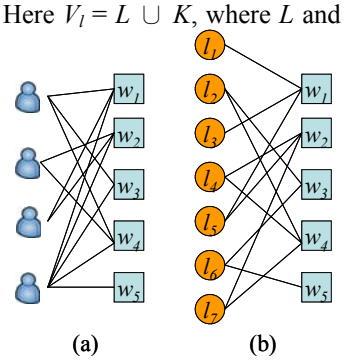


Figure 8. trips and tags of trips.

Given r locations and n keywords, we build an $n \times r$ adjacency matrix N for *LK* Graph. Formally, $N = [v_{ij}]$, $0 \leqq i < n$; $0 \leqq j < r$, where v_{ij} represents how many times the i_{th} keyword appears in the textual information associated with the j_{th} location. Formally, the random walk model applied to *LK* Graph can be described as follows:

$$\begin{aligned} x_{keyword}^{k+1} &= (\varepsilon M_{col}^{T} + (1-\varepsilon)\delta_1)x_{user}^{k} \\ y_{location}^{k+1} &= (\varepsilon N_{col}^{T} + (1-\varepsilon)\delta_2)x_{keyword}^{k+1} \\ y_{keyword}^{k+1} &= (\varepsilon N_{row} + (1-\varepsilon)\delta_3)y_{location}^{k+1} \\ x_{user}^{k+1} &= (\varepsilon M_{row} + (1-\varepsilon)\delta_4)y_{keyword}^{k+1} \end{aligned} \quad (12)$$

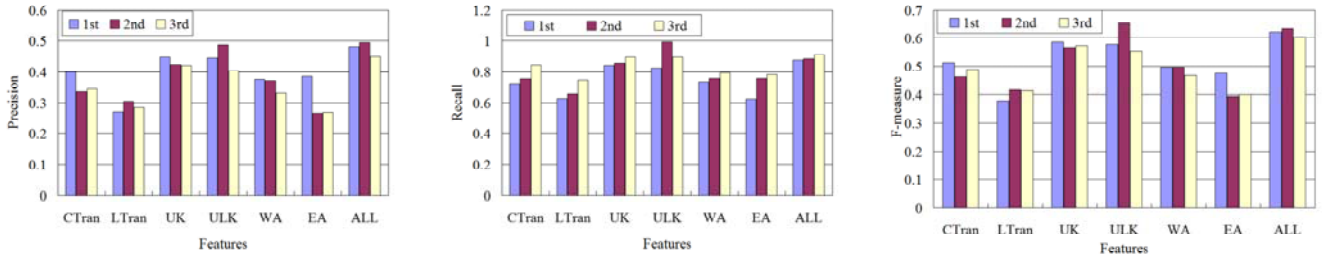
where k is the number of iterations, M_{col}, M_{row}, ε, δ_1 and δ_2 are the same as the random walk model applied to *UK* Graph, i.e., formula (11). Similarly, N_{col} is the column stochastic matrix of N (N_{col} is computed by normalizing each column in N), N_{row} is the row stochastic matrix of N (N_{row} is computed by normalizing each row in N), δ_3 is a matrix with all elements equal to $1/n$, δ_4 is a matrix with all elements equal to $1/r$. Note that we could extract two features from *Textual Property*, namely *UK* and *LK*.

Followee Recommendation

After the phase of feature extraction, features derived from all of social, geographical and textual properties are used as inputs for the followee recommendation phase to learn a classification model for each individual user. We choose SVM as the classifier because it has shown excellent performance in similar tasks [2, 4, 6]. The reason why we select SVM as our classifier is that SVM is hard to be effected by class-imbalanced problem. In our approach, for each user, all of other users are used for his SVM training, i.e., an instance followed by the user under examination is considered as a positive example, while users without being followe by the user serve as negative examples. For instance, users followed by *user 1* are positive examples for a classifier for *user 1*, but negative examples for a classifier for *user 2*.

EXPERIMENTS

In this section, we conduct a series of experiments to evaluate the performance for the proposed GTS-FR using EveryTrail dataset . All the experiments are implemented in Java JDK 1.6 on an Intel Core i7-2600 CPU 3.40 GHz machine with 7GB of memory running Microsoft Windows win7. We first describe the data preparation on the EveryTrail dataset and then introduce the evaluation methodology. Finally, we show our experimental results for following discussions.



(a) (b) (c)

Figure 9. Comparison of Various Features

EveryTrail Dataset

EveryTrail is a trip-sharing and social networking website on which users can upload, share and find trips. On EveryTrail, users can upload GPS logs and write travelogues and comments within a trip. Users also can label a tag on a trip. While the EveryTrail website provides the public API to let other applications integrate with their service, some functionality in the API is broken. For this reason, we mainly use the API and with crawling web pages as support to get all the data we need. We extract the data from 12/2011 to 3/2012, each month is a time period. We got 35,153 users and 4 snapshots. The data description of each snapshot is given in Table 1.

Snapshot	1st	2nd	3rd	4th
# of trips	116179	145,662	193,331	196,949
# of comments	337,519	293,453	315,585	379,020
# of links	700,103	777,738	1,056,077	1,139,832

Table 1. Data Description of Each Snapshot

All of the data is divided into the training data and the testing data. The first, second and third snapshots are formed as the training data, and the remaining snapshots are formed as the testing data. For example, if we select the first snapshot as training data, the testing data will be extract second snapshot. Since the problem we address is followee recommendation, we only care about the following links which are not created in training data. Thus, the testing data will be formed by ordered user pairs who do not link in training data.

Evaluation Methodology

The follows are the main measurements for the experimental evaluations. The Precision, Recall and F-measure are defined as Equations (13), (14) and (15), where p^+ and p^- indicate the number of correct recommendations and incorrect recommendations, respectively, and R indicates the total number of links in the testing data.

$$\text{Precision} = \frac{p^+}{p^+ + p^-} \tag{13}$$

$$\text{Recall} = \frac{p^+}{R} \tag{14}$$

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

We divide the experiment into two parts: 1) Comparison of Various Factors or Features (i.e., Internal Experiments) and 2) Comparison of Existing Recommenders (i.e., External Experiments). For the comparison of various features, we first compare the performance of our proposed Social Property, Geographical Property and Textual Property. Then, we compare the effectiveness of all of features. For the comparison of existing recommenders, we compare the effectiveness of GTS-FR with HGSM-based recommender [11] and followee-of-followee strategy in terms of Precision, Recall, and F-measure.

Comparison of Various Features

This experiment evaluates effectiveness of each factor in the proposed GTS-FR in terms of Precision, Recall, and F-measure. Figure 9 shows that, on average, all of Precision, Recall, and F-measure value of GTS-FR, under different features, i.e., CTran, LTran, UK, ULK, WA and EA, respectively. We observe that all of Precision, Recall, and F-measure of UK and ULK are better than those of other four features. The result shows that the features extracted from Textual Property are more important than those extracted from other features. If we focus on comparison of UK and ULK, overall, the UK is more stable but ULK could achieve highest recall. This is because the locations we detect are always changed. Sometimes, the change might benefit effectiveness but not usually. Moreover, we also can observe that the values of recall are always greater than the values of Precision. The reason is that social link is created slowly because the Everytrail website is established a long time. There are many links we recommend are created in the future but not in the next snapshot. Accordingly, we analyze the incorrect recommendations by the first snapshot model, which is tested by the second snapshot, whether they will become correct in the further snapshot. As shown in Figure 11, we can find that most incorrect recommendations become correct in the further

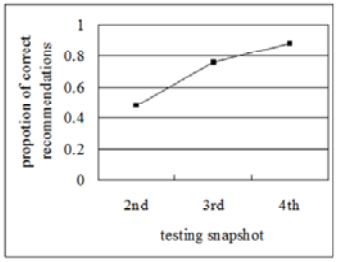
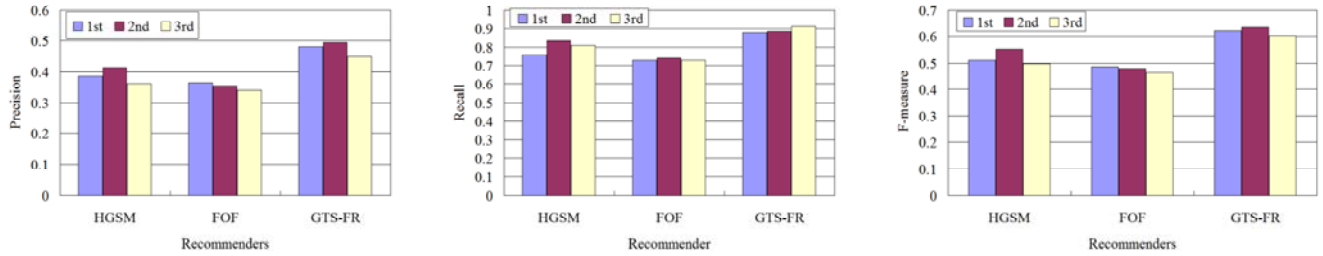


Figure 11. Analysis of incorrect recommendations.



(a) (b) (c)

Figure 12. Comparison of Various Recommenders

snapshot.

Comparison with Existing Recommenders

This experiment evaluates the effectiveness of our proposed GTS-FR comparing HGSM-based recommender and followee-of-followee strategy (FOF) in terms of Precision, Recall, and F-measure. HGSM-based recommender relays on the similarity of users' uploaded trajectory. It is similar to our proposed factor Geographical Property but more effective. followee-of-followee strategy (FOF) is widely used for followee recommendation in many existing LBSN websites. It is similar to our proposed factor Social Property. Figure 12 shows GTS-FR outperforms HGSM-based recommender and followee-of-followee in terms of Precision, Recall, and F-measure. The reason is that we consider users' relationship in the factor of users' information need, reflected by textual information, while other methods do not.

CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a novel approach named *Geographic-Textual-Social Based Followee Recommendation* (*GTS-FR*) for recommendation of interesting followees by mining users' information needs. Meanwhile, we have tackled the problem of user texting behaviors mining in information need discovering, which is a crucial prerequisite for effective recommendation of followees in a LBSN. The core task of followee recommendation in a LBSN can be transformed to the problem of the problem of binary classification. We evaluate the possibility of each ordered user pair by learning an SVM model. In the proposed *GTS-FR*, we have explored i) *Social Property* (*SF*), ii) *Geographical Property* (*GP*) and iii) *Textual Property* (*TP*) by exploiting the LBSN data to extract descriptive features. To our best knowledge, this is the first work on followee recommendation that consider social property, geographical property and textual property in LBSN data, simultaneously. Through a series of experiments by the real dataset obtained from EverTrail, we have validated our proposed *GTS-FR* and shown that *GTS-FR* has excellent effectiveness under various conditions. As for the future work, we plan to design more advanced link prediction strategies to further enhance the quality of followee recommendation for location-based social networks.

ACKNOWLEDGMENTS

This research was supported by National Science Council, Taiwan, R.O.C. under grant no. NSC101-2221-E-006-255-MY3 and NSC100-2218-E-006-017.

REFERENCES

1. EveryTrail: http://www.everytrail.com/.
2. Boutell, M. R., Luo, J., Shen, X. and Brown, C. M. Learning multi-label scene classification. Pattern Recognition, 37(9):1757–1771, 2004.
3. Cao, X., Cong, G. and Jensen, C. S. Mining significant semantic locations from GPS data, Proceedings of the VLDB Endowment, v.3 n.1-2, September 2010
4. Clare, A. and King, R. D. Knowledge Discovery in Multi-label Phenotype Data. In European Conference on PKDD, pages 42–53, 2001
5. Lee, J.-G., Han, J. and Whang, K.-Y. Trajectory Clustering: A Partition-and-Group Framework. In Proceedings of ACM SIGMOD, pp. 593-604, Jun. 2007.
6. Li, T. and Ogihara, M. Detecting emotion in music. In International Society for Music Information Retrieval Conference (ISMIR), 2003.
7. Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W. and Ma, W.-Y. Mining User Similarity Based on Location History. In Proceedings of ACM GIS, Irvine, CA, USA, Nov. 2008.
8. Lu, E. H.-C. and Tseng., V. S. Mining Cluster-Based Mobile Sequential Patterns in Location-Based Service Environments. In Proceedings of IEEE MDM, May. 2009.
9. Ye, M., Yin, P. and Lee, W.-C. Location Recommendation for location-based Social Network. In Proceedings of GIS, pages 458-461, 2010.
10. Ying, J. J.-C., Lu, E. H.-C., Lee, W.-C., Weng, T.-C. and Tseng, V. S. Mining User Similarity from Semantic Trajectories. In Proceedings of LBSN' 10, San Jose, California, USA, November 2, 2010.
11. Zheng, Y., Zhang, L. and Xie, X. Recommending friends and locations based on individual location history. ACM Transaction on the Web, 2011
12. Zheng, Y. Location-based social networks: Users. Computing with Spatial Trajectories, Yu Zheng and Xiaofang Zhou, Eds. Springer, 2011.

Geo-activity Recommendations by using Improved Feature Combination

Masoud Sattari
Middle East Technical University
Ankara, Turkey
e176326@ceng.metu.edu.tr

Ismail H. Toroslu
Middle East Technical University
Ankara, Turkey
toroslu@ceng.metu.edu.tr

Pinar Senkul
Middle East Technical University
Ankara, Turkey
senkul@ceng.metu.edu.tr

Murat Manguoglu
Middle East Technical University
Ankara, Turkey
manguoglu@ceng.metu.edu.tr

Panagiotis Symeonidis
Aristotle University
Thessaloniki, Greece
symeon@csd.auth.gr

Yannis Manolopoulos
Aristotle University
Thessaloniki, Greece
manolopo@csd.auth.gr

ABSTRACT
In this paper, we propose a new model to integrate additional data, which is obtained from geospatial resources other than original data set in order to improve Location/Activity recommendations. The data set that is used in this work is a GPS trajectory of some users, which is gathered over 2 years. In order to have more accurate predictions and recommendations, we present a model that injects additional information to the main data set and we aim to apply a mathematical method on the merged data. On the merged data set, singular value decomposition technique is applied to extract latent relations. Several tests have been conducted, and the results of our proposed method are compared with a similar work for the same data set.

Author Keywords Geospatial Recommendation, Matrix Factorization, Feature Combination

ACM Classification Keywords H.2.8 [Database Management] Database Applications – data mining

General Terms Algorithms, Experimentation, Performance

INTRODUCTION
With booming technology of smart mobile phones and satellite-assisted positioning systems, new demands for applications in this field are emerging. One of these requirements that most of the users are interested in is activity recommendation based on location (GPS) data, which is specially used to guide tourists and unfamiliar individuals in tourist-attracting cities. Therefore, location-based social networks (LBSN) and location-based recommendations have emerged as interesting research topics involving several dimensions [9, 10, 11]. In this work, we propose a method to improve location-based recommendation by injecting additional data into the original data set. The additional data comes from an external resource into the geospatial activity-location recommendation system.

Our work has actually been inspired from a similar earlier work of [8]. We also use the same data set, including three matrices, namely location-activity matrix, location-feature matrix and activity-activity matrix. The former matrix implies preference ratings of people on a specific activity in a specific location. Its entries actually correspond to the frequency of performing an activity in that location for all users. The second one contains available features of locations and finally, the last one represents relationships among different activities. As expected, the first data set, namely location-activity matrix, is very sparse and we want to determine the values of its unknown entries by utilizing the information in the other two matrices. Generally speaking, merging a data from one resource (Activity-Activity and Location-Feature) with the data from another resource (Location-Activity) is described as *Feature Combination* [1, 7] in the literature. Both [8] and this work apply feature combination to combine these three matrices and construct an integrated matrix. We apply Singular Value Decomposition (SVD) to uncover the latent relations within data. Basic difference and contribution of our work lies in the way integrated matrix is used for prediction and the application of SVD. At the end, we show the effectiveness of our approach by comparing it with the matrix completion approach of [8]. Our experiments show that, our method has higher prediction accuracy than the one in [8].

The rest of the paper has been organized as follows. In Section 2, we discuss related work and consider pros and cons of them. Section 3 explains the data model and we introduce our method in Section 4. Evaluation methods and results of our experiments are available in Section 5. Finally, in Section 6 we have conclusion part of the paper.

RELATED WORK
Nowadays, most of the recommender systems are used in e-commerce to help individuals in decision making [4]. Since

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

social networks like Facebook[1] and Google+[2] have attracted millions of people, several algorithms have been designed to recommend friendship requests and advertisements based on the geographical position of users [2]. Different recommender systems techniques have been proposed in literature and researchers have tried to combine them to build better hybrid models that make use of these techniques [7]. Also, there are applications that track people and get their feedback to build a GPS based recommendation [8].

Some works [9] add another attribute like *user* and model the data with a 3 dimensional tensor in order to have more targeted recommendations using collaborative filtering techniques. Beside this, they use a model-based method that benefits from machine learning techniques, to predict missing values in mentioned tensor. Similarly, the work in [10] presents a mobile recommendation system that, in addition to applying tensor factorization to whole data set, incorporates 2 other algorithms to predict missing values using partial data set.

The aim of our work, as well as the work of [8], can be described as a recommender system that recommends activities for a location. In [8], this is handled by completing the whole activity-location matrix. Thus, when there is a demand for making a recommendation, easily the entry corresponding to that recommendation request can be reached in the completed matrix and the recommendation can be made. However, if that matrix is very large this approach may not be feasible due to two reasons: even with a very effective method, matrix completion may take a lot of time, and there could be a need for a very large storage to store the result. Specially, if the recommendation requests correspond to a small portion of the whole matrix and they sparsely arrive, then responding to those requests on the fly may be a better approach.

In [8], the model that is used to predict unknown values is called *Collective Matrix Factorization*, which was proposed by Singh [6]. Collective Matrix Factorization begins with construction of an objective function. Then, this function is converted to an optimization problem, whereas this task is done iteratively by a numerical method that is known as *Gradient Descent* [5]. Note that this method finds local minima and does not guarantee to find the global minima. Practically, the more local minima are near to global minima, the better the model estimates the unknown values.

Our work aims to tackle the problem of generating recommendations when they are needed. Therefore, dimensionality reduction techniques appear as appropriate approach. In order to be able to deal with huge matrices, we use Singular Value Decomposition (SVD) [3] to generate low rank matrices, while injecting the additional information coming from two other data resources into the main activity-location matrix. Therefore, the main contribution of our work is combining more than one matrix into a single structure. Then, to make a recommendation we use only the main data part of the low rank approximation matrices that have been generated with SVD.

DATA SET CHARACTERISTICS

The data set that is used in this paper is gathered by Microsoft Research Asia. Pre-processed data set is available online and can be downloaded from Microsoft Research Asia website[3]. The data set is collected from a web-based application over 2.5 years so that, each user is equipped with a GPS installed tool (GPS Navigator, smart phone) during visiting Beijing city in China. For each location that is visited, users may insert comments about that place and possible 5 activities that are done in that location (food and drink, shopping, movie and shows, sports and exercise, tourism and amusement) thus, these comments with location info can be used to create a matrix called Location-Activity matrix, whose rows are locations and columns are activities. This matrix consists of 167 locations and 5 activities that each entry of it denotes the frequency of performing an activity for all users in that location.

To make it more informative, location-feature data extracted from *Point of Interest (POI)* database, which is based on city's yellow pages, is also added to our model. Usually in big cities for any given location area in the city this database gives the type and number of activities like cinemas, restaurants, shopping centers, sport complexes and so on. Gathered data can be modeled as a Location-Feature matrix whose entries are nonnegative integer values that for a given location from available 167 locations shows the frequency of each 13 available features and vice versa [8].

As explained in [8], there is a statistical relation between activities. For example if a user goes to cinema s/he may go to restaurant to eat food too. This is the kind of relationship that is aimed to be captured as Activity Correlation. This information is also available in this data set as a 5-by-5 matrix, which is named as Activity-Activity matrix whose entries are real values in interval [1,-1] and show the correlation between activities represented as the rows and columns.

The aim is mainly to make activity recommendation to the users based on their spatial location. Since, the Location-Activity matrix is very sparse, it makes sense to use the additional information captured in Location-Feature and Activity-Activity matrices in order to make more accurate activity recommendations. That was the main idea behind the work of [8]. Simply, the aim is to make use of Location-

[1] https://www.facebook.com

[2] https://plus.google.com

[3] http://research.microsoft.com/pubs/143146/aaai10.uclaf.data.zip

Feature and Activity-Activity matrices to predict the missing entries of the Location-Activity matrix.

OVERVIEW OF THE PROPOSED METHOD

In this section we describe our proposed method to determine the values of unknown entries of the Location-Activity sparse matrix. In the following sub-sections, we explain the procedure of the merging matrices, construction of the low-rank matrices and Activity-Location recommendation routine. We also describe the whole process through an example thoroughly.

Feature Combination and Low-Rank Matrix Construction

As discussed in Section 3, we have merged main matrix *X* (Location-Activity) with two additional matrices *Y* (Location-Feature) and *Z* (Activity-Activity) to construct an integrated matrix *T*. In order to preserve the structure of a matrix we have injected zero values in the rest of the null entries of *T*.

$$T_{(l+a)\times(f+a)} = \begin{bmatrix} Y_{l\times f} & X_{l\times a} \\ 0_{a\times f} & Z_{a\times a} \end{bmatrix} \quad (1)$$

Then, we simply apply a well-known technique that is *Singular Value Decomposition (SVD)*, to reveal the latent semantic indexing of data. The idea in SVD is to decompose a matrix T into three matrices *U*, S and *V* such that, *U* is left singular matrix, *V* is the right singular matrix and finally *S* contains singular values.

$$T_{r\times s} = U_{r\times r}\ S_{r\times s}\ V^T_{s\times s} \quad (2)$$

In general, SVD is used for dimensionality reduction so that, with selecting the top *k* values of *S* and *k* columns of *U* and V^T and by multiplying them respectively, we can represent matrix *T* with reduced matrix $\mathbb{T}$ which has rank of k ($k \leq Rank(T)$).

$$\mathbb{T}_k = U_{r\times k}\ \ S_{k\times k}\ \ V^T_{k\times s} \quad (3)$$

For simplicity, in the rest of paper we show the reduced rank components of *T* with U_k, S_k and V_k^T.

Activity Recommendation

Decomposing original matrix *T* reveals an interesting characteristic of U_k and V_k^T. Actually, using U_k for a given activity we can easily find similar locations that this activity is also done and using V_k^T, for a given location we can find similar activities that are done in that location. Moreover, combining these two may even lead to better results. In this step, to predict a frequency rating for a given location *i* and activity *j*, $a_{i,j}$ we search for similar rows of *i* in U_k and also for similar columns of j in V_k^T. In order to have an accurate estimate for frequency rating, instead of selecting one similar neighborhood, we pick the most similar *m* rows in U_k and also the most similar *n* columns in V_k^T. The following equations define these operations.

$$M\_Rating_{sim\ row} = Mean(\textstyle\sum_s Rating(a_{s,j})) \quad (4)$$

$$s = top\ m\ similar\ rows\ in\ U_k$$

$$M\_Rating_{sim\ col} = Mean(\textstyle\sum_s Rating(a_{i,s})) \quad (5)$$

$$s = top\ n\ similar\ columns\ in\ V_k^T$$

After that, the average of $M\_Rating_{sim\ row}$ and $M\_Rating_{sim\ col}$ is used as the predicted rating value for an activity at a location.

Notice that, since the row count of U_k is more than the number of actual locations (due to merge operation) to find similar locations in U_k we trim it so that, its rows corresponds to locations (*l*) only. This is done by selecting the first *l* rows of U_k for the similarity search. Similarly, to find similar activities in V_k^T we trim it so that, its columns corresponds to the activities (*a*) only. It is performed by selecting last *a* columns of V_k^T for the similarity search. Thus, equations (4) and (5) are applied to only these rows and columns.

In order to see influence of similarity metrics, we have applied both Euclidean distance and Cosine similarity to get similarity matrices. On the basis of our experiments, we have observed that, Cosine similarity yields more accurate results than Euclidean distance. Euclidean distance is defined as follows where, $\|\boldsymbol{x}\|$ denotes the Euclidean norm of $\boldsymbol{x}$.

$$sim(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|\ \|\boldsymbol{b}\|} \quad (6)$$

Example

In this section we demonstrate a sample example of our method with a simplified data set. As we have already seen, $X_{5\times3}$ is Location-Activity matrix, $Y_{5\times4}$ is Location-Feature matrix and $Z_{3\times3}$ is Activity-Activity correlation matrix.

$$X = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 17 & 17 \\ 0 & 53 & 53 \\ 76 & 4 & 82 \\ 2 & 0 & 0 \end{bmatrix} \quad (7)$$

$$Y = \begin{bmatrix} 0.0037 & 0.0037 & 0.0038 & 0 \\ 0 & 0 & 0 & 0 \\ 0.0082 & 0.0082 & 0.0165 & 0.0131 \\ 0.0100 & 0 & 0.0100 & 0.0318 \\ 0 & 0.0757 & 0 & 0 \end{bmatrix} \quad (8)$$

$$Z = \begin{bmatrix} 1 & 0.0650 & 0.0008 \\ 0.0650 & 1 & 0.0017 \\ 0.0008 & 0.0017 & 1 \end{bmatrix} \quad (9)$$

At the beginning, we combine *X*, *Y* and *Z* matrices based on (1) to construct the integrated matrix *T*. In order to illustrate how our method works and how accurately it determines some missing values, we select 3 nonzero entries from *X* randomly and change their values to 0. The selected entries are x_{11}, x_{22} and x_{33} which are bolded in *T*.

$$T = \begin{bmatrix} 0.0037 & 0.0037 & 0.0038 & 0 & \mathbf{0} & 3 & 1 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{0} & 17 \\ 0.0082 & 0.0082 & 0.0165 & 0.0131 & 0 & 53 & \mathbf{0} \\ 0.0100 & 0 & 0.0100 & 0.0318 & 76 & 4 & 82 \\ 0 & 0.0757 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0.0650 & 0.0008 \\ 0 & 0 & 0 & 0 & 0.0650 & 1 & 0.0017 \\ 0 & 0 & 0 & 0 & 0.0008 & 0.0017 & 1 \end{bmatrix} \quad (10)$$

According to (2), SVD method is applied to T which yields matrices $U_{8\times8}$, $S_{8\times7}$ and $V^T_{7\times7}$ as follow:

$$U = \begin{bmatrix} \mathbf{-0.025} & \mathbf{-0.010} & -0.999 & 0.035 & 0.001 & -0.016 & 0.005 & 0 \\ \mathbf{-0.279} & \mathbf{-0.124} & 0.008 & -0.031 & 0.952 & -0.008 & 0.008 & 0 \\ \mathbf{-0.870} & \mathbf{-0.385} & 0.026 & 0.010 & -0.305 & 0.003 & -0.003 & 0 \\ \mathbf{-0.405} & \mathbf{0.914} & 0 & -0.026 & -0.001 & 0 & 0.001 & 0 \\ \mathbf{-0.010} & \mathbf{0.024} & 0.035 & 0.999 & 0.033 & 0.001 & 0 & 0 \\ -0.001 & 0.001 & 0 & 0 & 0.006 & -0.302 & -0.953 & -0.001 \\ 0 & 0 & -0.017 & 0 & 0.010 & 0.953 & -0.302 & 0.004 \\ 0 & 0 & 0 & 0 & 0 & -0.004 & 0 & 1 \end{bmatrix} \quad (11)$$

$$S = \begin{bmatrix} \mathbf{79.36} & \mathbf{0} & 0 & 0 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{75.48} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.12 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0076 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.00069 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0000058 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0000017 \end{bmatrix} \quad (12)$$

$$V^T = \begin{bmatrix} -0.00001 & 0.00001 & -0.0002 & -0.001 & \mathbf{-0.36} & \mathbf{-0.63} & \mathbf{0.69} \\ -0.00001 & -0.000002 & -0.00004 & 1 & \mathbf{-0.01} & \mathbf{0.01} & \mathbf{0.004} \\ -0.00002 & 0.000004 & -0.0002 & 0.0005 & -0.73 & -0.27 & -0.63 \\ -0.00003 & 0.00003 & 0.00002 & -0.009 & -0.58 & 0.73 & 0.36 \\ -0.39 & 0.92 & 0.04 & -0.0000002 & 0.00003 & -0.00002 & -0.00002 \\ -0.66 & -0.25 & -0.71 & -0.00004 & 0.0001 & 0.0001 & -0.000008 \\ -0.64 & -0.30 & 0.71 & 0.00002 & -0.0001 & -0.0001 & 0.00001 \end{bmatrix} \quad (13)$$

Since we are interested in specific part of U and V^T, we trim them so that, they meet the same indices of X in T. In order to reduce the integrated matrix to rank 2, first 2 columns of U, S and first 2 rows of V^T are selected as shown in (14), (15) and (16) correspondingly.

$$U_k = \begin{bmatrix} -0.025 & -0.01 \\ -0.279 & -0.124 \\ -0.87 & -0.385 \\ -0.405 & 0.914 \\ -0.01 & 0.024 \end{bmatrix} \quad (14)$$

$$S_k = \begin{bmatrix} 79.36 & 0 \\ 0 & 75.48 \end{bmatrix} \quad (15)$$

$$V_k^T = \begin{bmatrix} -0.36 & -0.63 & 0.69 \\ -0.01 & 0.01 & 0.004 \end{bmatrix} \quad (16)$$

To predict the value of x_{ij} in X we search for similar rows of i in U_k and similar columns of j in V_k^T. Remember that to compute similarity matrix, we made use of Cosine similarity between vectors. Related similarity matrices are presented in (17) and (18).

$$Sim(U) = \begin{bmatrix} 1 & 1 & 0.9352 & 0.6974 & 0.9997 \\ 1 & 1 & 0.9356 & 0.6983 & 0.9997 \\ 0.9352 & 0.9356 & 1 & 0.906 & 0.9441 \\ 0.6974 & 0.6983 & 0.906 & 1 & 0.7158 \\ 0.9997 & 0.9997 & 0.9441 & 0.7158 & 1 \end{bmatrix} \quad (17)$$

$$Sim(V^T) = \begin{bmatrix} 1 & 0.998 & 0.511 \\ 0.998 & 1 & 0.461 \\ 0.511 & 0.461 & 1 \end{bmatrix} \quad (18)$$

Using these similarity matrices we compute *Sim_index(U)* and *Sim_index(V^T)*. Each row in *Sim_index (U)* shows the index of most similar rows in descending order from left to right. Similarly, each column in *Sim_index(V^T)* shows the index of most similar columns in descending order from top to down.

$$Sim\_index(U) = \begin{bmatrix} 2 & 5 & 3 & 4 & 1 \\ 1 & 5 & 3 & 4 & 2 \\ 5 & 2 & 1 & 4 & 3 \\ 3 & 5 & 2 & 1 & 4 \\ 2 & 1 & 3 & 4 & 5 \end{bmatrix} \quad (19)$$

$$Sim\_index(V^T) = \begin{bmatrix} 2 & 1 & 1 \\ 3 & 3 & 2 \\ 1 & 2 & 3 \end{bmatrix} \quad (20)$$

As a final step, we are to predict the value of given x_{ij} from Location-Activity matrix X. We find the mean value of top m (in this example m is chosen to 3) similar nonzero rows of i and mean value of top n (in this example n is chosen to 1) similar nonzero columns of j incorporating (19) and (20). The estimated value of x_{ij} is mean value of these 2 terms. Prediction steps for x_{11}, x_{22} and x_{33} are as follow.

Top 3 similar rows of row 1 are 2, 5 and 3 with corresponding values of 0, 2 and 0 in column 1. Since values of rows 2 and 3 are zero we put them aside and select next similar rows which in this case is row 4 only.

$$Mean\ (1, 76) = 38.5 \quad (21)$$

According to first column of (18), column 2 is the most similar column to column 1 with value of 3 in row 1. Thus, the predicted value for x_{11} is mean value of this value and the value that is calculated in (21).

$$Mean\ (38.5, 3) = 19.25 \quad (22)$$

In x_{22}, top 3 nonzero similar rows of row 2 are 1, 3 and 4 with values of 3, 53 and 4 in column 2.

$$Mean\ (3, 53, 4) = 20 \quad (23)$$

Also, column3 is the most similar nonzero column of column 2 with value of 17 in row 2.

$$Mean\ (20, 17) = 18.5 \quad (24)$$

Procedure for x_{33} is same as the previous entries thus, we just show the values.

$$Mean\ (17, 1, 82) = 33.33 \quad (25)$$

$$Mean\ (33.33, 53) = 43.16 \quad (26)$$

In order to show better comparison of predicted values with original values and similar work in [8], we organized them at Table 1.

	Predicted	Work in [8]	Original
x_{11}	19.25	10.4205	1
x_{22}	18.5	15.3352	17
x_{33}	43.16	7.6185	53

Table 1. Predicted value vs. work in [8] and original value

EXPERIMENTS

In this section we explain the evaluation method and afterwards, present experimental results of our work. Finally we compare the results with similar works.

Evaluation Method

In order to measure the accuracy of our approach, well-known *k*-fold cross-validation method is used to partition the whole data set into a training data set and a validation data set. To be able to apply this validation technique we have assumed that our data set is accurate. Then, we have set $(1/k)^{th}$ of nonzero entries of the Location-Activity matrix to zero. Afterwards, we have applied our approach on the data set to predict the values of those entries. The *k* results from the folds then can be averaged (or otherwise combined) to produce a single estimation. In order to compare the results numerically, we have used *Root Mean Squared Error (RMSE)* and *Mean Absolute Error (MAE)* which both measure difference between observed values (original values) and estimated values.

$$MAE(O,E) = \frac{1}{n}\sum_{i=1}^{n}|o_i - e_i| \quad (27)$$

$$RMSE(O,E) = \sqrt{\frac{\sum_{i=1}^{n}(o_i - e_i)^2}{n}} \quad (28)$$

Where, O and E are observed value and estimated value correspondingly. We have compared our results with a state-of-the-art work that is studied by Zheng et al. in [8].

Experimental Results

In this section we show the experimental results obtained from the proposed model for prediction without applying abstraction method and with abstraction method.

Evaluation without Abstraction

As we discussed in previous subsection, in order to prepare training and validation data, we have used *k*-fold cross-validation. As it is used usually, we put $k = 10$ that is, in each fold of execution, we select 10% of nonzero entries in location- activity matrix randomly and put them as validation set, remaining part is the training set and it is used to construct the prediction model.

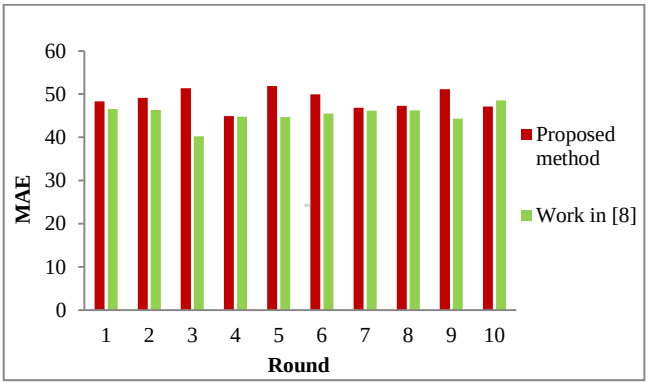


Figure 1. MAE values for proposed work vs. work in [8]

As discussed in Section 3, by using additional data, we implement the proposed method to predict the rating value of all entries in location-activity matrix. Since for validation data we have both observed value and predicted value, we can calculate *RMSE* and *MAE* for this fold. Therefore, in each loop we acquire a value for *RMSE* and *MAE*. At the end of last loop we sum up results for all folds and calculate the mean value of corresponding error terms. Obviously, all steps of 10-fold cross-validation were also applied to recommendation method in [8]. Each column in Figure 1 shows the mean value of *MAE* for 10 folds. In order to have a comprehensive view of results we run the application for 10 times and put the mean value of *MAE* in a new column.

Figure 2 shows the same comparison of RMSE for our proposed method and work in [8]. Like MAE, we run the application for 10 times that each column shows mean value of RMSE for 10 folds.

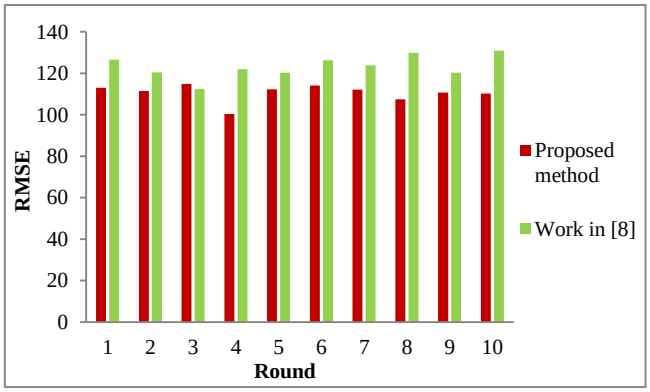


Figure 2. RMSE values for proposed work vs. work in [8]

Evaluation with Abstraction

As mentioned in [8], there are always places that are rated more than other places hence, they tend to have an abnormally greater rating values comparing to other places that are visited frequently but, does not receive much ratings. Beside this, comment adding is an optional case that users are requested to do it and they usually do not provide so many useful comments. In this data set which has 12,765 GPS trajectories 162 users have participated and a total number of 530 comments are collected and based on the previous discussion, most of them are related to some famous places and final Location-Activity matrix becomes very sparse. An explicit drawback of this sparseness is that, the interval between maximum and minimum values is largely extended and moreover, the available frequencies are distributed around specific points within the interval.

Having such a large interval among the values of Location-Activity matrix causes two problems. One is related with the interpretation of these values. Some values are very large, in the range of a few hundreds, and on the other hand, there are also so many values less than 100. It is obvious that the former group corresponds to "strong recommendation". However, without knowing the whole distribution, it is not very easy to determine whether the values in the latter group correspond to "weak recommendation" or "natural". The second problem is

related with the error calculation. When the range of values is very large and they are not clustered, their impact in error calculations will be related to their actual values. If there is an activity in a location with frequency 250, and if a system predicts a value such as 300, then this will contribute some error. However, both of these values actually correspond to "strong recommendation". Thus, rather than using these actual values, it would be better if abstract and discrete values are used in a small range. To alleviate the impact of this problem on final evaluations, we propose an abstraction method and calculate error terms to this method respectively.

In this abstraction method, we partition the nonzero values of Location-Activity matrix to 5 clusters using *k*-means clustering algorithm. In this algorithm, 5 random points are selected as initial mean values randomly within the ratings values then, each point (rating value in this case) is assigned to a cluster according to shortest value of Euclidean distance from each mean values. In this step for each cluster, mean value is calculated and assigning points to each cluster is performed iteratively, until no point assigned to a new cluster.

In this abstraction method, instead of working on rating values we have examined the cluster that each value belongs to. Note that, the clusters are ranked in a single dimension. Instead of computing error between original value and predicted value, we find the distance between the clusters that original value belongs to and the cluster that predicted value falls into it. As in previous method, we start this method by applying 10-fold cross-validation but in each fold, instead of keeping the predicted value, we find the cluster for this value. Also, for the similar work in [8], each predicted values fall into one of the clusters. In the last step, we use cluster numbers to calculate the error terms. In our experiments, we have clustered values to 5 clusters. In order to clarify, this procedure is demonstrated for the example in previous section. The nonzero values of matrix *X* in (7) are clustered to 3 clusters using k-means and corresponding clusters are:

$$C1 = \{1,1,2,3,4,17,17\} \quad C2 = \{53,53\} \quad C3 = \{76,82\} \qquad (29)$$

According to (29), validation data $x_{11}=1$, $x_{22}=17$ and $x_{33}=53$ belongs to *C1*, *C1* and *C2* and regarding to Table 1 the predicted values for them are *19.5*, *18.5* and *43.16* which indicates that new clusters for predicted values are *C1*, *C1* and *C2*. As an example, let's assume that for a given validation data x_{ij} the original cluster is *C1*, our predicted value falls into cluster *C3* and the predicted value in work [8] falls into *C2* thus, the MAE error can be calculated consequently as $|1-3|=2$ for proposed method and $|1-2|=1$ for similar work in [8].

Figure 3 shows the results of MAE when we apply the abstraction technique. Similar to the previous method each column shows mean value of MAE for 10 folds. We run application for 10 times to show a comprehensive view of results.

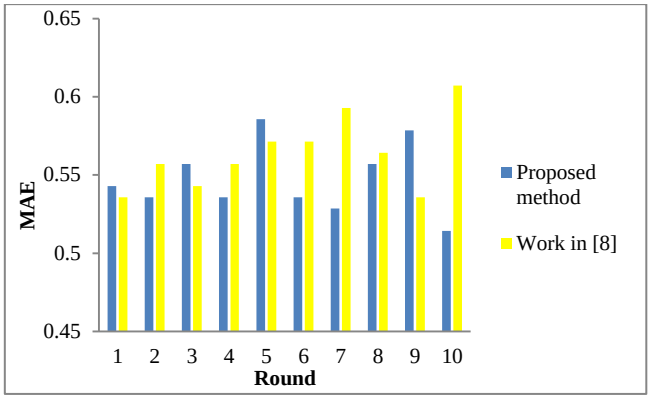


Figure 3. MAE values for proposed work vs. work in [8] with applying abstraction

Finally, Figure 4 shows the mean value of *RMSE* for 10 folds in one column. For the same reason the application is run for 10 times and each column shows mean value of *RMSE* in each time of execution.

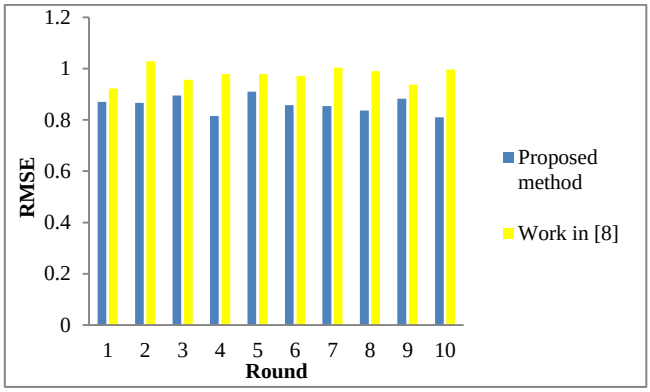
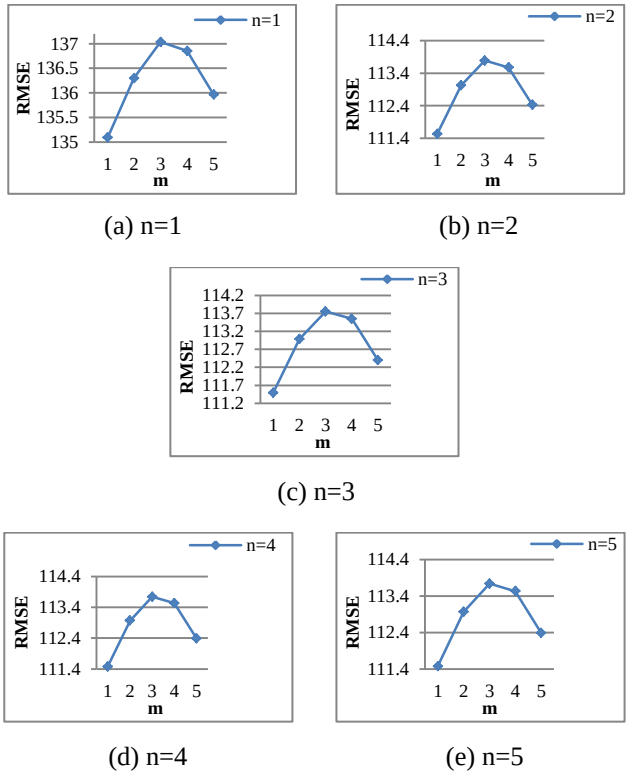


Figure 4. RMSE values for proposed work vs. work in [8] with applying abstraction

Parameter Optimization

Since each data set has its own characteristics, it is not possible to have one set of fixed parameters that works for all. In our work, we have parameterized possible number of top similar locations and top similar activities.

Due to the fact that value of parameters *m* and *n*, which denote the most similar values of rows and columns affect the errors, we have probed different values of these parameters to find the optimal values that reduce the *RMSE*. For each of *m* and *n* we choose values from 1 to 5 thus, we have obtained 25 combinations of them that can be organized in 5 diagrams. We have also implemented it for *RMSE* without abstraction and with abstraction as discussed in previous subsections.

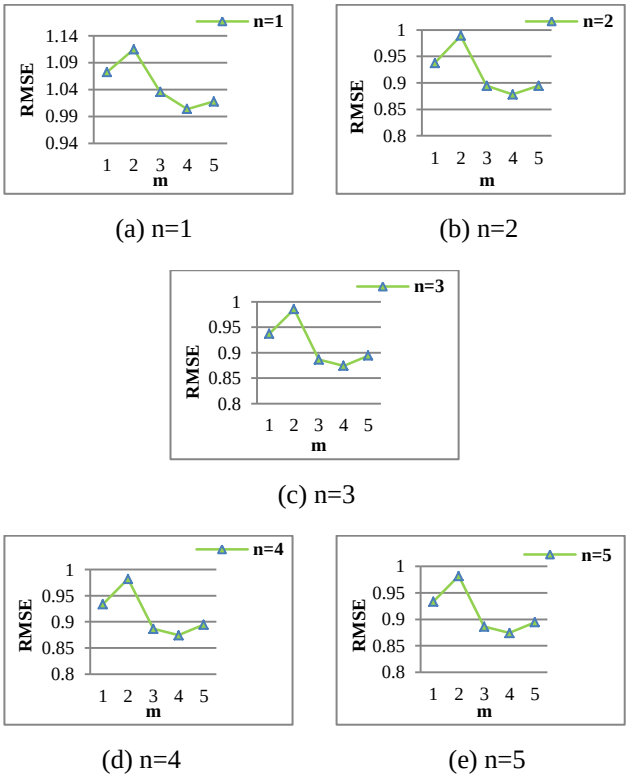


(a) n=1 (b) n=2

(c) n=3

(d) n=4 (e) n=5

Figure 5. *RMSE* vs. parameter *m* without abstraction



(a) n=1 (b) n=2

(c) n=3

(d) n=4 (e) n=5

Figure 6. *RMSE* vs. parameter *m* with abstraction

As illustrated in Figure 5, values of *m=1* and *n=3* when results are not abstracted lead to minimum *RMSE*.

Similarly, Figure. 6 shows that *m=4* and *n=4* are the best values when the abstraction is applied, that gives the minimum *RMSE*.

As shown in Figure 5, in this data set for a fixed *n*, RMSE increases up to *m=3*, and then it starts to decrease. However, for even larger values of *m* it does not reach to the level for *m=1*. Therefore, *m=1* seems like the most appropriate choice. Although for different values of *n* slightly different results are obtained, except for 1 they are quite close to each other. Moreover, among them *n=3* gives the smallest *RMSE* values.

On the other hand, when the abstraction is used, for a fixed n, RMSE shows rather fluctuated behavior for increasing *m*. In all the cases we have tested, *m=4* gives the smallest *RMSE* value, and similar to the previous case except for 1, for all other *n* values results are very close to each other, and *n=4* is the smallest among them.

Considering the size of Location-Activity matrix (167 by 5), searching for similar entries more than 5 would not be feasible. It is not possible to interpret why these values produce the best *RMSE* values, but, it is quite clear that it is directly dependent to the data set (the matrix).

CONCLUSION

In this paper, a new approach has been proposed for combining additional information into main sparse data set for making recommendation. This idea has been applied to geo-spatial data set for activity-location recommendation system. Activity correlation data and location feature data has also been added into the system in order to improve the accuracy for predicting the missing values of the sparse location-activity data set. The same problem has already been investigated in [8]. Unlike that work, which aimed to complete the whole matrix, we have used low-rank approximation approach of SVD in order to reply recommendation requests when they arrive. Since the original matrix has been reduced to smaller matrices, its memory requirement and construction times are much less than the method of 7. Furthermore, we use a different method for prediction, which aims to make prediction only cell-wise. This leads to further time efficiency. Indeed, both approaches have their own pros and cons. In large data sets with low recommendation requests our method is more applicable. Moreover, through some experiments, we have also shown that the accuracy values of our approach are better than the one obtained in [8].

ACKNOWLEDGMENTS

This work has been partially funded by the Greek GSRT (project number 10TUR/4-3-3) and the Turkish TUBITAK (project number 109E282) national agencies as part of Greek-Turkey 2011-2012 bilateral scientific cooperation.

REFERENCES

1. Burke, R. D. Hybrid web recommender systems. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web, Methods and Strategies of Web Personalization*, volume 4321 of *Lecture Notes in Computer Science*, Springer, 2007. 377-408.
2. Huang, Q. and Liu, Y. On geo-social network services. In *Proc. of the 17th IEEE Int. Conf. on Geoinformatics*, Fairfax (VA, USA), 2009.
3. Lax, P. *Linear Algebra and its Applications*, Wiley, 2007.
4. Linden, G., Smith, B. and York, J. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), 2003.
5. Nocedal, J. and Wright, S. J. *Numerical Optimization.* Springer, 1999.
6. Singh, A. P. and Gordon, G. J. Relational learning via collective matrix factorization. In *KDD '08: Proc. of the 14th ACM SIGKDD Intl. Conf. on Knowledge Discovery & Data Mining*, New York, NY, USA: ACM. 650–658.
7. Spiegel, S., Kunegis, J. and Li, F. Hydra: A Hybrid Recommender System [Cross-Linked Rating and Content Information]. In *CNIKM '09: Proceeding of the 1st ACM international workshop on Complex networks meet information; knowledge management* (2009), 75-80.
8. Zheng, V. W., Zheng, Y., Xie, X. and Yang, Q. Collaborative location and activity recommendations with GPS history data. In *WWW '10: Proc. of the 19th International World Wide Web Conference*. New York, NY, USA: ACM.
9. Zheng, V. W., Cao, B., Zheng, Y., Xie, X. and Yang, Q. Collaborative Filtering Meets Mobile Recommendation: A User-centered Approach. In *AAAI 2010*. ACM, 236-241
10. Zheng, V. W., Zheng, Y., Xie, X. and Yang, Q. Towards mobile intelligence: Learning from GPS history data for collaborative recommendation. In *Artificial Intelligence Journal (AIJ)*, 184-185 (2012) 17-37.
11. Zheng, Y. Location-based social networks: Users. In *Computing with Spatial Trajectories*, Zheng, Y. and Zhou, X. Eds. Springer, 2011

TraMSNET: A mobile social network application for tourism

Jorge Gaete-Villegas
Department of Computer Science
KAIST
Daejeon, Republic of Korea
jorge@kaist.ac.kr

Dongman Lee
Department of Computer Science
KAIST
Daejeon, Republic of Korea
dlee@cs.kaist.ac.kr

Meeyoung Cha
Graduate School of Culture Technology
KAIST
Daejeon, Republic of Korea
meeyoungcha@kaist.ac.kr

In-Young Ko
Computer Science Department
KAIST
Daejeon, Republic of Korea
iko@kaist.ac.kr

ABSTRACT
By leveraging location data in online social networks, Location-based Social Networks (LBSNs) can support diverse human activities such as tourism. Different applications aim to aid tourists and provide better experience in their travels by matching co-located users based on what they have in common. However, users with little in common but with potential to help each other given the context and place could not be matched. In this paper we introduce traMSNet, a LBSN that implements a matching algorithm considering homophily, as well as users complementary skills in a touristic location. Our idea is validated with a survey that asked potential travelers about their needs when looking for a travel partner. Moreover, we present a matching algorithm that is evaluated it with real tourists. The evaluation shows that considering complementarity when matching individuals is preferred by users. Therefore, by only considering similarities, important issues are left aside.

ACM Classification Keywords
J.0 Computer Applications: General—*Mobile computing, user recommendation*

General Terms
Algorithms, Human Factors

Author Keywords
Location-Based Social Networks, Homophily, Human Factor, Matching algorithm

INTRODUCTION

Recent advancements in location-acquisition and mobile communications are allowing the emergence of Location-based Social Networks (LBSN) [19]. By enabling users to leverage online social networks with location data, these emerging technologies are supporting the most diverse human activities [9], in particular, tourism [20].

Amongst the elements playing a role into tourists' experiences, human factors and place characteristics are key to achieve a rewarding touristic experience [15]. Tasks such as, interaction with local residents and other tourists and to have a cultural understanding of the location, have demonstrated to be important goals for tourists [11]. These lead to the problem of matching tourists to other people (locals or tourists), in order to enhance their touristic experience and understanding of the location.

Existing applications for tourism [7, 17, 5] implement user matching algorithms from online social networks. The core of these matching algorithms is the concept of homophily, which states that similar individuals are likely to associate with each other more than others [10]. By using this idea, current applications recommend users based on their similarities [12].Nevertheless, relationships between users and places, expressed as knowledge of the place or the locally supported activities, are missed. These relationships, defined as complementary skills, might be beneficial for users that do not have them [14]. However, homophily-based approaches do not consider such relationships.

Considering the above, we state as our research hypothesis that: "In LBSN applications focused on tourism, matching users only in terms of their similarities does not fulfill the needs of tourists." Based on our work, we identify as requirements of the solution: (1) Support on-site serendipity among tourists, (2) Consider the user-place relationship when matching users, and (3) Allow users to communicate on a mobile environment.

In particular, our main contribution is to propose a mechanism that matches users in consideration of their comple-

UbiComp '12, Sep 5-Sep 8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09

mentary skills for a given a place. These skills are inferred from the characteristics not shared by the different users.

We evaluate our approach by means of an online survey to real users. This study faces the respondants to a fictional touristic scenario, and presents five different lists of users to instantiate a LBSN. Each list considering different values oft homophily and complementarity. The results show that consider complementarity is preferred by the users (77% of the sample), therefore, considering only similarities leaves important issues aside.

The rest of the paper is organized as follows. We first survey related work, pointing out how our approach differs from others. Secondly, we introduce the tackled problem and our research hypothesis by a motivation scenario, to later infer the solution requirements out of user survey. Our approach is introduced by presenting the functionalities to match the found requirements and our matching algorithm. After this, we describe our evaluation design, settings and threats to validity in order to clarify the extensions of our conclusion. Then, we present the evaluation results and discussed them to finally, conclude the paper and present future research directions.

RELATED WORK

In this section we briefly introduce relevant study fields related with our work. We first introduce the Location-Based Social Networks, to locate our work in the community. Then, we introduce the concept of homophily into the online social networks, to finally, describe the existing applications supporting the touristic activities.

Location-Based Social Networks

LBSN emerge from the technical advances in location acquisition and mobile communication, which enable users to leverage online social networks with location data [19]. In fact, the user experience in a location can be enhanced with the features of online social networks by, discovering relevant users and supporting serendipitous interactions between them [9]. The goal is to match users sharing similar characteristics within a certain geographic range. Individuals are ranked between them according to the level of similarity they have. Final recommendations are conducted by using stable marriage matching-type algorithms [12].

The generation of LBSN is usually driven by homophily and based on profile information, as hobbies and preferences [7, 5], or information form the location and recurrent visited places [17, 18]. For example, in [1], the authors propose a method to extract similarities between co-located users based on location history. This is later leveraged in [13] to generate a friend recommender amongst co-located users. In [16], the authors extend the user matching problem to the place category (i.e., restaurants, shopping malls, bars, amongst others).

Nevertheless, how to match users with little in common but with potential synergies given their location is not usually considered. This point is what we aim to examine.

Homophily in online social networks

The concept of homophily has been widely studied in the social sciences [10], and extended into the online social network as a driver to generate matching between users [14]. This has been carried out particularly in terms of similar preferences, profiles [8] and behaviours [18]. However, recent literature indicates that online communities are not necessarily formed by homophily between users [6]. In addition, it has been stated that the homophily phenomena from the real world cannot be directly extended to the virtual world [3].

In [14], it is shown how homophily between users is useful to replicate known social patterns, but fail at fostering serendipity. Even more, it is also shown that by reflecting and reinforcing the real life social structures, homophily leads to segregation of users. As a consequence, important synergies between individuals with different interests, preferences and capabilities are missed.

Mobile applications to support tourism

In [2], tourists in a location are identified in terms of the familiarity with the place and the travel distance to get there. A tourist to a place is then characterized as an outsider with little knowledge of the place and the locals. Considering this, the touristic activity represents an interesting yet, challenging domain for supporting users' goals. By one hand, tourism is an activity highly dependent on the place [11]. On the other hand, tourists in a given place share the interest to discover it, but are unfamiliar with each other and the place [15].

The above has fostered the existence of multiple applications supporting LBSN and, in general, aiding the user to achieve his touristic goals. In [18], the authors propose an itinerary recommender system by using location history of past tourists. Other series of applications serving as mobile tourist guides are summarized in [20], offering services such as ticketing to local attractions, virtual tours, and communication tools between co-located users.

In general, the place is always considered as a source of similarities. Nevertheless, how the place characteristics align the differences of users into synergies is not well studied. This representing an interesting issue addressed in this paper.

HYPOTHESIS AND REQUIREMENTS

In this Section we firstly introduce our research hypothesis through a motivation scenario. Then, we infer the requirements for the solution from a prospective survey.

Motivation scenario and requirements

Consider the following scenario, introduced to better ilustrate our research hyphotesis:

"Benjamin is a medicine student backpacking through Europe. He speaks French and Spanish, loves football and barbecue. Nevertheless, he is too shy to just randomly talk to anyone. Barbara is an art student from Spain, vegetarian.

Concern	Answer	Included answer (examples)
Language	20	Speaks the native language, English
Age/Gender	11	Age, gender
Experience in the place	9	Experience in the place or similar places
Personality	10	Is he open minded, funny?
Nationality	9	Local, native
Knowledge background	10	Understanding the place and culture

Table 1. Survey results and concerns derived

Concern	Objective 1	Objective 2
Language	Partner communication	Interaction with locals
Age/Gender	Similarity	——-
Experience in the place	Understanding the place	——-
Personality	Similarity	——-
Nationality	Partner communication	Understanding local culture
Knowledge background	Saftey and similarity with partner	Understanding the place and culture

Table 2. User concerns vs user objective

She doesn't likes sports and can't speak French. They are strangers to each other.

She is in the Louvre, just a couple of blocks away from Benjamin, looking for someone to share the Louvre, a place she knows well in advance since she studies Art. She looks up her smart phone where a suggestion for chatting with Benjamin pops up.

Since Benjamin needs someone to guide him and Barbara needs someone who can speak Spanish, they get connected and enjoy the day at the museum".

The above scenario presents the issue of matching users engaged in a touristic situations. The main problem is to match users that do not share interests necessarily, but may benefit from each other in the touristic domain.

In consideration of the above, we state as our research hypothesis that: "In a LBSN applications focused on tourism, matching users in terms of their similarities does not fulfil the needs of tourists".

A user requierements survey

A survey was conducted in order to validate the hypothesis introduced above, and to understand the user requirements to be considered when matching travel partners.

The study was conducted through an online questionnaire to 20 potential users [1], 9 female (45%) and 11 males (55%). In terms of age, the sample is composed by 3 under 19 years old (15%), 11 between 20 and 25 years (55%), 4 between 26-30 years old (20%) and, 2 between 31-35 years old (10%). Demographically, the sample is divided as follows: 9 Americans (45%), 5 Europeans (25%), 4 Africans (20%) and 2 Asians (10%). Participants were asked about their top five concerns when looking for a travel partner. Since this is an exploratory survey, the questionnaire had no fixed answer. For analysis purposes, the answers were manually classified into six concerns [2] as shown in Table 1.

[1] A potential user was considered to be an individual with experience in a foreing country for a short period of time in typical touristic activities as described in [2, 11]

[2] These appeared out of the data and where not defined in advance

Based on these concerns, we can infer the underlying motivations of users when they look for these characteristics in a travel partner. For instance, while (i) "Does he speaks English?" and (ii) "Does he speaks the local language?" account for a language concern, in (i) the user is assessing the communication potential with his travel buddy, and in (ii), he is looking to improve his communication with locals through his travel partner. Table 2 presents an interpretation of the underlying objectives of users for each found category.

Considering the above, an important question arises: How well does homophilly as a matching criteria accounts for these concerns? Table 3 presents a qualitative analysis on the feasibility of similarity and complementary to support the objectives listed in Table 2.

In Table 3 it is shown that, as mentioned in the literature [14], similarity is not enough to support all of the user needs. Moreover, supporting four out of six user's objectives, complementarity proves its importance as a matching criteria.

The mayor conclusions from the survey are: (i) language is the most important characteristic *-it was common to every surveyed, regardless of gender, nationality or age-*, (ii) the rest of the categories are equivalently important *-therefore only considering similarity leaves relevant issues aside-*, and (iii) in this touristic context, potential partner's knowledge on the place itself and/or related activities is, at least, as important as personality aspects *-this, since the number of responses for "Experience in the place" and "Knowledge backgrounds" equals the number of responses for "Personality" -*.

Finally, considering our hypothesis and motivation scenario, our requirements survey and the related work drawbacks [14], we deduce the following three requirements for our solution:

1. **Consider the user-place relationship when matching users :** To meet the concerns of users related with the place characteristics, such as "Experience in the place" and "Knowledge background".
2. **Support on-site serendipity among tourists:** To exploit the capabilities of other unknown co-located users.
3. **Allow users to communicate on a mobile environment:** To allow interactions occur dynamically.

Concern	Homophily	Complimentarism
Language	Yes	Yes
Age/Gender	Yes	No
Experience in the place	No	Yes
Personality	Yes	No
Nationality	Yes	Yes
Knowledge background	Yes	Yes

Table 3. User concerns vs how homophily and complementarism support these

PROPOSED SOLUTION

In this section we present the proposed schema to match the inferred requirements. We firstly present the set of functionalities needed by the application to account for the mentioned requirements. Then we present the data used to model both, the user and the place. Finally, propose a matching algorithm considering our research question and the results from the user requirements survey.

Functionalities

Functionalities identified are common to many applications found in the literature [9, 20]. Following is a brief presentation of these and how they respond to our functional requirements.

- *Local map generation and display:* Its two main objectives are: (i) allow tourista to locate other co-located users and, (ii) identify interesting touristic places. By doing this, the map fosters the communication between co-located users while provides the user with a better knowledge of the sorroundings. The provided map shows touristic points in the nearby and recommends partners according to a ranking algorithm.
- *Mobile social network:* Its objective is to materialize the interaction between users through a chat service. By doing this, it encourage on-site serendipity and communication amongst users.
- *User filtering, ranking and recommendation:* This is the core application of our work. It supports on-site serendipity amongst users by considering the location where the interaction is taking place. This is done by ranking users considering different parameters, accounting for complementary skills given the place (i.e. language) and, by regular homophily driven parameters (i.e. preferences).

Data for the modeling of users and locations

In the following, we present the data used for modeling the users and places. These data respond to the results of our conducted survey. They reflect the identified user concerns, while being used as inputs to the matching algorithm.

1. User profile
   - Age(numeric): User Age
   - Gender(string): User gender
   - Nationality(string): User nationality
   - Spoken language(string list): User spoken language
   - Major(string): User highest academic degree
   - Hobbies(string list): User hobbies
   - Past touristic experience (string list): Touristic locations visited by the user.
2. Place profile
   - Country (string): Country of location of the touristic place.
   - Spoken language (string list): Native languages spoken at the location.
   - Major activities (string list): Major activities performed at the location.
   - Related places (string list): Similar locations based on the above characteristics.

Given this characterization for users and places, users can be compared between them and with places. Comparisons between users are done by matching the correspondent fields of the two vectors representing them. On the other hand, users and places are compared only in the fields defined as equivalent between them. These are: (i) location country and user nationality, (ii) location spoken language and user spoken language, (iii) location major activities and user major and hobbies and, (iv) location related places and user past touristic experience.

Matching algorithm

Considering *u* as the target user, our matching algorithm returns a ranked list of the users nearby him, in consideration of his affinities and needs given his current location and, commonalities and synergies with the nearby users.

The algorithm first calculates the needs of *u* given the place *l* (See Algorithm 1, line 2). These needs are represented as a complementary user to *u* in *l*. This new user, *u'*, is generated by eliminating from *l* all the characteristics already possesed by *u*, (i.e. same language or nationality). The resultant is a new characteristics vector, containing the needs of u in the place l, as defined in Equation 1.

$$\vec{u'}(u,l) = \vec{l} - \vec{u} \tag{1}$$

Later, similarities and complementary skills are calculated between *u* and *j* [3] (Algorithm 1, lines 4 and 5). Similarities are calculated as the percentaje of fields sharing the same metric between the descriptive vector for users *u* and *j*, while complementary skills are computed in the same way but between *j* and *u'*, the complement of *u*. Both metrics are calculated by the Homophily function H, evaluated as [5, 9]:

[3]They are both calculated as explained in the footnote number 2

Algorithm 1 UserMatching(homophily, user, users, place)

Require: Homophily value α
Require: User characteristics vector *u*
Require: Place characteristics vector *l*
Require: Other users characteristics matrix *J*
1: $\beta \leftarrow 1 - \alpha$
2: $u' \leftarrow diff(l, u)$
3: $recommendation \leftarrow Array[5][2]$
4: **for all** collumn j in J **do**
5: $homophily \leftarrow sim(u, j)$
6: $complementarity \leftarrow sim.2(u', l)$
7: $rank \leftarrow \alpha * homophily + \beta * complementarity$
8: **for all** n in $\{1...5\}$ **do**
9: **if** $recommendation[n][2] < rank$ **then**
10: $recommendation[n][1] \leftarrow j$
11: $recommendation[n][2] \leftarrow rank$
12: **end if**
13: **end for**
14: **end for**
15: **return** $recommendation$

$$\vec{H}^n_{i,j} = \begin{cases} 1 - (i(n) - j(n))/i(n) & \text{if n is numeric value} \\ 1 - (\max i(n) - \min j(n))/N & \text{if n is a numeric range} \\ \sum_{n=1}^{N} (i(n) \wedge j(n))/N & \text{if n is a list of strings} \end{cases} \quad (2)$$

Finally, the matching algorithm implements the ranking function presented in Equation 1. Initially this function is expressed as in Equation 3:

$$\vec{R}(u, j) = \alpha \vec{H}(u, j) \cdot \vec{W}^H_m + \beta \vec{P}^l(u, j) \cdot \vec{W}^P_n \quad (3)$$

The right side of the Equation (3) is composed by a measurement of homophily between *u* and *j* (multiplied by α) and a measurement of complementarity between *j* and *u* given a place *l* (multiplied by β). α and β represent the relative importance between homophily and complementarity, thus $\alpha + \beta = 1$. Both sides of Equation 3 are multiplied by weight vectors $\vec{W}^H$ and $\vec{W}^P$, accounting for the importance of the features compared by the functions *H* and *P*.

While *H* compares similarities and *P* compares complementary skills between users considering a place, by considering Equation (3) we can express the ranking function only in terms of homophily. By considering *u'*, *P* can be expressed in terms of the function *H* as $\vec{P}^l(u, j) = \vec{H}(u', j)$. In turn Equation (3) can be represented as:

$$\vec{R}(u, j) = \alpha \vec{H}(u, j) \cdot \vec{W}^H_m + \beta \vec{H}(u', j) \cdot \vec{W}^P_n \quad (4)$$

Finally, with this representation, the implemented UserMatching algorithm returns a recommendation by considering the top ranked users by Equation 4 (line 9 and 10 in Algorithm 1).

EVALUATION

In this section we present the evaluation process by explaining its design and implementation. The main goal is to verify our research hypothesis). In order to do so, the object of analysis considered was the user's opinion on the appropriateness of a LBSN for a given tourist situation. In turn, a LBSN is represented as a list of users, ranked by some eligibility criteria.

At this point, the ranking algorithm is introduced to suggest a LBSN. This algorithm is the focus of the evaluation, considering it is our main contribution.

Evaluation design

The main concern for the evaluation design and measurement process is the subjectivity of the analysis object: the user's opinion. By one hand, preferences are hard to capture based on objective metrics [4]. On the other hand, we need to ensure that the user chooses amongst different options with the same background information.

Considering the above, the evaluation was conducted by using a Web-based survey. The survey tool allows to (i) directly measure the user's opinion, avoiding inferences made from qualitative metrics and, (ii) ensure that the respondent evaluates the different suggestions with the same knowledge about the evaluation scenario, since multiple scenarios can be presented simultaneously. The procedure for the evaluation is listed below:

1. The surveyed is presented an online questionnaire asking for his/her profile information.
2. After completion of (1), the user is presented a fictional scenario and asked to read it. The fictional scenario includes country, language, place description, etc.
3. After reading (2), the user is shown simultaneously five LBSN.
4. Finally, the surveyed is asked to choose the lists he would have preferred to receive in the given situation (from step 2).

Implementation of the matching algorithm in the evaluation

The web-based survey presented above, implements the ranking function (Equation 1) to generate the five LBSN presented to the user (step 3).

This is done by computing the function in running time while the survey is being performed. Each list is created by running the matching algorithm and considering the top five ranked users.

Each list considers different values for homophily and complementarity. The values for alpha range from 100% (as any homophily-based algorithm) to 0%. Correspondingly, the values for beta range from 0% to 100%.

The required data for the process is obtained from the surveyed profile (input from step 1) and the tourists profile database gathered as explained in the next subsection.

These five recommendation lists are simultaneously displayed to the user in the same Web pagef. The disposition of these lists in the Web page is randomly assigned, this in order to avoid the questionnaire effect [4] and the effect of users randomly choosing options.

Evaluation settings

The evaluation considers presenting the surveyed possible users for the instantiation of a LBSN. In order to make our survey as realistic as possible, we consider real tourist profiles for the matching generation.

The data was gathered by an online survey and includes the profiles of 57 tourists (34 female and 23 male) between the ages of 19 and 50, from Africa (8), America (11), Asia (20) and Europe (18).

In accordance to the data description in the Proposed solution section, the profile information includes age, gender, nationality, spoken languages, formal studies, occupation and hobbies. Respondents were asked about their travel experience as well. They were presented 10 major touristic cities and asked if they have: i) been there as tourists and ii) what activities they performed.

The cities are[4]: Paris (28), London (23),Tokyo (18), New York (20), Dubai (10), Singapore (21), Kuala Lumpur (19), Hong Kong (19) and Seoul (51).

The activities considered are: attendance to museums or expositions, attendance to theatrical or musical performances, attendance to sport events, cultural/traditional sightseeings, eco tourism, other. Given that the option "other" was prefered in a 5% in average (min 3%, max 11%), the given response options made the respondents felt identified by the given alternatives.

Since the great majority of the respondants had visited Seoul, the location for conducting the evaluation is the GANA Art Gallery, located in Seoul, South Korea. Specifically near Gwanghwamun Gate in downtown Seoul, a district known for its afluence.

Threats to validity

The evaluation objective is to validate the importance of considering complementarity between users as a matching criteria. Therefore, some aspects of our approach and the effect of these in the user's preferences are not considered in the current evaluation.

The above mentioned threats are challenging issues for future work. Nevertheless, given the scope of this paper, keeping these unknown effects as constant for the entire sample allows us to evaluate the relative importance between homophily and complementarism.

Evaluation model

The model is tested by modifying the values of α and β in Equation (2) *ceteribus paribus*. By keeping the rest of the variables fixed, the effects of these on the user's opinion is not considered. For instance, giving more importance to spoken language rather than hobbies, may lead the user to choose differently, even for the same values of α and β.

[4]In parentesis the number of contestants that have visited as tourists

Effects on the evaluation display

Presentation effects (such as images and colors) were not considered in the final evaluation visual interface. We aim to take this effects out of the evaluation by showing the information in the most plain and impartial possible fashion. Some of the effects not considered are modifying the visualization interface and the use of pictures for enriching the profiles. For example, if images are available as part of the profile, users may choose based on attractiveness of the user instead of the objective information.

RESULTS AND DISCUSSIONS

This section contains the results obtained from the performed evaluation, as well as a brief discussion on relevant observation. The survey was answered by 21 volunteers, none of these participated in any of the initial surveys included in this paper. The volunteers were considered as tourists according to the criterions defined in [2], this is, (i) they are not from Seoul, and (ii) they have never being in the GANA art gallery.

Results

Results are presented in terms of the values of α and β used for computing the recommendations. Since α and β add 1, as α decreases, homophily is less relevant for the recommendation and the more complementarism is considered. Table 4 presents the overall results obtained from the Web based evaluation.

Homophily and Complementary (α, β)	**Preferences**	%	**Cumulative** %
(0.0;1.0)	3	14.29	14.29
(0.25;0.75)	2	9.52	23.81
(0.5;0.5)	6	28.57	52.38
(0.75;0.25)	5	23.81	76.19
(1.0;0.0)	5	23.81	100
Total	21	100	—-

Table 4. Results for abroad toursits vs local tourists

As shown in Table 5, only 23.81% of the preferences include solely homophily while, 76.19% of the users chose a recommendation considering some degree of complementarism. The larger portion of observations are located in the β range between 1.0 and 0.5 (76.19% of the sample)

Another important dimension of the sample is the comparison between overseas tourists and local tourists. We considered as local tourists those having some cultural background on the touristic scenario. Given our data, we consider as local tourists those with Korean nationality and/or declaring proficiency in Korean language. By this criteria, five out of the twenty one samples were considered as local tourists. The results are exposed in Table 5.

Homophily and Complementary (α, β)	Preferences	Local tourists	Overseas tourists
(0.0;1.0)	3	0	3
(0.25;0.75)	2	1	1
(0.5;0.5)	6	0	6
(0.75;0.25)	5	1	4
(1.0;0.0)	5	3	2
Total	**21**	**5**	**16**

Table 5. Results for abroad toursits vs local tourists

Four out those five local tourists considered as more appropriate the suggestions highly influenced by homophily. Only one out of the five locals chose a LBSN with high values of complementarity.

Discussion

In general, the results confirm the importance of considering both, the place an the personal characteristics for LBSN in the touristic domain. This can be pointed out since over three quarters of the respondents chose recommendations combining complementary skills and homophily.

It is interesting to notice that our algorithm only considers complementary skills as they are needed. If the user have all the necessary skills to enjoy the location, the complementary user does not exist and the recommendations are only driven by homophily. In particular, this explains the different preferences between local and overseas tourists. Since overseas tourists need more skills (i.e. language), they chose suggestions combining homophily and complementary skills. On the other hand, locals preffered recommendations based solely on homophily since the influence of the complementary user is less important.

CONCLUSIONS AND FUTURE WORK

In this paper we addressed the idea and evaluate the importance of using complementary skills between users as a matching criteria for the generation of spontaneous LBSN in the tourism domain. We successfully verified our research hypothesis, concluding that the complementary skills of users should be considered, when possible, in order to recommend a LBSN.

An important observation is that our matching algorithm behaves as a homophily-based algorithm for locals, which indicates our algorithm is a generalized form for the algorithms based on similarities.

Even when our results are positive, they open new research questions, which constitute our future work. The effect of the variables of the ranking algorithm is still to be evaluated, as well as the extensibility of the model to other domains.

In this way, the calibration and importance of the variables considered in the ranking function (Equation 1) are important towards the provisioning of a personalized Location-Based Social Network.

ACKNOWLEDGMENTS

This research was supported by the KCC(Korea Communications Commission), Korea, under the RD program supervised by the KCA(Korea Communications Agency) (KCA-2012-11911-05005). The authors would also like to thank Angel Jimenez-Molina and Pablo Loyola for their comments on the draft, and Blandine Yochie for her collaboration in the project.

REFERENCES

1. Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, GIS '08, pages 34:1–34:10, New York, NY, USA, Nov. 2008. ACM.

2. N. Leiper. The framework of tourism: Towards a definition of tourism, tourist, and the tourist industry. *Annals of Tourism Research*, 6(4):390 – 407, Oct.-Dec. 1979.

3. M. A. Ahmad, I. Ahmed, J. Srivastava, and M. S. Poole. Trust me, i'm an expert: Trust, homophily and expertise in mmos. In *SocialCom/PASSAT*, pages 882–887. IEEE, Oct. 2011.

4. C. M. Anderson-Cook. Experimental and quasi-experimental designs for generalized causal inference. *Journal of the American Statistical Association*, 100(470):708–708, 2005.

5. V. Arnaboldi, M. Conti, and F. Delmastro. Implementation of cameo: A context-aware middleware for opportunistic mobile social networks. *A World of Wireless, Mobile and Multimedia Networks, International Symposium on*, 0:1–3, June 2011.

6. H. Bisgin, N. Agarwal, and X. Xu. Investigating homophily in online social networks. In J. X. Huang, I. King, V. V. Raghavan, and S. Rueger, editors, *Web Intelligence*, pages 533–536. IEEE, Aug. 2010.

7. A. Garcia-Crespo, J. Chamizo, I. Rivera, M. Mencke, R. C. Palacios, and J. M. G. Berbis. Speta: Social pervasive e-tourism advisor. *Telematics and Informatics*, 26(3):306–315, Aug. 2009.

8. N. Kayastha, D. Niyato, P. Wang, and E. Hossain. Applications, architectures, and protocol design issues for mobile social networks: A survey. *Proceedings of the IEEE*, 99(12):2130 –2158, Dec. 2011.

9. G. Lugano. Mobile social software: definition, scope and applications. *Proceedings of the IEEE*, 99(12):2130 –2158, Dec. 2011.

10. M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

11. E. Wickens. The sacred and the profane, a tourist typology. *Annals of Tourism Research*, 29(3):834 – 851, July 2002.

12. H. Rahnama, A. Madni, A. Sadeghian, C. Mawson, and B. Gajderowicz. Adaptive context for generic pattern matching in ad hoc social networks. In *Communications, Control and Signal Processing, 2008. ISCCSP 2008. 3rd International Symposium on*, pages 73 –78, March 2008.

13. Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma. Recommending friends and locations based on individual location history. *ACM Trans. Web*, 5(1):5:1–5:44, Feb. 2011.

14. J. Thom-Santelli. Mobile social software: Facilitating serendipity or encouraging homogeneity? *IEEE Pervasive Computing*, 6:46–51, July-Sept. 2007.

15. M. Su and G. Wall. Implications of host-guest interactions for tourists' travel behaviour and experiences. *TOURISM - An International Interdisciplinary Journal*, 58(1), May 2010.

16. X. Xiao, Y. Zheng, Q. Luo, and X. Xie. Finding similar users using category-based location history. *GIS'10*, pages 442–445, Nov. 2010.

17. Y. Zheng, Y. Chen, X. Xie, and W.-Y. Ma. Geolife2.0: A location-based social networking service. In *Mobile Data Management: Systems, Services and Middleware, 2009. MDM '09. Tenth International Conference on*, pages 357 –358, May 2009.

18. H. Yoon, Y. Zheng, X. Xie, and W. Woo. Smart itinerary recommendation based on user-generated gps trajectories. In *Proceedings of the 7th international conference on Ubiquitous intelligence and computing*, UIC'10, pages 19–34, Berlin, Heidelberg, Oct. 2010. Springer-Verlag.

19. Y. Zheng. Location-based social networks: Users. In Y. Zheng and X. Zhou, editors, *Computing with Spatial Trajectories*, pages 243–276. Springer New York, July 2011. 10.1007/978-1-4614-1629-6_8.

20. M. Kenteris, D. Gavalas, and D. Economou. Evaluation of mobile tourist guides. In *The Open Knowledge Society. A Computer Science and Information Systems Manifesto*, volume 19 of *Comminications in Computer and Information Sciences*, pages 603–610. Springer Berlin Heidelberg, 2008. 10.1007/978-3-540-87783-7_77