TSFR: Time-aware Semantic-based Friend Recommendation in LBSNs

Xiaoyan Zhu, Linjie Zhang, Yizhe Huang, Baoming Bai, Jianfeng Ma
State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
(xyzhu@mail, linjiezhang@stu, yzhuang@stu, bmbai@mail, jfma@mail).xidian.edu.cn

***Abstract*—Advances in broadband wireless networks and location sensing technologies have led to the emergence of location-based online social networks (LBSNs) in recent years. Users' passion for sharing locations has attracted much attention to traditional social networks. Therefore, the great amount of check-in data can be used to make recommendations for interesting places and to make friends. Because both semantic and time information on check-in data reflect preference and interests of users, we take both of them into consideration and propose a time-aware semantic-based recommender system in this paper. We use the Term Frequency-Inverse Document Frequency (TF-IDF) model and the Kullback-Leibler divergence (K-L divergence) to combine the semantic and time information of check-in data to make friend recommendations. To evaluate our recommender system, we get a dataset of Gowalla and build a system using the Collaborative Filtering recommender system structure. The experiment results show that our system, with the consideration of time and semantic information of check-in data, outperforms the classic collaborative filtering recommender system.**

***Index Terms*—Location-based social networks; Recommender system; Time aware; TF-IDF; Collaborative filtering**

I. Introduction

Location-based online social networks (LBSNs) like Foursquare help people share their locations online to find interesting places and make friends. Additionally, with the popularity of smart phones, the built-in GPS can detect locations more accurately, which makes users share their locations more conveniently. Therefore, the check-in service attracts more and more users. Meanwhile, the recommender system plays an important role in social networks. The check-in history contains a great amount of preference data of users. Collaborative filtering [1] is used widely in the recommender system. In a location-based social network, a user typically checks in a small part of the locations in the dataset, in which most entries to the user-item/location matrix appear to be zero. Therefore, the check-in data is too sparse to use directly to make recommendations [2]. To handle the data sparsity in location-based social networks, Koren *et al.* [3] use matrix factorization techniques to make use of relatively denser implicit feedback to infer preferences of users. Additionally, He *et al.* [4] contribute improvements on both the effectiveness and efficiency of Matrix Factorization method for implicit feedback. Qi *et al.* [5] propose a time-aware service recommendation approach named $SerRec_{time}$. Concretely, they first calculate the time-aware user similarity; afterwards, indirect friends of the target user are inferred by the Social Balance Theory. Besides the data sparsity, we also need to quantify the raw check-in data to explicit scores properly to make recommendations in a Collaborative Filtering-based recommender system. In the process of quantifying the check-in data, both Zheng *et al.* [6] and Bao *et al.* [7] use the Inverse Document Frequency (IDF) in information retrieval to lower the weight of very popular locations. The TF-IDF model tends to filter out common words and retain important words, so we also use this technique to get categories aptly. The advantage of the TF-IDF model is that the simple and fast results are more in line with the actual situation.

Strong temporal effects have been pointed out in the user movement in location-based social networks. We believe that time plays a significant role in recommendations because most users tend to visit different places in the different time in a day. It is a propitious time to concern these temporal effects on a user's mobile behavior. As observed, human movement exhibits strong temporal effects in terms of hours of the day and days of the week. Recommendation accuracy would be decreased if the time factor is overlooked, as service quality often varies with time. To the best of our knowledge, some studies [8] [9] [10] find the importance of temporal dynamics in human activities. It encourages us to exploit these temporal effects for modeling a user's temporal preferences. Therefore, we regard the precise time information on check-in data as a wealthy resource to dig users' preference and interests. We present the results of such study and outline application areas where the conjunction of location and temporal-aware data can help in the further search.

As far as recommender systems considering time information, Luo *et al.* [11] revamp collaborative filtering recommendation approaches to model the drift of users' preferences. They evaluated their system on the large movie rating dataset of MovieLens. Furthermore, Shi *et al.* [12] propose a network evolution method to simulate the mutual feedback between the recommender systems and their users' decisions in the evolving network with time. Ye *et al.* [13] discuss the time attribution of a location, such as weekend or weekday locations, and whether it is visited during daytime or in the night. Then, they use the time attribution to predict the type of untagged places based on the check-in time of users. The works above

This work was supported in part by projects of National Nature Science Foundation of China under Grant 61772406 and Grant U1636209, in part by the project of Fundamental Research Funds for the Central Universities under Grant JB180110, in part by Industrial Research Project of Shaanxi Province under Grant 2015GY008. *(Corresponding author: Xiaoyan Zhu)*

usually analyze the long term time information to infer the trend of users' behavior. Their usage scenario is usually used in the e-commerce web sites or the ratings of movies. Compared with purchasing history or movie ratings, the check-in data is more sensitive to the specific time of a day. Because people typically make the online purchase at their leisure, checking the purchasing history or ratings is not sufficient enough to be used for the friend recommendation. Besides, yuan *et al.* [14] show that time has a significant influence on accuracy of Point-Of-Interest (POI) recommendations, improving on the recommendation accuracy by 37% to 51% over the method without considering time. Tuan *et al.* [15] propose a location-based collaborative filtering recommendation system with dynamic time periods for recommending timely and suitable POI to mobile users. The system expedites calculating similarity based on POI recency and enables mobile users to promptly obtain recommended items that closely match their current space-time conditions by selecting different strategies for dissimilar situations. Li *et al.* [16] put forward a fourth-order tensor factorization-based ranking methodology to recommend users locations by considering their time-varying behavioral trends while capturing their long-term preferences and short-term preferences simultaneously. By judging the degree of conformity among the behavior patterns of different users based on the time frame, we utilize relative entropy of check-in time among users to adjust cosine similarity and make friend recommendations. In order to achieve significantly superior recommendations compared to other state-of-the-art recommendation techniques with temporal influence, we use multi-level granularity for time analysis considering the sparseness of data.

The main contributions of TFSR are listed as follows:

- We analyze the check-in distribution in a day to illustrate the check-in pattern of users. We also use TF-IDF to balance users' preferences and location popularity when quantifying the raw check-in data.
- We take the time information on check-in data into consideration to make friend recommendations by calculating K-L divergence among users' check-in distribution. Aside from recommender systems based on GPS sequences, our scheme is suitable for discrete check-in information, which is common in many social networks.
- To evaluate our recommender system, we run it on a dataset of Gowalla and the results show that our recommender system works efficiently. The experimental results on real-world location-based social networks datasets validate the power of temporal effects in capturing user mobile behavior.

The remaining paper is structured as follows: In Section II, we introduce some preliminaries of our scheme and present the whole process of our recommender system. Section III presents the whole process of our recommender system. Following in Section IV, we show the setup and results of our experiment. We finally conclude this paper in Section V.

II. Preliminaries

A. TF-IDF

TF-IDF is a numerical statistic that is intended to reflect how important a word is to a document. Ramos *et al.* [17] describe the use of TF-IDF in information retrieval system and provide evidence that a word with high TF-IDF value implies a strong relationship with the documents it appears in. As for the calculation of TF-IDF, it is the product of two statistics: Term Frequency (TF) and IDF.

The term frequency $tf(t,d)$ is a measure of the number of times that term t occurs in document d. In our scheme, we calculate TF as follows:

$$TF(t,d) = \frac{|t : t \in d|}{|d|} \tag{1}$$

where $|t : t \in d|$ is the number of term t in document d, and $|d|$ is the total number of terms in document d.

The inverse document frequency is a measure of how much information the word provides, that is, whether the term is common or rare across all documents. It is obtained by Equation 2:

$$IDF(t,D) = \log \frac{N}{|\{d \in D : t \in d\}| + 1} \tag{2}$$

where N is the total number of documents in the corpus, $|d \in D : t \in d|$ is the number of documents where the term t appears.

Then, with the TF and IDF, TF-IDF is calculated as:

$$TF\text{-}IDF(t,d,D) = TF(t,d) \times IDF(t,D) \tag{3}$$

Essentially, TF-IDF works by determining the relative frequency of words in a specific document compared to the inverse proportion of that word over the entire document corpus.

B. Kullback-Leibler divergence

In probability theory and information theory, K-L divergence, also known as relative entropy, is a measure of the difference between two probability distributions P and Q. The K-L divergence of Q from P is defined as:

$$D\left(P \parallel Q\right) = \sum P\left(i\right) \log\left(P\left(i\right)/Q(i)\right) \tag{4}$$

Note that although K-L divergence is often intuited as a metric of distance, it is not a true metric. It is not symmetric; the K-L divergence from P to Q is generally not the same as that from Q to P.

In our scheme, we only use the discrete form of K-L divergence; we do not introduce the continuous random variable form here.

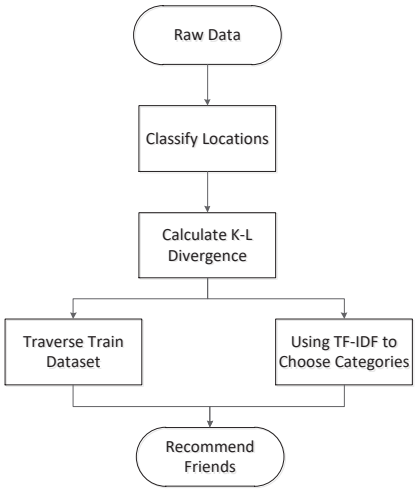

Fig. 1. System Architecture

III. TSFR

TSFR consists of compressing the user-location matrix, calculating K-L divergence of each category among users, and determining the lists to use and make recommendations.

To handle the aforementioned data sparsity in a conventional Collaborative filtering based recommender system, we classify the locations into some categories based on the semantic information. In the process of quantifying the check-in data, we use TF-IDF in an information retrieval system to balance users' preferences and location popularity. In order to blend the temporal information about check-in data into our system, we calculate the K-L divergence between users check-in distribution in a day. Then, we get an adjusted factor based on the K-L divergence to use in the similarity calculating. Last, we pick top-n users of the similarity list to make friend recommendations.

Besides the location information used widely in the location-based recommender system, our system takes the time information about check-in data into consideration to make friend recommendations, which is rarely used in other location based recommender systems. Then, we build it on a collaborative filtering structure.

A. Compress the user-location matrix

The collaborative filtering-based recommender system mentioned above needs a user-item matrix to store the rating information of users and corresponding items. In our recommender system, it is a matrix about user and locations. The entries of the matrix represent the amount of users' check-ins of the corresponding locations. However, in a location-based social network, a user could usually just visit a small portion of the locations in the dataset, so it appears that many of the entries in the user-location matrix are zero. The sparsity of data will lower the usability of the check-in data and waste storage space. To handle this problem, we will try to compress the matrix without affecting the quality of recommendations.

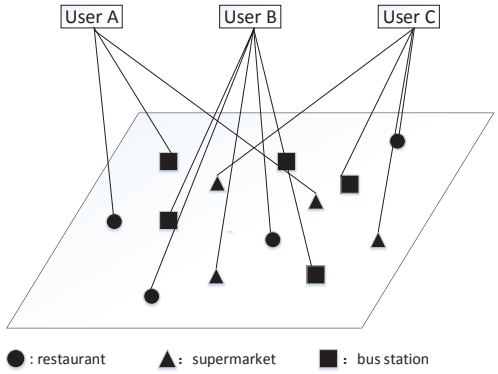

Fig. 2. check-in distribution

A typical check-in distribution on the location dimension is shown in Figure 2. Two adjacent geographical locations may be quite different based on the preference and interests information they contain. Compared with the geographical information about check-in locations, the preference and interests information contained in the check-in data are more valuable to make recommendations. Therefore, we try to extract the locations of similar interests in a category. Then, we compress the sparse user-location matrix in order to classify the locations in the dataset based on preference and interests. We need to get the semantic information about the locations. Using the venue search API provided by Foursquare, after querying Foursquare with the longitude and latitude of the locations, we will get specific details about the locations including the location category information. Using the category information from Foursquare, we can compress the original user-location matrix into a denser and easier to use user-category matrix. Every entry of the matrix represents the amount of check-in data of a user and locations belonging to the corresponding category.

Then, we try to quantify the amount of check-in data in the user-category matrix to illustrate users' preference and interests on locations. We use the TF-IDF model in the information retrieval system to transform the raw check-in data to explicit scores of users and corresponding location categories.

B. Calculate Kullback-Leibler divergence of each category among users

The behavior of users has certain rules in time. Therefore, if the time attributes of behaviors are taken into account in the recommendation method, the user's willingness to make friends will be more accurately reflected. As mentioned above, some locations are usually checked in only in a period time of a day. For example, a bar is usually checked in at night,

and a restaurant is often checked in at noon or evening. These characteristics of check-in time distribution can be extracted as a pattern of locations. Likewise, every user has his own check-in pattern, which appears as probability distribution of time in a day. We find it meaningful to recommend friends with similar check-in distribution to users. We use the K-L divergence of check-in distribution to measure the difference between users.

After calculating the K-L divergence for each user, we can compile a list of each location category in ascending order by K-L divergence. Then, we can use these lists to make friend recommendations.

C. Determine the lists to use and make recommendations

Every user has his/her preferable check-in locations, so using all of the K-L divergence lists to make recommendations cannot satisfy the users. Then, we need to choose the proper lists to make the best recommendations.

One intuitive way to choose the lists is to split the data to train the dataset and test the dataset. Then, we traverse the train dataset and use every list to make a recommendation using a combination of the highest precision in the test dataset.

Another way to determine location categories in which users are most interested in is by extracting key words from an article. In TSFR, we use a TF-IDF based method to determine the location categories to make recommendations.

First, we use the Equation 5 to calculate the TF, which illustrates how important the category is to the user. Then, we use Equation 6 to calculate the IDF, which illustrates the category's relative importance to a user compared with other categories. In an information retrieval system, and the product of TF and IDF is used to extract key words of an article.

In our recommendation system, we use TF to multiply the power of IDF to get proper categories to make recommendations. In Equation 7, we traverse a range of power α of IDF on the train dataset to get proper power of IDF and use it on a test dataset to get categories to use and make recommendations with common users in them.

$$TF_{u,c'} = \frac{|\{u.v_i : v_i.c = c'\}|}{|u.V|} \tag{5}$$

$$IDF_{c'} = \log \frac{|U|}{|\{u_j : c' \in u_j.C\}|} \tag{6}$$

$$TF - IDF^* = TF * IDF^{\alpha} \tag{7}$$

where $|u.v_i : v_i.c = c'|$ is user u's number of check-ins in category c', $u.V$ is the total number of user u's check-ins, and $|u_j : c' \in u_j.C|$ is the number of users who have visited category c' among all the users U in the system.

IV. Experimental Evaluations

A. Experiment Setup

In order to evaluate the quality of TSFR, we run our system on a check-in dataset collected by Cho *et al.* [18]. They collected a total of 6,442,890 check-ins of 196,591 users over the period of Feb. 2009 - Oct. 2010 from Gowalla, a location-based social networking website. Friendships are undirected and there are 950,327 edges of these users. The check-in data is shown as Table I, and in order to protect the privacy of users, the user ID and venue ID have been anonymous. In our evaluation, we use the longitude and latitude to filter check-in data in New York City, and to solve the data sparsity problem, we pick the data with more than 10 check-ins per user to run our recommender system. The statistics of both datasets are shown in Table II.

Then, we use the Foursquare venues/search API to get the semantic information of the check-in locations in the dataset. In order to handle the problem that several places with different semantic information may use the same longitude and latitude, we pick the top 3 results when using the searching API with the longitude and latitude. Therefore, the dataset is like Table III now. The category information taken from Foursquare API constitutes a hierarchy system. Considering the data sparsity, we only use the first hierarchy category information, which are Arts & Entertainment, College & University, Event, Food, Nightlife Spot, Outdoor & Recreation, Professional & Other Places, Residence, Shop & Service, and Travel & Transport.

TABLE I
SAMPLE OF THE DATASET

User ID	Check-in Time	Latitude	Longitude	Venue ID
0	2010-10-19 23:55:27	30.235909	-97.795139	22847
0	2010-10-18 22:17:43	30.269102	-97.749395	420315
22	2010-10-01 17:02:14	34.017273	-118.447508	59838

TABLE II
STATISTICS OF THE DATASET

	Raw Dataset	New York	Data Filtered
Amount of users	196591	7112	2365
Amount of check-in	6442890	138690	121573
Check-in per user	33	20	51
Amount of categories	10	10	10

TABLE III
DATASET WITH SEMANTIC INFORMATION

User ID	Check-in Time	Venue Category
0	2010-10-19 23:55:27	Airport;Airport Terminal;Airport Lounge
0	2010-10-18 22:17:43	Burger Joint;Beer Bar;Neighborhood
22	2010-10-01 17:02:14	Office;Neighborhood;Dog Run

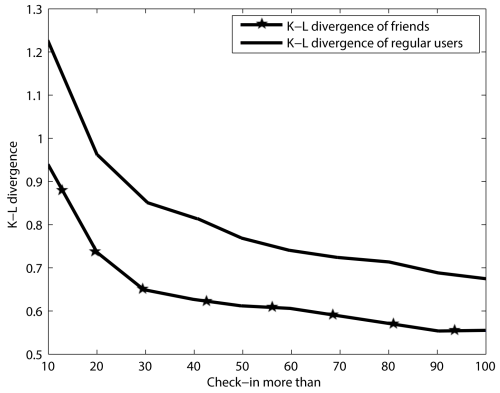

Fig. 3. Kullback-Leibler divergence of Friends and Regular Users

When calculating the K-L divergence between all of the users in New York, we divide one day into 24 equal time slots and calculate the K-L divergence between all of the users in New York. Then, we compare K-L divergence between friends and regular users. As figure 3 shows, there are significant differences between them. Then, we run TSFR and a classic collaborative filtering based friend recommender system on the filtered dataset. When we run TSFR, we split the dataset into the train dataset and the test dataset randomly, which takes up 2/3 and 1/3 of the filtered dataset each. Last, we compute the recall, precison, and F-Measure according to Equation 8, Equation 9, and Equation 10 of two systems.

$$\mathrm{R}ecall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{|R(u)|} \tag{8}$$

$$\mathrm{Pr}\,ecision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{|T(u)|} \tag{9}$$

where $R(u)$ represents the friend list of the dataset, and $T(u)$ represents the recommendation list of the recommender system.

$$F = \frac{2 * PR}{P + R} \tag{10}$$

where P represents the precision and R represents the recall of the recommender system.

Note that to avoid the data sparsity of users' friend lists when calculating the precision and recall of two systems, we only pick the users who have more than 5 friends and regard friends recommended with common friends of users as good recommendations.

B. Experimental Results

The task of the recommendation system is to recommend items that a particular user would be fond of. Recommending more popular or ranked items to users is an effective way that would gain high accuracy. But users can easily find the recommended products in many ways, even in the hot ranking of the home page. To reach the goal of personalization, we should find a list that contains the current users prefer and other users do not like them. A good recommendation system should be able to meet certain accuracy under the premise of diversity. So we use F-Measure to balance accuracy and diversity.

Figure 4, Figure 5 and Figure 6 show the average precision, recall and F-Measure of users who have more than 100 check-ins in New York. We run a classic collaborative filtering based recommender system on the check-in data with semantic information as a comparison.

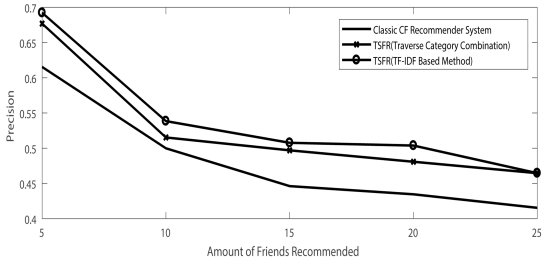

Fig. 4. Precision of Two Systems

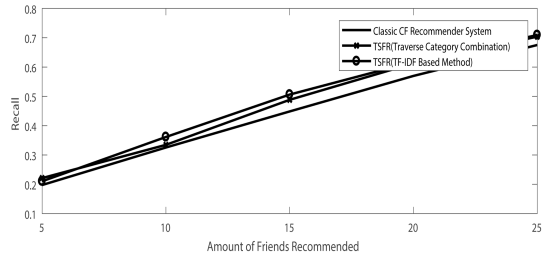

Fig. 5. Recall of Two Systems

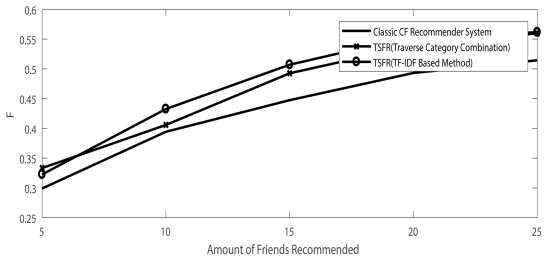

Fig. 6. F-Measure of Two Systems

The figures illustrate that using the TF-IDF based method or traversing the train dataset to choose the location categories both gain a significant advantage over the classic collaborative filtering based friends recommender system. The precision and recall vary with the number of users recommended; the

precision and recall of our scheme are about 10% higher than the classic collaborative filtering-based recommender system. Note that using the TF-IDF based method can supply a better explanation to users when presenting the recommendations to them. Then, we calculate the F-Measure of two recommender systems, which is often used to measure the performance of recommender systems. Figure 6 confirms the assumption we made that time information about check-in data plays an important role in analyzing users' check-in patterns and making friend recommendations.

V. Conclusion and Future Work

With the emergence and rapid development of network technology, the way of human communication is becoming more diverse. Internet application services are designed to help people build social networks. In this paper, we propose a friend recommender system based on the semantic and time information of users' check-in data in location-based online social networks. To handle the data sparsity of a conventional collaborative filtering recommender system, we divide the locations into categories based on the semantic information about the locations. In addition, we use the TF-IDF model in an information retrieval system to seek a balance between users' preferences and popularity of locations when quantifying users' check-in data. Then, we use K-L divergence to measure the differences between every two users' check-in time distribution, and transform them into an adjustable factor used in calculating their similarity. Last, we select top-n most similar users to make friend recommendations. To evaluate our recommender system, we run it on a dataset taken from Gowalla. The experimental results show that TSFR outperforms the system just by considering the semantic information of locations.

Although our recommender system outperforms the classic collaborative filtering-based friend recommender system, the precision and recall of both systems are at a low level. In our opinion, the reason is that using check-in data alone to make friend recommendations is a difficult task because there are many factors influencing the friendships of users. Additionally, check-in service has evolved to another form; there is an increasing number of social network applications encouraging users to attach their location to the content they share online. Therefore, the location information of users is suitable to assist in making friend recommendations. In the future, we will try to make use of these various forms of location information in social networks to make recommendations.

References

[1] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *CoRR*, vol. abs/1301.7363, 2013.

[2] N. Li and G. Chen, "Multi-layered friendship modeling for location-based mobile social networks," in *6th Annual International Conference on Mobile and Ubiquitous Systems:Computing, Networking and Services, Toronto, Canada,July 13-16*. IEEE, 2009, pp. 1–10.

[3] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[4] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, Pisa, Italy, July 17-21*, 2016, pp. 549–558.

[5] L. Qi, X. Xu, W. Dou, J. Yu, Z. Zhou, and X. Zhang, "Time-aware ioe service recommendation on sparse data," *Mobile Information Systems*, vol. 2016, pp. 4 397 061:1–4 397 061:12, 2016.

[6] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web (TWEB)*, vol. 5, no. 1, p. 5, 2011.

[7] J. Bao, Y. Zheng, and M. F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems, Redondo, Beach, CA, USA, November 7-9, 2012*, 2012, pp. 199–208.

[8] S. Bannur and O. Alonso, "Analyzing temporal characteristics of check-in data," in *Proceedings of IW3C2 23rd International World Wide Web Conference, Seoul, Republic of Korea, April 7-11,Companion Volume*, 2014, pp. 827–832.

[9] H. Gao, J. Tang, X. Hu, and H. Liu, "Exploring temporal effects for location recommendation on location-based social networks," in *Seventh ACM Conference on Recommender Systems, Hong Kong, China, October 12-16*, 2013, pp. 93–100.

[10] Q. Yuan, G. Cong, and A. Sun, "Graph-based point-of-interest recommendation with geographical and temporal influences," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, Shanghai, China,November 3-7*, 2014, pp. 659–668.

[11] C. Luo, X. Cai, and N. Chowdhury, "Self-training temporal dynamic collaborative filtering," in *Proceedings of ACM Advances in Knowledge Discovery and Data Mining - 18th Pacific-Asia Conference, Tainan, Taiwan, May 13-16*, 2014, pp. 461–472.

[12] X. Y. Shi, X. Luo, M. Shang, and X. Y. Cai, "Empirical analysis of collaborative filtering-based recommenders in temporally evolving systems," in *Proceedings of 14th IEEE International Conference on Networking, Sensing and Control,Calabria, Italy, May 16-18*, 2017, pp. 406–412.

[13] M. Ye, K. Janowicz, C. Mülligann, and W.-C. Lee, "What you are is when you are: the temporal dimension of feature types in location-based social networks," in *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Chicago, IL, USA,November 1-4*, 2011, pp. 102–111.

[14] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann, "Time-aware point-of-interest recommendation," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, Dublin, Ireland,July 28-August 01*, 2013, pp. 363–372.

[15] C. C. Tuan, C. F. Hung, and Z. Wu, "Collaborative location recommendations with dynamic time periods," *Pervasive and Mobile Computing*, vol. 35, pp. 1–14, 2017.

[16] X. Li, M. Jiang, H. Hong, and L. Liao, "A time-aware personalized point-of-interest recommendation via high-order tensor factorization," *ACM Transactions on Inf. Syst.*, vol. 35, no. 4, pp. 31:1–31:23, 2017.

[17] J. Ramos, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*, 2003.

[18] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, CA, USA, August 21-24*, 2011, pp. 1082–1090.