Local Experts Finding using User Comments in Location-based Social Networks

Jiuxin Cao[1], Yuntao Yang[2], Biwei Cao[3], Lingyun Xue[1], Shancang Li[4*], Muddesar Iqbal[5], Shahid Mumtaz[6]

1. Key Laboratory of Computer Network Technology of Jiangsu Province, School of Cyber Science and Engineering, Southeast University, Nanjing, China (Email: jx.cao@seu.edu.cn)
2. School of Computer Science and Engineering, Southeast University, Nanjing, China (Email: yuntao_yang@seu.edu.cn)
3. Australia National University, Canberra, Australia (Email: tarasom0804@gmail.com)
4. University of the West of England, Bristol, UK (Email: Shancang.Li@uwe.ac.uk).
5. London South Bank University, London, UK (Email: M.Iqbal@lsbu.ac.uk).
6. Instituto de Telecomunicações, Universitário de Santiago, Portugal(Email: smumtaz@av.it.pt).

ABSTRACT

The opinions of local experts in the location-based social network are of great significance to the collection and dissemination of local information. In this paper, we investigated in-depth how the user comments can be used to identify the local expert over social networks. We first illustrate the existences of potential local experts in a social network using a scored model by considering the personal profiles, comments, friend relationship, and location preferences. Then, a multi-dimensional model is proposed to evaluate the local expert candidates and a local expert discovery algorithm is proposed to identify local experts. Meanwhile, a scoring algorithm is proposed to train the weights in the model. Finally, an expert recommendation list can be given based on the score ranks of the candidates. Experimental results demonstrate that effectiveness of proposed model and algorithms.

KEYWORDS

Comments over Social Network, Scoring model, Local expert

1 Introduction

With the rapid development and a large amount of interesting applications, the social network is becoming a necessary platform for daily life and information sharing [1]. The popularization of social networks also makes it feasible for users to find solutions when facing problems. There are two commonly used ways to find solutions: the first is to search the content of social networks (such as microblogging, Twitter, etc.) which contains vast amount information; the second is asking for the help of the local experts online [1, 2, 3]. For the second way, the key problem is how to find who are local experts and which local expert can answer the user question precisely, which is also the research aims of this paper.

The local expert application will be a vital service in the location-based social network [1] (LBSN) such as *Yelp* review network and *Dianping* network [2], which incorporate online relationships and offline behaviors of users, bringing us a richer user experiences and attracting thousands of users at the same time. There will be a lot of needs when we visit a new place for the first time, such as food, shopping, etc. A relevant social network platform like *dianping.com* can only provide us with specific recommendation list. But most of the time, this recommendation cannot meet our real need when it becomes more complicated. For example, when a user arrives at *Xinjiekou* in *Nanjing* city for the first time going to a restaurant, the user might aims at finding a restaurant with local cuisine, nice environment, in a time and cost-effective way. In this case, a local expert might be very helpful for providing recommendations for the user online, who is familiar about not only the topic but also the location, thus providing a higher quality service. Compared to general topic experts, except the topic attribute, the local experts are also based on geographical location, so the research problem turns to find the experts from the general topic expert set, which are well similar to the special position.

The works in [2] shown that first coming to a strange place, people are more willing to consult local experts. Local experts are always the best choice when querying the best of some local businesses on Yelp. They play an important role in the local information collection and communication. Yelp has a huge dataset including a large number of user information, reviews and spatial data which can provide wealthy resources for the discovery of local experts,

The existing expert finding methods mainly focus on specific topic experts rather than local experts, so that the location and semantic information are not being fully utilized. In addition, the study of expert identification and recommendation based on fine granularity like a location points with latitude and longitude is rarer. The main contributions in this work is: (1)We propose a review-based local expert discovery mechanism to measure the degree of local experts from different aspects; (2) A multi-dimensional model is proposed to evaluate the local expert candidates and a local expert discovery algorithm is proposed to identify local experts; (3) A scoring algorithm is proposed to train the weights in the model.

The structure of this paper is organized as follows: Section 2 introduces related works of expert finding. Section 3 gives the description of the experimental data and the definition of the problem. Meanwhile, this section analyzes the feasibility of the expert discovery in the Yelp network; Section 4 details the review based local expert scoring model; Section 5 presents the learning method of the scoring model; Section 6 gives the design and result of the experiment. At last, conclusions and future works are provided in Section 7.

2 Related Works

Social networking research began in the nineties of the twentieth century, since then experts finding research has gained some success. Early experts finding mainly uses information retrieval techniques [3], mining experts who meet the requirements using specific terms. More and more research methods came into being with the in-depth research of expert. The main expert identification methods can be divided into three categories: *1) Probability-model-based experts finding*. It is mainly based on the probability statistics model, which solves the expert users' ranking by calculating the probability that users are experts. There are two classic expert discovery models [4]. One is the profile-centric method, which measures the correlation degree between the profiles created for users and the queries; the other one is the document-centric method, which ranks the experts and the documents related to the query. These co-occurrence relations based models had achieved a good effect. But with the diversification of expert activities and relationship network, this single word-based and document-based expert identification method is no longer applicable to this kind of complex network relationships and unstructured textual information; *2) Graph-model-based experts finding,* which is influenced by page sorting algorithm. The join of the relationship between users makes experts identification get a further developing. The classic sorting algorithms such as PageRank [6], Hyperlink-Induced Topic Search (HITS) [5], and a series of PageRank based algorithms like TwitterRank [5] and InfluenceRank [6] *etc,* are gradually used to find the expert users in the networks; *3) Topic-model-based experts finding* gives experts ranking by analyzing the relationship between experts and implied topics [7].

As the LBSN appears, more and more social content has location information, which improves customer satisfaction and at the same time provides opportunities for researchers. So local experts discovery has become a real need. *Antin* et al. [2] conducted a survey about people's attitudes towards local knowledge and found that most people feel they are local experts and are willing to be consulted local issues to give advice. The study shows that 43% of people are more willing to ask for local experts, and 39% of people do not mind answering questions. Cheng et al. [8] crawled tagged expert users, tags and the relations between them in Twitter dataset and a local experts identification algorithm was proposed to find local experts on different topics in different cities. The proposed LocalRank algorithm includes two aspects of local experts: topic authority and location authority. The topic authority is defined as how well the candidate is recognized about the topic, considering about user's link relation and the information disseminated in the network and proposing a distance weighted social graph to identify expertise level of users in a given topic. The location authority is defined as the local reputation of the candidate. *Haokai* Lu [9] studied personal expert recommendations on the same dataset, using matrix decomposition model from different aspects (location preference, topic preference, social relations preference) of users to recommend personal experts. *Tanvi* Jindal [10] used Yelp dataset to study local experts which extracts users' features firstly and then used classification algorithm like Bayesian to mining category experts. Based on the algorithm, a Gaussian mixture model is proposed to cluster the review locations. Then the distance between the cluster center and the query location is used to estimate the location authority. However, this algorithm ignores the network topology and the large and abundant content information and the cluster center in a city cannot fully reflect a user's active points. Wei Niu [11] [12] et al. proposed a local expert sorting algorithm named LExL [12], using Microsoft's famous LambdaMART [14] algorithm in "Learning to rank" [13] from four dimensions: user's own attributes, tag table, location authority and location-based random walk to sort candidate users.

Although many research works have been done in this area, there are still many challenges, including local expert finding with the abundant location and text semantic information. The classical expert discovery method has been far from satisfying in the location-based social networking environment. Relevant research in the follow-up study also puts forward some different approaches. This paper proposes a review based local expert discovery algorithm in a fine granularity, making fully use of various kinds of information, mining local experts in the Yelp network for high-quality expert services.

3 Preliminary

In this section, we will define the research problem and address the dataset used in this paper. Afterwards combining analysis on spatial point pattern of the dataset and two instances, the feasibility of the research is verified. Finally, the POIs are abstracted to simplify calculation and local expert candidates of each POI are marked among existing category experts in the dataset using DBSCAN clustering algorithm.

3.1 Problem Description

Compared to general experts, local experts are not only experienced in the area, but active in some locations. When users visit a new place, having their own demands, the question is which person who can offer high qualified service for users should be recommended. The people which we need are called local experts, formally described as follows: Given a review query q, which contains category $c(q)$, location $l(q)$, the task is finding k local experts who are familiar with the location and also good at query category.

3.2 Data Description

The experiment dataset used in this paper is from the public Yelp dataset, which is available on the Yelp website. Yelp website is a typical LBSN platform including business about restaurants, hotels, tourism, shopping and other areas. More than 33 million users' monthly access and rich review information can offer a good data base to this study. The original data set used in this article includes three categories:

- **Business information**: business ID, business location with latitude and longitude, business categories, business city, business ratings, business review times, etc.
- **User information**: user ID, user review counts, creation date of user account, user's friend list, number of fans, expert tag, average score of user reviews, etc.
- **Review information**: review ID, review user ID, review business ID, score, date, text, number of useful votes, etc.

In this section, two basic statistical information of dataset are analyzed and shown as follows: Figure 1 shows the frequency distribution of number of users' 'friend' and Figure 2 presents the frequency distribution of the number of reviews number published by users. The distribution graph is represented with the double

logarithmic coordinate system, where the abscissa is the number of indicators and the ordinate indicates the statistics under that number. It can be seen from both Figure 1 and Figure 2 that the basic feature of a user in the network obeys power-law distribution, which means that minority users may have larger structural feature values and influence, reflecting the power-law characteristics of nodes in *Yelp*.

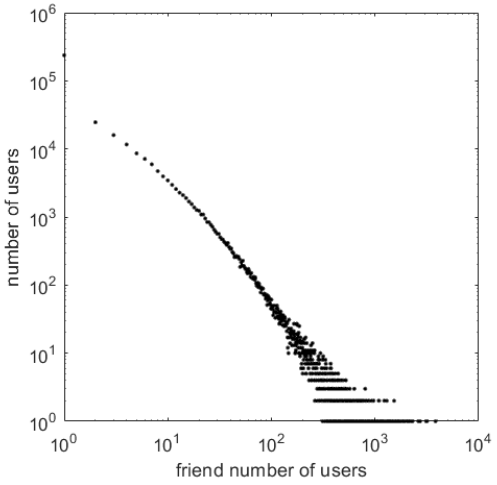


Figure 1 Frequency distribution of friend accounts

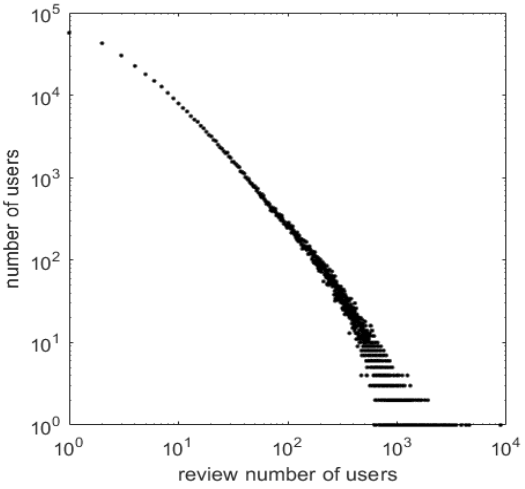


Figure 2 Frequency distribution of review accounts

3.3 Features Analysis

Category and location of user reviews are two important features which are widely used in the model that can be used to evaluate the probability of a user can be a local expert. The categories used are set by Yelp and statistical analysis of *Yelp* dataset is conducted to discover the distribution of users in these two aspects.

3.3.1 Spatial Point Pattern Analysis. The spatial point model proposed in [15] is based on the distribution of all observation points on the map, which can be used to analyze the spatial distribution patterns of discrete geographic objects or event points according to their spatial position. The point pattern distribution can generally be divided into three basic types: *aggregation*, *random distribution* and *even distribution*. In this work, we apply the spatial point pattern of user reviews in Yelp network to analyze the spatial distribution of user reviews.

Two analytical methods are commonly used for point-space model analysis: *density-based* and *distance-based* methods. The density-based methods study spatial patterns using the features of point density distribution. The distance-based methods are generally used to measure the nearest neighbor distance such as *Nearest Neighbor Index* (NNI), which is a complicated tool to precisely measure the spatial distribution of a patter [15]. In this work, we use the NNI method to analyze the spatial pattern of user reviews.

In the NNI method, the nearest-neighbor distance of any point is firstly calculated, then the mean of all these nearest neighbor distances is taken as the evaluation index of the model distribution. For the same dataset, the NNIs are different under various distribution patterns. Compared to the NNI results of the distribution of all user review points in the dataset and that of the complete random mode which equals to 1, the type of the distribution model can be augmented [15]. The NNI can be described

$$NNI = \frac{\overline{d_{min}}}{2\sqrt{n/s}} \quad (1)$$

where n is the number of points, $\overline{d_{min}} = \frac{1}{n}\sum_{i=1}^{n} d_{min}(i)$ is the average nearest neighbor distance of all points where $d_{min}(i)$ is the distance from the point i to the nearest neighbor of it, and s is the specific spatial area all the points reside in. According to [15], in the aggregation mode the distance between the points is short because of the spatially clustered points, so the NNI is less than 1 [15]. In the even distribution mode, the distance between two points is larger and the NNI is greater than 1. Therefore, the spatial distribution pattern of review locations can be evaluated and augmented by NNI. Figure 3 shows the distribution of NNI of all users in Las Vegas whose review counts is more than 20.

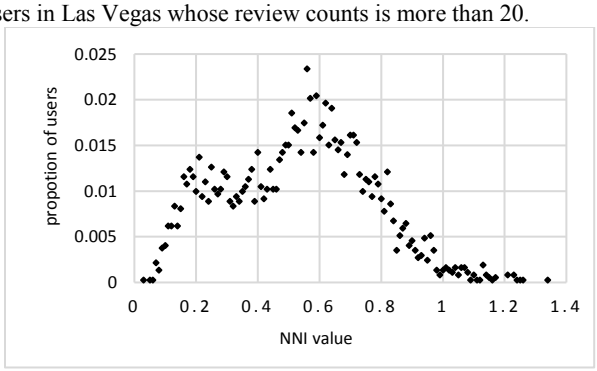
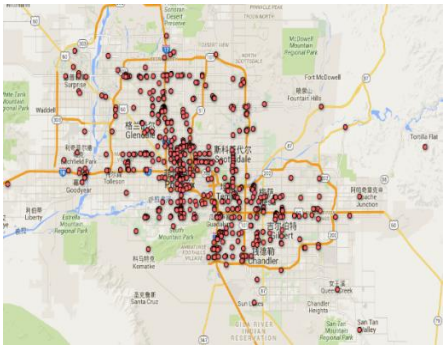


Figure 3 NNI distribution of users

From Figure 3, it can be summarized that the NNIs of most users are less than 1, indicating that a large number of reviews points of each user in *Yelp* are spatially close to each other, which means the majority of user review distribution in the network is spatially aggregate.

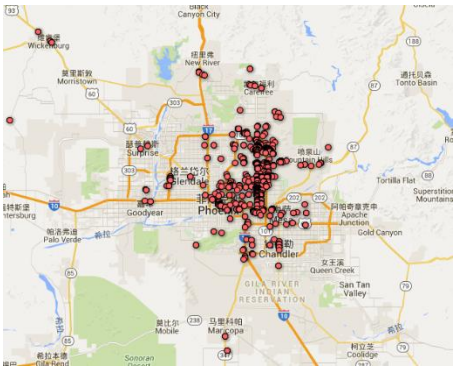
3.3.2 Instance Analysis. Two users are randomly selected to analyze their review locations on the map. Figure 4 and 5 show the review distributions of user A and B, respectively. It can be seen that user A is more active in Phoenix and user B often comments in both Las Vegas and Phoenix, which means user A has position authority in Phoenix, while user B has position authority in both.

(a) The location distribution of user A in Las Vegas

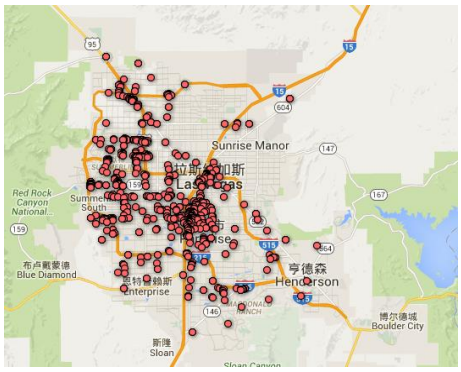


(b) The location distribution of user A in Phoenix

Figure 4: The location distribution of user A



(a) The location distribution of user A in Las Vegas



(b) The location distribution of user A in Phoenix

Figure 5 The location distribution of user B

Category features of users are also analyzed. In view of a large number of categories in *Yelp*, some common categories are selected to analysis the category features of two randomly selected users.

The statistics result of two users in 12 categories are shown in Figure 6 and Figure 7, respectively. It can be seen that user A is more adept at entertainment and user B reviews most at eating. Obviously, user reviews always focus on some certain categories, which is in line with the habits of user behavior. Such as if you need a food local expert in Phoenix, user B is more appropriate and authoritative than user A.

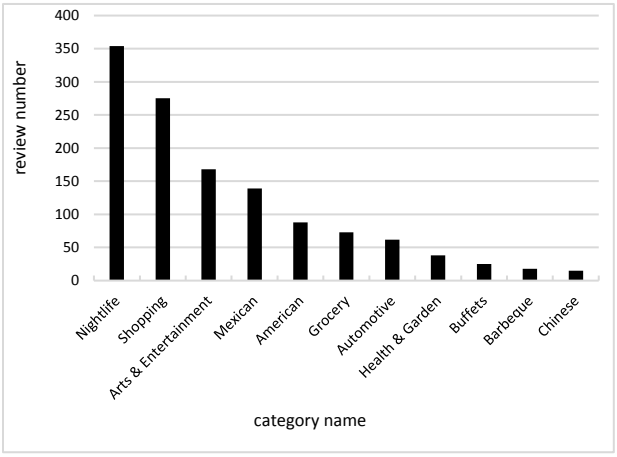


Figure 6 The category distribution of user A

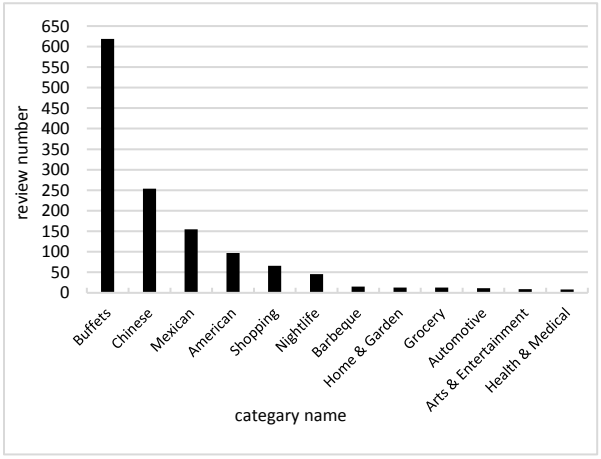


Figure 7 The category distribution of user B

3.4 POI Abstracting and Candidate Marking

3.4.1 POI Abstracting. In this work, we assume that the identification and excavation of local experts is based on a specific location. It is impossible to cover all geographical points in an area, so POIs are abstracted in the city assuming that all queries are based on POIs.

However, for the given city, the number of POIs in the city cannot be predicted. Therefore, it is necessary to use density-based clustering algorithm that does not need to determine the number of clusters. In this work, a typical density-based DBSCAN clustering

algorithm is used to select POIs. The two parameters needed by the algorithm are the neighborhood radius e, and the minimum number of MinPts, which is the smallest number of object points in a neighborhood.

The parameters in the algorithm are selected by experiment where the object points are user review location points in the city. The radius e is users' mostly frequent travel distance. According to above results of NNI in Figure 3, the location distributions of most are aggregate. As a result, people have their own geographically active location range, and they always visit the place not that far [19]. To match the user activity rule in the city, the most active distance of users is chosen as the radius. Sorting all user reviews in temporal order as shown in Figure 8, users in the city often travel within the range of 10 km between adjacent reviews [19]. In this work, we set the value of the radius as 10km.

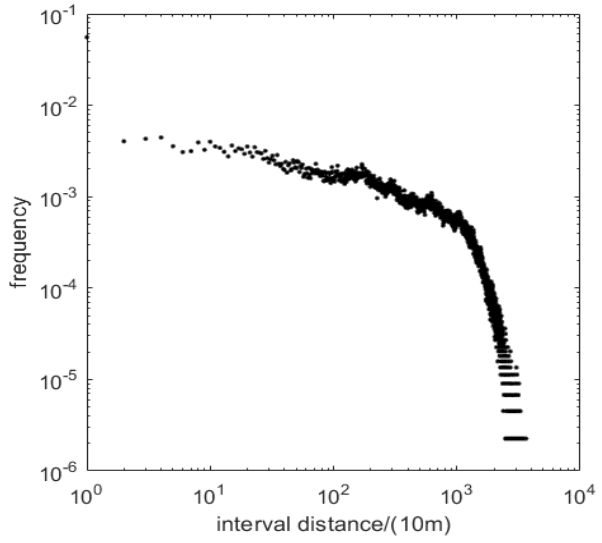


Figure 8 The interval distance distribution of user adjacent reviews

In city area, all users review locations can be clustered using DBSCAN. In view of the MinPts selection experiment, the minimum object point numbers are selected from 20, increased by 5 one step, and then the final experiment cluster numbers and cluster center points distribution on the map are stable when MinPts reach a certain number.

3.4.2 Candidate Marking. Though there are some tagged category experts in the dataset, it is still necessary to identify the local expert candidates by considering the category and location features in the meantime to satisfy user's local expert queries when going to a new place.

The local experts are based one or more active location points of their own. Firstly, the DBSCAN clustering algorithm is processed to determine the central location points of expert candidates in dataset, and then mark local experts whose central points are within the radius of POIs. The algorithm is described as follows:

Algorithm 1 local expert candidate marking algorithm

input: All pair of candidates and their review points set C_i: $\mathrm{U}(u_i, C_i)$, POIs set L

output: All pair of POIs and their local experts *LE*

1. FOREACH $(u_i, C_i) \in$ U
2. c_i = DBSCAN(C_i)// Calculate the central location points of user I using DBSCAN algorithm
3. ADD (u_i, c_i) to UC//UC is all pair of candidates and their central location point
4. END-FOR
5. FOREACH $l_i \in L$ Do
6. FOREACH $(u_i, c_i) \in$ UC Do
7. IF $\mathrm{distance}(c_i, l_i) <$ r
8. ADD u_i, l_i TO *LE*
9. END-FOR
10. END-FOR
11. RETURN *LE*

4 Proposed Method

In order to reduce computational complexity and alleviate the impact of sparsity, all users in the dataset this work are a collection of users who have commented on a given category over a threshold set as 20 in a given city area. Then the candidate set is extracted through the business locations of these users' reviews. Finally, friend relationship edges are added to the candidate set.

In this section, we first introduce the local experts scoring model, and then address the local expert's assessment indicators. And then, based on the scoring model, a local expert discovery algorithm will be presented.

4.1 Scoring Model

The scoring model is designed based on reviewer features in four aspects, including *personal*, *review*, *friendship* and *location authority attributes*, in which the scoring vector sets of all users are denoted as Ψ^P, Ψ^R, Ψ^F and Ψ^L respectively. Among them, Ψ^P consists of two elements, the structural attribute scoring vector Ψ_1 and the influence scoring vector Ψ_2. Ψ^R contains three elements which are valid text number scoring vector Ψ_3, review semantic scoring vector of users Ψ_4 and semantic scoring vector of reviews Ψ_5, and Ψ^F only has single element which is friendship scoring vector Ψ_6. Ψ^L includes two elements, review number scoring vector Ψ_7 and review centroid scoring vector Ψ_8. Then linear model is utilized to construct the scoring model for measuring local expert candidate's ranking on the given query. Assuming that $\Psi = \Psi^P \cup \Psi^R \cup \Psi^F \cup \Psi^L = \{\Psi_1, \Psi_2, \dots, \Psi_8\}$ and for candidate u, 8 corresponding scores in Ψ on location l is donated as vector $\varphi(l,u) = \{\varphi_1, \varphi_2, \dots, \varphi_8\}$, the final score of user u at point l is defined as follow:

$$r(l,u) = \theta \cdot \varphi(l,u) = \sum_{k=1}^{8} \theta_k * \varphi_k \quad (2)$$

where $\theta = \{\theta_1, \theta_2, \dots, \theta_8\}$ is the weight vector for the score vector and θ_k represents the weight for the k^{th} element score, i.e. the importance degree of the score to the total results. The weight vector θ in the scoring model needs to be learned through data training, which will be introduced in the following section and local

expert recommendation will be carried out according to the rank of final score.

4.2 Scoring Local Expert Candidate

There are two main parts of the local expert's assessment: (1) Category authority of candidates, which estimates the level of candidates under given categories through taking full account of their own attributes, review semantic preferences and network structure attributes, corresponding to the previous personal, review and friendship authority attributes. (2) Location authority of candidates. The review location information contains location preferences of candidates. The above two parts are comprehensively used to identify local experts and then to achieve the purpose of enhancing the recommended results of local experts.

4.2.1 Scoring Personal Attributes. Similar to general LBSN network, Yelp users also have their own friends and fans. Besides, the network has specific feedback mechanism for user reviews that will receive useful, funny, and cool votes. By analyzing feedback votes received, the audience level of the user reviews can be measured. The personal attributes evaluated in this article are divided into two aspects: static and dynamic. The static attributes are structural characteristics of user and the dynamic one is user's influence in the network.

(1) Structural attributes: Yelp is a network for review. The review content of user is valuable for others, and users who has high-quality reviews may have relatively more friends and fans. Social network topology structures and user's own characteristics are integrated to obtain candidates' eigenvalues, and the average normalized score of a user is used to measure structural feature, denoted as follows:

$$\varphi_1 = (u_{friends} + u_{fans})/2 \tag{3}$$

Where $u_{friends}$ is the normalized score of user's friends, and u_{fans} is the normalized score of user's fans.

(2) Influence: the influence is a significate indicator measuring the degree of local experts. The greater the impact of a candidate is, the higher the degree of expert level is. The user's score of their influence φ_2 in this paper is mainly attributed to the following two aspects:

Activity: the valid review numbers released by candidate during a period of time;

Authority: the useful vote number of candidate reviews.

After normalization, candidate's influence score is presented by the average value. For the candidate set, the scoring vector sets of structural characteristics and influence are calculated and denoted as Ψ^P.

4.2.2 Scoring Reviews. Yelp network has a large number of review texts of specific categories. The text contains a lot of information which can be used to evaluate expert degree of candidates. In this paper, LDA (Latent Dirichlet Allocation) model is used to carry out semantic analysis of text content. All the texts are put into the same semantic space to construct the topic model and measured from the following three aspects:

(1) Valid text number

Valid text is the valid review text for a given category, having immense reference value to new users. People having adequate valid texts shows their frequent activities and more meaningful reviews on that category. Getting valid texts of review needs the following steps: 1) remove the stop words; 2) remove the punctuations; 3) handle the stem; 4) remove the low frequency words. The valid text number of a user is donated as φ_3.

(2) Review semantic score for a user

In this paper, all of a user's reviews are merged as a user description document and the reviews of all users under a category are merged as a category description document. Local experts are excavated in the massive dataset by measuring the semantic similarity between user description documents and category description documents. If a user's reviews often appear in the same category, the user's document semantic vector distribution will be more inclined to the category semantic vector. Suppose that the topic vectors extracted of user u is θ_u and that of category c is θ_c. Then the cosine similarity is used to calculate the similarity of two topic vectors. The formula is as follows:

$$\varphi_4 = cos(\theta_u, \theta_c) = \frac{\theta_u \cdot \theta_c}{|\theta_u||\theta_c|} \tag{4}$$

(3) Semantic score for a review

The study [16] has showed that all reviews of a business can describe the business information. On this basis, if the text semantic information of a review on a business is in line with business description, this review will be more authoritative and reliable. Likewise, the average similarity degree between the reviews of user u under the given category c and the business corresponding to the certain review b is calculated by cosine similarity to evaluate the credibility of a user's review in the given category, donated as φ_5 and the formula is as follows:

$$\varphi_4 = \frac{\sum_{i \in c} cos(\theta_i, \theta_{b_i})}{n} \tag{5}$$

It is necessary to determine topic numbers when using LDA model. Different topic numbers can affect the model effect, directly having influence on the result of semantic similarity. In this paper, the topic number is set through multiple experiments and the perplexity is used to define the most appropriate topic numbers. The smaller the perplexity is, the better the model effect is. The perplexity formula is as follows:

$$P = exp(-\frac{\sum_{m=1}^{M} \sum_{n=1}^{N_m} log(p(w_{m,n}|\Omega))}{\sum_{m=1}^{M} N_m}) \tag{6}$$

Where M represents the number of semantic space documents, N_m is words number in m^{th} document, $w_{m,n}$is the n^{th} word of the m^{th} document, p is probability model learned from data set and Ω is model parameter value.

Based on the experimental results on shopping category shown as Figure 9, the model has the least perplexity when the topic number is 30. Therefore, in this paper topic number used in expert discovery on shopping category is 30.

For all candidates, the review semantic scoring vector is quantified by the above ratings, denoted as Ψ^R.

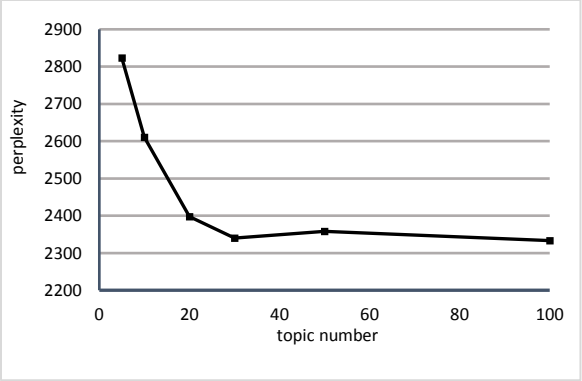


Figure 9 The perplexity distribution of different topics

4.2.3 Scoring Friendship. Expert finding is similar to the authoritative ranking of the webpage [20, 21, 22]. The larger the user's friend number is, the larger the influence of the user will be. However, users have different influence under different topics. The traditional PageRank algorithm is topic-independent web link algorithm which will mistakenly give a high degree of value to a number of web pages that are unrelated to the topic, resulting in serious topic drift phenomenon. The improved PageRank algorithm in this paper can not only restrain topic drift but effectively improve the effect of expert finding.

Users with similar interests will be friends in the network, which is called as homogeneity [17]. The homogeneity indicates that a user is not casual when choosing a friend. There are more similarities between users who have friend relationship than the ones who do not have. The traditional PageRank algorithm simulates a web surfer who transfers to any other link web page at the probability of $1/k$ when the current page has k out links. In order to solve the problem of termination and trap, the algorithm also has a jump probability of $1/n$, which means users may skip at any other page at the probability of $1/n$. It defines row vector of PageRank value of every page, transfer probability matrix, jump probability and damping factor which is usually set as 0.85, then calculate iteratively until row vector of PageRank value converges, where n is the number of iterations. In view of the problem described in this article, the transfer probability and jump probability is redefined. In YELP network, user review contents and user review positions are important for users to select the business, based on which the edge probability $E_c(i,j)$ is defined as follows:

$$E_c(i,j) = \begin{cases} \frac{|N_j|}{\sum_{k\in Friends(i)}|N_k|} * sim_c(i,j) * \frac{d_r}{distance(i,j)+d_r}, \\ (edge\ between\ i\ and\ j) \\ 0, (no\ edge\ between\ i\ and\ j) \end{cases} \quad (7)$$

where N_j is the review number of user j, $sim_c(i,j)$ is the similarity between user i and j in topic c, and $distance(i,j)$ is the distance between two users' review center location, d_r is to reduce the impact of distance on results. In equation (7) if there is an edge between i and j, edge probability $E_c(i,j)$ depends on three aspects: 1) $\frac{|N_j|}{\sum_{k\in Friends(i)}|N_k|}$: Compared to all the other friends of the user i, the impact degree of j on it. More information a user receives from a friend, the more affection this user gets. 2) $sim_c(i,j)$: In a given category, the similarity between two users' semantic vectors is calculated and more similar the vectors are, the greater the influence is. 3) $\frac{d_r}{distance(i,j)+d_r}$: the distance between the most active positions of two users. Users can easily be affected by the ones who are geographical close to them. The transfer probability T_c is the normalized edge probability E_c between two users.

The users having a large number of reviews will be more easily selected when choosing an expert on one category, so we redefine the jump probability in Eq.(8), in which both the number of reviews and their categories are taken into account.

$$M_c = \frac{review\ number\ on\ the\ category\ of\ the\ user}{number\ of\ all\ review\ on\ the\ category} \quad (8)$$

Therefore, the authority of candidates based on their friend relationships according to traditional PageRank algorithm is measured as follows, here the damping factor $q = 0.85$:

$$\overrightarrow{\varphi_f^{(k+1)}} = (1-q) * M_c + qT_c * \overrightarrow{\varphi_f^{(k)}} \quad (9)$$

where $\overrightarrow{\varphi_f^{(k)}}$ is the row vector of users' authority analogous to PageRank value after k iterations. The scoring vector of users' friendship evaluated by the improved PageRank algorithm proposed is denoted as Ψ^F.

4.2.4 Scoring Location Authority. It is another tough work to add location information into the expert assessment. Existing researches have achieved some results which consider about the distance from the query point to user's home or work location to measure the location authority. But the ignorance of the users' active points except the workplaces or home will undoubtedly affect the final result. Therefore, this paper proposes a new method for the location authority.

The participation time and the spatial distribution of candidate review locations are both important. Therefore, inspirited from the centroid calculation formula of the irregular objects in Physics shown in equation (10), similar method is proposed to evaluate spatial distribution of candidates.

$$r_{centroid} = \frac{\sum_i^n m_i r_i}{\sum_i^n m_i} \quad (10)$$

In equation (10), assuming that the object consists of n particles, m_i represents the quality of particle i and r_i indicates the diameter vector of particle i relative to a fixed point in the particle coordinate system. $\sum_i^n m_i$ is the total quality of the object so the diameter vector of centroid $r_{centroid}$ can be calculated.

The user activity range radius is set according to the frequent travel distance in Figure 8, which means for the query point, only the user review location within the query radius is considered. The review number in the scope of the query point and the distance between candidate's reviews centroid and the query point is used to estimate location authority, donated as φ_7 and φ_8.

As for φ_8, analogous to the calculation of centroid in Physics, centroid calculation formula of user review locations is presented as follows:

$$revcen_u = \frac{\sum_{u_{l_j}\in U_O(u)}\left(l\left(u_{l_j}\right)-O\right)*\left|u_{l_j}\right|}{|U_l(u)|} \quad (11)$$

Where l is the location set within the 10 km radius of the query point according to 3.4.1 and u_{l_j} represents reviews of user u at location l_j, whose absolute value indicates review numbers at that location and is equivalent to the quality, $U_l(u)$ represents the user's review set of the location set l, which can be regarded as the total quality and as φ_7, O is the coordinates of the query point which is equivalent to the fixed point and $l(u_{l_j})$ is that of the location l_j hence $\left(l\left(u_{l_j}\right) - O\right)$ represents the vector from query point to the location l_j. Therefore, the review centroid relative to the query point can be calculated.

Based on a specific location, the evaluation of candidates not only consider their active level in the scope of the query point, but also consider their familiarities with the query point, recorded as Ψ^L.

4.3 Local Expert Discovery Algorithm

Local expert discovery algorithm is conceived based on the score quantification method and local expert scoring model in the above sections, shown as following steps: (1) Determining the candidate set and the weight vector θ; (2) Calculating scoring vectors of candidates using the above scoring algorithm, as φ; (3) Calculating final score r according to the weight vector θ and the scoring vector φ; (4) Ranking candidates based on the final scores. The specific algorithm is shown as below:

Algorithm 2 Review based local expert discovery algorithm

Input: Scoring Set $\Psi = \Psi^P \cup \Psi^R \cup \Psi^F \cup \Psi^L$, YELP network N, Query POI l_0, Weight Vector θ

Output: Local Experts $Top - N$ List U'

$U = selectCandidateUserSet(l_0)$
FOREACH $u_i \in U$ Do
  New Array φ //store the scoring vectors
  FOREACH $\Psi_j \in \Psi$ Do
    IF ($\Psi_j \in \Psi^P$)
      $\varphi[j] = computeProfileValue(l_0, u_i, \Psi_j)$
    ELSE IF ($\Psi_j \in \Psi^R$)
      $\varphi[j] = computeReviewValue(l_0, u_i, \Psi_j)$
    ELSE IF ($\Psi_j \in \Psi^F$)
      $\varphi[j] = computeFriendsValue(l_0, u_i, \Psi_j)$
    ELSE IF ($\Psi_j \in \Psi^L$)
      $\varphi[j] = computeLocationValue(l_0, u_i, \Psi_j)$
  END-FOR
  $r(l_0, u_i) = \theta * v = \sum_{k=1}^{n} \theta_k * \varphi[k]$
END-FOR
$U' = Top - N(U, r)$
RETURN U'

5 Model Learning

Implicit feedback data is used to learn the weight vector in this section, and the optimization objective function is defined. To maximize the function, the gradient rise method is used to estimate the weight vector.

5.1 Optimization Objective

For the scoring model, the features' weight represents the importance degree to scoring. The implicit feedback [18] is used as training data to learn the weight vector. Different from explicit feedback, implicit feedback only represents the interaction between candidate and location point. Traditional parameter learning methods based on classifier or score loss function optimization can't work well for implicit feedback. In this paper, a method based on maximum likelihood estimation is used to learn the weight vector. The goal of the learning is to optimize the rank of all candidate-pairs for POIs, which means the marked local experts should be ranked before ones having no marks. According to this idea, we define the Bayesian formulation of the optimization criterion, which is to maximize the posterior probability as below:

$$p(\theta|R) \propto p(R|\theta)p(\theta) \tag{12}$$

where θ is the weight vector, R represents the set of all candidate-pairs with right order for all POIs.

For calculability, we assume that candidates are independent from each other and the POIs are also independent. According to the assumption, $p(R|\theta)$ can be rewritten as below:

$$\begin{aligned} p(R|\theta) &= \prod_{l \in L} p(R_l|\theta) \\ &= \prod_{l \in L} \prod_{(u_i > u_j) \in R_l} p\left(u_i >_l u_j|\theta\right) \end{aligned} \tag{13}$$

where R_l represents the set of all candidate-pairs with right order for POI l, $p\left(u_i >_l u_j|\theta\right)$ represents the probability of candidate u_i ranked before candidate u_j for POI l, which is defined as below:

$$p\left(u_i >_l u_j|\theta\right) = \sigma(r(l, u_i) - r(l, u_j)) \tag{14}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$. In order to reduce the number of hyper-parameters, let $p(\theta)$ denote as a Gaussian distribution with a mean of 0 and $\sum_\theta = \lambda I$. According to the definitions above, we can derive the final optimization objective function as below:

$$\begin{aligned} OF &= ln\left(p(R|\theta)\right) = ln\left(p(R|\theta)p(\theta)\right) \\ &= \prod_{l \in L} \sum_{(u_i > u_j) \in R_l} ln\, \sigma\left(r_{i,j}^l\right) - \lambda \|\theta\|_2^2 \end{aligned} \tag{15}$$

where $r_{i,j}^l = r(l, u_i) - r(l, u_j)$, λ is the coefficient of regularization term.

According to implicit feedback data, the weight vector θ in the scoring model can be calculated through maximizing the objective function OF.

5.2 Parameter learning

As the optimization objective function is differentiable and need to be maximized, the gradient rise method can be used to estimate the weight vector. The gradient in each iteration when using standard gradient rise method can be calculated as below:

$$\begin{aligned} \frac{\partial OF}{\partial \theta} &= \frac{\partial}{\partial \theta}\left(\sum_{l \in L} \sum_{(u_i > u_j) \in R_l} ln\, \sigma\left(r_{i,j}^l\right) - \lambda \|\theta\|_2^2\right) \\ &= \sum_{l \in L} \sum_{(u_i > u_j) \in R_l} \frac{\partial}{\partial \theta} ln\, \sigma\left(r_{i,j}^l\right) - \lambda \frac{\partial}{\partial \theta} \|\theta\|_2^2 \end{aligned}$$

$$\propto \sum_{l\in L}\sum_{(u_i>u_j)\in R_l}\frac{e^{-r_{i,j}^l}}{1+e^{-r_{i,j}^l}}\cdot\frac{\partial}{\partial\theta}r_{i,j}^l-\lambda\theta \quad (16)$$

From the Eq. (14), it can be seen that there is too much calculation in each iteration using the standard gradient descent, so we employ the SGD (Stochastic Gradient Rise) method to deal with the estimation. In every iteration of the learning process, only one candidate-pair of a POI is randomly extracted from the training set to update the weight vector, as Eq. (15):

$$\theta=\theta+\alpha\left(\frac{e^{-r_{i,j}^l}}{1+e^{-r_{i,j}^l}}\cdot\frac{\partial}{\partial\theta}r_{i,j}^l-\lambda\theta\right) \quad (17)$$

where α is the learning rate, which controls the convergence speed of the learning process, and λ can control the training effect of the whole model. Under the premise of limiting the number of convergence iterations, the specific λ can match the appropriate α, which will be determined through the experiment below.

6 Experiments

In this section, experiment is designed and carried out. The metrics of precision and recall are chosen to evaluate the model effect and several local expert finding methods are compared with proposed method to validate the effectiveness of the approach.

6.1 Experiment Design

6.1.1 Evaluation Metrics. In this paper, the metrics of precision and recall are chosen which are often used to evaluate the effectiveness of recommendation method. The metric of precision reflects the accuracy of the recommendation model, which means the proportion of the correct experts in the recommended list accounting for the recommended experts, and is defined as Eq. (16). The recall metric reflects the comprehensiveness of the recommendation model, which means the proportion of the correct experts in the recommended list accounting for all correct local experts in dataset, and is defined as Eq. (17).

$$Precision=\frac{\sum_{l\in L_{test}}|R(l)\cap T(l)|}{\sum_{l\in L_{test}}|R(l)|} \quad (18)$$

$$Recall=\frac{\sum_{l\in L_{test}}|R(l)\cap T(l)|}{\sum_{l\in L_{test}}|T(l)|} \quad (19)$$

where L_{test} denotes the set of POIs in the training set, $R(l)$ is the set of local experts which is the result calculated by the algorithm of this paper, and $T(l)$ is the set of local experts which is marked as the ground truth.

6.1.2 Comparison Methods. To validate the effectiveness of our approach, the proposed method is compared with several local expert finding methods. The methods involved in the experiment are described as follows:

LER : The method proposed in this paper.

LocalRank: The topic and location authority comprehensive algorithm proposed in references [8]. The formula is as follows:

$$s(v_i,q)=s_l\big(l(v_i),l(q)\big)*s_t\big(t(v_i),t(q)\big) \quad (20)$$

where $s_l\big(l(v_i),l(q)\big)$ represents the location authority of candidate v_i at the query location $l(q)$, and $s_t\big(t(v_i),t(q)\big)$ is the category authority in the query category $t(q)$. The algorithm ranks the final results by multiplying.

PageRank & PB : The classic PageRank based algorithm, recommending candidates whose home position are closer to the query point, shown as follows:

$$s(v_i,q)=\alpha\cdot s_l'\big(l(v_i),l(q)\big)+\beta\cdot s_t'\big(t(v_i),t(q)\big) \quad (21)$$

Where $\alpha+\beta=1$. The algorithm weighting adds the candidate authority and the category authority.

MR&PB : Distance and review numbers based recommend method, which likewise weighting adds the score of candidate's location authority and category authority, and obtains the optimal parameter pair by experiment.

Proximity Based (PB): Distance based method, recommending candidates whose home location are closer to the query point.

Most Reviewed (MR): Review number based method, recommending candidates who have relatively more reviews on the given category.

For the comparison algorithms involved above, the algorithms except MR and PB need to learn parameters. Among them, α will be set from 0 to 1 and each time increased by 0.1 to obtain the best results. The following describes the parameter selection of this article.

The parameters in this paper are also selected by experiments, and the best parameters combination (α,λ) is selected through the display of the final result. During the process of model learning, two parameters need to be set which are the learning rate α and the regularization coefficient λ. The experiment evaluates the weight of the model by different hyper-parameters and calculates the average accuracy of the final results. By default, here we set $\lambda\in\{10^{-2},10^{-3},10^{-4},10^{-5},10^{-6},10^{-7},10^{-8}\}$, and $\alpha\in\{0.01,0.02,0.03,0.04,0.05\}$.The result shows in figure 10, from which the combination $\lambda=10^{-7},\alpha=0.02$ is selected because of the biggest accuracy.

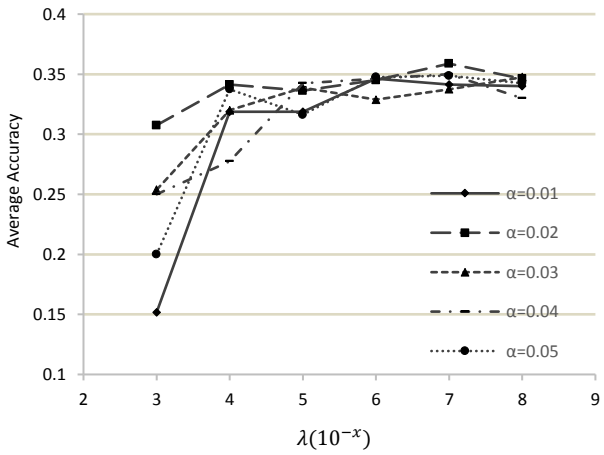
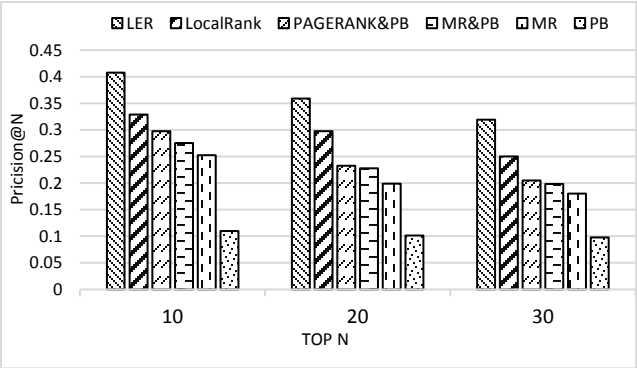


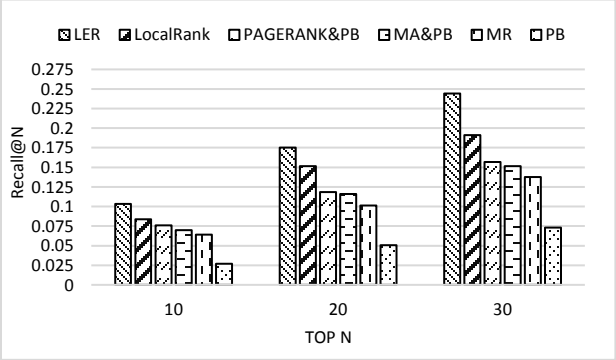
Figure 10 The accuracy of different hyper-parameters combinations

6.2 Experiment Results

The experiments of Top-N local expert recommendation are carried out using the above-mentioned several local expert finding algorithms on shopping category of city Las Vegas, comparing the performance on the metrics of precision and recall, shown as Figure 11.



(a) The comparisons of precision



(b) The comparisons of recall

Figure 11 The comparisons of experiment result

It can be seen from the experiment results is that the results of the local expert recognition algorithm proposed in this paper are superior compared to those of the other algorithms for different N values. It is probably because of the data sparsity causing center location deviation, directly leading to worse recommending results for PB method. The experiment on MR illustrates that local expert level cannot only depend on review numbers of a category. Candidate's active location, structural information, and semantic information are all important in the identification of local experts. The PageRank & PB algorithm that adds network structure of candidates is slightly better than MR & PB, but the classic network structure ranking algorithm cannot satisfy the effect of expert recognition in such network filled with abundant information. The algorithm LocalRank is a classic way for local expert research and is often used as the primary contrast experiment to evaluate results. LocalRank algorithm that combines the candidate's authority degree of location and category presents relatively better, but the algorithm proposed in this paper is more comprehensive and has the best recommend results. It not only makes full use of candidate's different aspects of information, improving the existing classic expert recognition algorithm, but also give a unique method of dealing with points processing, which can obtain more comprehensive expert information for different locations, thus playing a good effect on local experts recommend.

7 Conclusion

In this paper, a review-based local expert discovery algorithm is proposed. After the feasibility analysis of local expert research in Yelp, the candidates set is selected for the given query combining the scores of the personal attributes, the review semantic, the friend relationship and the location preference. Scores based on network structure and context are calculated to measure candidates' preference, and local expert model is proposed to evaluate the level of candidates. The implicit feedback data is used to learn the weight vector in the model. Experiments show that our approach has a better effect on real datasets compared with other typical methods.

In the future, we will further improve the recommendation method to make it suitable for the online real-time network data flow environment. Personalized information of users such as personality will also be considered to recommending personalized local expert users. In recent, a few research works have discussed the device discovery [23], user availability [24], and privacy issues [25] in expert discovery, in our future works we will take these issues into consideration.

Acknowledgement

This work is supported by National Natural Science Foundation of China under Grants No. 61772133, No.61472081, No. 61402104. Jiangsu Provincial Key Project BE2018706. Key Laboratory of Computer Network Technology of Jiangsu Province, Jiangsu Provincial Key Laboratory of Network and Information Security under Grants No. BM2003201, and Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grants No. 93K-9.

REFERENCES

[1] Traynor D, Curran K.Location-based social networks [J]. From Government to E-Governance: Public Administration in the Digital Age, 2012: 243.

[2] Antin J, de Sa M, Churchill E F. Local experts and online review sites[C]//Proceedings of the acm 2012 conference on computer supported cooperative work companion. ACM, 2012: 55-58.

[3] Y. Ma, Y. Wu, J. Ge, J. Li, "An Architecture for Accountable Anonymous Access in the Internet-of-Things Network," IEEE Access, vol. 6, pp. 14451-14461, 2018.

[4] Balog K, Azzopardi L, De Rijke M. Formal models for expert finding in enterprise corpora[C]//Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2006: 43-50.

[5] Weng J, Lim E P, Jiang J, et al. Twitterrank: finding topic-sensitive influential twitterers[C]//Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010: 261-270.

[6] Chen W, Cheng S, He X, et al. Influencerank: An efficient social influence measurement for millions of users in microblog[C]//Cloud and Green Computing (CGC), 2012 Second International Conference on. IEEE, 2012: 563-570.

[7] Lin S, Hong W, Wang D, et al. A survey on expert finding techniques[J]. Journal of Intelligent Information Systems, 2017: 1-25.

[8] Cheng Z, Caverlee J, Barthwal H, et al. Who is the barbecue king of texas?: a geo-spatial approach to finding local experts on twitter[C]//Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. ACM, 2014: 335-344.

[9] Lu H, Caverlee J. Exploiting geo-spatial preference for personalized expert recommendation[C]//Proceedings of the 9th ACM Conference on Recommender Systems. ACM, 2015: 67-74.

[10] Jindal T. Finding local experts from Yelp dataset[D]. 2015.

[11] Niu W, Liu Z, Caverlee J. LExL: A learning approach for local expert discovery on twitter[C]//European Conference on Information Retrieval. Springer International Publishing, 2016: 803-809.

[12] Niu W, Liu Z, Caverlee J. On Local Expert Discovery via Geo-Located Crowds, Queries, and Candidates[J]. ACM Transactions on Spatial Algorithms and Systems (TSAS), 2016, 2(4): 14.

[13] Cao Z, Qin T, Liu T Y, et al. Learning to rank: from pairwise approach to listwise approach[C]//Proceedings of the 24th international conference on Machine learning. ACM, 2007: 129-136.

[14] Burges C J C, Svore K M, Bennett P N, et al. Learning to Rank Using an Ensemble of Lambda-Gradient Models[C]//Yahoo! Learning to Rank Challenge. 2011: 25-35.
[15] Clark P J, Evans F C. Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations[J]. Ecology, 1954, 35(4):445-453.
[16] McAuley J, Leskovec J. Hidden factors and hidden topics: understanding rating dimensions with review text[C]//Proceedings of the 7th ACM conference on Recommender systems. ACM, 2013: 165-172.
[17] Aral S, Walker D. Identifying influential and susceptible members of social networks[J]. Science, 2012, 337(6092): 337-341.
[18] Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence. AUAI Press, 2009: 452-461.
[19] GonzÃ lez M C, Hidalgo C A, BarabÃ si A L. Understanding individual human mobility patterns[J]. Nature, 2 008, 453(7196):779-782.
[20] Shancang Li, Theo Tryfonas, Gordon Russell, Panagiotis Andriotis: Risk Assessment for Mobile Systems Through a Multilayered Hierarchical Bayesian Network. IEEE Trans. Cybernetics 46(8): 1749-1759 (2016)
[21] Shancang Li, Li Da Xu, Shanshan Zhao: 5G Internet of Things: A survey. Journal of Industrial Information Integrity, 10(1): 1-9 (2018)
[22] Y. Zuo, Y. Wu, G. Min, L. Cui, "Learning-based Network Path Planning for Traffic Engineering," Future Generation Computer Systems, vol. 92, pp. 59-67, DOI: 10.1016/j.future.2018.09.043, 2019.
[23] Maniak, T., Jayne, C., Iqbal, R., Doctor, F., (2015): “Automated Intelligent System for Sound Signalling Device Quality Assurance” "Information Sciences, Elsevier, vol 294, pp. 600-611.
[24] Iqbal, R., Shah, N., James, A., Duursma, J., (2011): "ARREST: From Work Practices to Redesign for Usability", The International Journal of Expert Systems with Applications, Elsevier, 38(2), pp.1182-1192.
[25] Z. Guan et al., "ECOSECURITY: Tackling Challenges Related to Data Exchange and Security: An Edge-Computing-Enabled Secure and Efficient Data Exchange Architecture for the Energy Internet," in IEEE Consumer Electronics Magazine, vol. 8, no. 2, pp. 61-65, March 2019