A Novel Social Search Model Based on Clustering Friends in LBSNs

Yang Sun, Jiuxin Cao(✉), Tao Zhou, and Shuai Xu

Jiangsu Provincial Key Laboratory of Network and Information Security,
School of Computer Science and Engineering, Southeast University, Nanjing, China
{sunyang,jx.cao,zhoutao,xushuai}@seu.edu.cn

Abstract. With the development of online social networks (OSNs), OSNs have become an indispensable part in people's life. People tend to search information through OSNs rather than traditional search engines. Especially with the appearance of location-based social networks (LBSNs), social search in LBSNs is increasingly important in the burgeoning mobile trend. This paper proposes a novel social search model, harnesses users' social relationship and location features provided by LBSNs to design a ranking algorithm that takes three kinds of ranking scores into account comprehensively: Social Score (scores based on social influence), Searching Score (scores based on professional relevance) and Spatial Score (scores based on distance), finally produces high-quality searching results. Once receiving users' query, the social search engine aims to return a list of ranking POIs (points of interests) that satisfies users. The dataset is extracted from Foursquare, a real-world LBSN. The experiment results show that the ranking algorithm can benefit the social search model in LBSNs evidently.

Keywords: LBSNs · Social search model · Social Score · Searching Score · Spatial Score

1 Introduction

In the past few years, with the rapid development in the mobile field, location-based social network services such as Foursquare and Yelp, have seen increasing popularity, attracting millions of users. Supported by the capabilities of portable devices like smart phones, the location-aware technology like GPS and Wi-Fi, people can easily share their locations, comments and other information with other users. The LBSN [1] services not only help users to strengthen their social connections, but also provide useful searching information.

Information retrieval and knowledge discovery are the main purposes of the web search. Because of the fast and updated information available on the web, users usually rely on search engines to obtain the information. Searching is always considered as an individual activity [2] in traditional social engines like Google, however, with the popularity of OSNs, people are pursuing personalized searching and mass collaboration. Social search [3] could meet people's needs, which makes users find out right people (friends, other similar users or domain experts) quickly and accurately to answer

S. Wang and A. Zhou (Eds.): CollaborateCom 2016, LNICST 201, pp. 679–689, 2017.
DOI: 10.1007/978-3-319-59288-6_68

questions. Recently, Facebook has partnered with Bing and introduced a social search engine called "Graph Search" [4] that associates the results with friends' suggestions.

Applying social search on the LBSNs is an appealing trend. When users search a nearest POI with friends visiting experiences provided by LBSNs, in addition to the traditional social information, exploiting useful location information could make searching results more accurate. For example, if a user wants to search a suitable restaurant for dinner, however, he does not be familiar with the surrounding restaurants, then all the restaurants are candidates and it is better to pick a restaurant which is near the place and has received high evaluations from his friends. On the one hand, the searching results are high-quality. The picked restaurants are both short-distance and high-evaluation, which is better than the traditional social search that only considers social relation; on the other hand, the social search engine provides believable results to users. Users are more inclined to believe and choose POIs once showing friends' experience or evaluations.

Considering such problems, in this paper, we propose a social search model and design a novel ranking algorithm. The dataset is extracted from Foursquare that is a heterogeneous network, and the data is quite sparse. Sparse data could largely influence the accuracy of results. To enhance the data density, we creatively cluster user's friends in the research of social search. Based on clustering friends, the ranking algorithm creatively considers Social Score, Searching Score and Spatial Score comprehensively. Social Score means social influence, social features include not only the traditional social relationship but also location features; Searching Score means professional relevance, which measures the similarity between the query and POIs; Spatial Score is the distance between the locations of users and POIs, the shorter distance means the better score.

The contributions of this paper can be summarized as follows:

1. To enhance the data density and reduce the influence of the sparse data, as far as we know, it is the first time to apply clustering users' friends in the research of social search in LBSNs;
2. To get high-quality ranking results and consider the distance factor in reality, in addition to the traditional Social Score, we take Searching Score and Spatial Score into account in the research of social search.

The rest of this paper is organized as follows. Section 2 reviews related work on social search in OSNs and LBSNs. Then an overview of the social search model is introduced in Sect. 3. The details of the ranking algorithm is presented in Sect. 4. Section 5 describes the validation of our model. Finally, we conclude this paper and state several directions for future work in Sect. 6.

2 Related Work

With the increasing popularity of social networking platforms, social search is attracting significant number of interests in the research field since traditional search engines do not always provide high-quality searching results. However, social search is personalized, so there are different social search engines and social search algorithms [5], a lot of works are done based on different start points.

Some researches concentrate on the problem of designing social search engine. HeyStaks [6, 7] is an Irish social search engine, it applies the recommended technology on Google, Bing and Yahoo based on users' interests and reputation, then returns searching results from Twitter and OSNs. M. R. Bouadjenek, H. Hacid and M. Bouzeghoub [8] introduce a social search engine called LAICOS, which includes social information and personal services. On the one hand, it can provide personalized social document representations; on the other hand, users can use its personalized social query expansion framework to expand searching process. Horowitz and Kamvar [9] design a large-scale social search engine called Aardvark. They use an intimacy metric between users and connect users with specific questions to find the user who is most likely to be able to answer the question. The intimacy is set based on many features, including vocabulary match, profile similarity, social connection and so on.

Some other researches focus on the problem of improving social algorithms. D. Sharma et al. [10] present a self-adapting social search algorithm based on proximity, similarity and interaction. Bao et al. [11] explore the use of social annotations, they propose SocialPageRank to measure the page popularity based on its annotations and SocialSimRank for the similarity between social annotations and web queries. Guo Liang et al. [12] present two ranking algorithms: topic relevance rank (TRR) evaluates users' professional score on the relevant topics; social relation rank (SRR) captures the social relation strength between users.

However, there are quite few researches on social search in LBSNs. Hu et al. [13] define friends-based k nearest neighbors (F-KNN) query, which aims at finding objects near the query location as well as receiving high evaluation from user's friends. But they pay main attention to increasing the searching speed, so they design a F-Quadtree index, and do not perform well on the searching accuracy based on social features. Yuan et al. [14] propose a KNN search on road networks by incorporating social influence, but they do not perform well on the speed of the computation of the social influence over large road and social networks. In contrast to the above works, our research aims to design a good social search model that provides accurate results quickly.

3 The Social Search Model

The social search model is shown in Fig. 1. Vertically, like traditional search engines, the whole architecture is divided into two parts: offline crawler and online searching. Horizontally, there are three main parts: Social Score, Searching Score and Spatial Score. The different functions of these components are described below.

Database. Database maintains the data basis of the social search architecture. The dataset is crawled and extracted from Foursquare, the data types include user's ID and relation; POIs' information that users need to search, including POIs' name, ID, category, description, latitude and longitude; check-ins' ID; timezone.

Searching Score. In this part, we design a search engine based on Lucene [15]. The core of search engine is Inverted Index [16]. We build an Inverted Index based on POIs' information. Then by calculating the similarity scores between user's query and the

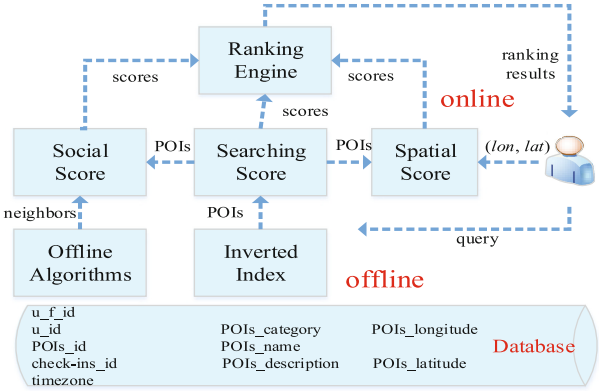

Fig. 1. This is the figure displaying the model of social search

"document" in the index, the alternative POIs' names are picked out and sent to the Social Score and Spatial Score. The similarity scores are sent to Ranking Engine.

Social Score. This part provides offline algorithms, which are updated K-means and updated KNN, the purpose of the updated K-Means is to cluster users' friends to enhance the data density; the purpose of updated KNN is to find out some friends that are most similar with users. Users are always more inclined to believe the most similar friends [17]. Then we extract some social features including friends' activity and evaluation on the alternative POIs to calculate social scores that are sent to the Ranking Engine.

Spatial Score. Users are more inclined to visit short-distance POIs, so the distance between the locations of users and alternative POIs could be transformed as scores that are sent to the Ranking Engine.

Ranking Engine. This part produces the final alternative POIs' ranking results. We assign proper weight coefficients to the three kinds of ranking scores according to their own importance, the sum of three weight coefficients is 1. The final results are returned to users.

4 The Ranking Algorithm

The ranking algorithm of the social search in this paper is to take Searching Score, Social Score and Spatial Score into account comprehensively. Each kind of score has a different weight coefficient according to their own importance. The whole process is that users input a query like "Starbucks coffee", the ranking result is a list of POIs' names about Starbucks and coffee, the aim in our research is fast speed and accurate results. The detailed description of the algorithm is given below.

4.1 Searching Score

Searching Score means professional relevance. In this part, we first design a search engine, the purpose is that when users input a query, the ranking result is some alternative POIs according to the ranking scores of similarity. Then the searching scores are sent to Ranking Engine and the alternative POIs are sent to Social Score and Spatial Score. This part is divided into offline Inverted Index and online searching.

Offline Inverted Index. Inverted index is the core of search engine. In traditional search engines, the forward index is the "document" and the backward index is the "term". Similarly, based on our dataset, we take POIs' name, category, description as the "document" and POIs' name as the "term" to build a Inverted Index.

Online Searching. Apache Lucene is a free and open-source information retrieval software library, the search engine in this paper is designed based on it. The process is as follows. When users input a query - "Starbucks coffee", the tokenizer in the Lucene divides the phrase into "Starbucks" and "coffee". Then the similarity algorithm in Lucene is used to calculate similarity scores between the query and the "document", the scores are sorted in descending order and we pick out the searched POIs' names as the candidates. In the similarity algorithm, to improve professional relevance, we give POIs' name the highest weight, then POIs' category, finally POIs' description. The candidates are sent to Social Score and Spatial Score. The similarity scores are sent to Ranking Engine.

4.2 Social Score

Social Score means social influence. In this part, offline algorithms are used to pick out the user's some most similar friends (10, 20, 30, …); online we extract some social features from these friends such as activity, evaluation on the alternative POIs. The purpose is to make sure that the most similar friends produce the highest-quality social scores.

Offline Algorithms. As mentioned above, social search is to find out the right people to answer questions [18]. Friends are the right people, so we use updated KNN algorithm to find out some most similar friends. Before that, to reduce the bad influence of sparse data, we use updated K-means algorithm to cluster friends to improve the data density. Because of so many data, K-means algorithm has good clustering effect among the clustering algorithms. "Check-ins" is an unique location-based data type from the Four-square. When a user goes to a POI and feels good, he would click the "Check-In" tag, which means he has gone there once, the more "check-ins" means the higher evaluation on the POI.

We use updated K-means algorithm, the similarity metric is the vector, not the distance, so compared with Euclidean Distance, Cosine Similarity [19] is better to measure the similarity between friends. If two friends have more "check-ins" on the same POI, they are more similar. The similarity formula is given below.

$$sim(f_{i_1}, f_{i_2}) = \frac{\sum_{l_j \in L} c_{f_{i_1}, l_j} c_{f_{i_2}, l_j}}{\sqrt{\sum_{l_j \in L} c_{f_{i_1}, l_j}^{\ 2}} \sqrt{\sum_{l_j \in L} c_{f_{i_2}, l_j}^{\ 2}}} \tag{1}$$

Where c_{f_i, l_j} means the number of "check-ins" a friend visited a POI; L means all the POIs.

K-means requires a certain number of clustering -k, after many experiments, we find that when k is 3, the clustering effect is best.

Algorithm 1. Updated K-means

Input: k =3, data={ f_i, l_j, c_{f_i, l_j} };

Output: $matrix(k, n)$, $matrix(m, k)$;

1.Select a friend randomly as an initial clustering center in each cluster of k;

2.For $i = 1 : m$ {

3. Calculate similarity scores between f_i and clustering center k;

4. Pick out the highest similarity score;

5. If(the highest score > threshold){

6. f_i belongs to the cluster;

7. }

8.}

9.In each cluster, take the average number of all the friends' check-ins as the clustering center, recalculate every clustering center;

10.Until the clustering centers remain stable.

We get $matrix(k, n)$ and $matrix(m, k)$. n is the number of POIs and m is the number of user's friends. $matrix(k, n)$ means clustering centers' evaluation on POIs. $matrix(m, k)$ means the similarity between friends and clustering centers.

Algorithm 2. Updated KNN

Input: $matrix(k,n)$, $matrix(m,k)$, the user's "check-ins";

Output: the user's h nearest neighbors;

1.Calculate the similarity between the user and k clustering centers and get $1\times k$ vector, $(v_1, v_2, ..., v_k)$;

2.Calculate Euclidean Distance between the vector and $matrix(m,k)$;

3.Take h friends that get smallest distance as the user's nearest neighbors.

And the Euclidean Distance formula is given below.

$$dis(u,f_i) = \sqrt{\sum_{j=1}^{k} (sim(u,k_j) - sim(f_i,k_j))^2} \tag{2}$$

Therefore, when the user inputs a query, the *h* friends could produce social influence on alternative POIs for the user.

Online Social Features. We have got alternative POIs and *h* nearest neighbors, then we need suitable social features to show social influence based on the "check-ins". There are two such social features: activity and evaluation.

Friends' activity can be measured by the number of "check-ins". A user's friend who has more check-ins is more active in the district, so the POIs he recommends are more believable. The activity formula is given below.

$$act(f_i) = \sum_{l_j \in L} C_{f_i,l_j} \tag{3}$$

Where $act(f_i)$ is the number of check-ins of f_i on all the POIs in the district.

Friends' evaluation on alternative POIs can be also measured by the number of check-ins, a friend who has more check-ins on a POI means that he often goes there, so the evaluation is high. The evaluation formula is given below.

$$eva(f_i,l_j) = c_{f_i,l_j}, l_j \in CL \tag{4}$$

Where *CL* means the alternative POIs.

So the social influence formula is given below.

$$social(l_j) = \sum_{i=1,j=1}^{h,x} eva(f_i, l_j) * act(f_i) \tag{5}$$

Where x is the number of candidate POIs.

4.3 Spatial Score

Users are more inclined to go to short-distance POIs, so distance is an important factor that influences ranking results. We need to calculate the distance between the locations of users and each candidate POI, the distance formula is given below.

$$dis(l_1, l_2) = R * \arccos(\sin(lat_1) * \sin(lat_2) * \cos(lon_1 - lon_2) + \cos(lat_1)\cos(lat_2)) * PI/180 \tag{6}$$

Where R is the radius of the Earth.

4.4 Ranking Engine

We have got three kinds of scores: Social Score, Searching Score and Spatial Score. To produce the final ranking result, we assign three different weight coefficients to them. The ranking formula is given below.

$$\begin{aligned} r(u, l_j) = \alpha * query\left(l_j\right) + \beta * social(l_j) + (1 - \alpha - \beta) * dis(l_j) \\ (\alpha + \beta) \in [0, 1] l_j \in CL \end{aligned} \tag{7}$$

In this paper, β is the highest, α is the second, $(1 - \alpha - \beta)$ is the smallest. Because we study the social search in LBSNs, the social coefficient should be the highest to emphasize social influence. Then compared Searching Score with Spatial Score, the professional relevance is more importance. After many experiments, when $\alpha = 0.3$, $\beta = 0.61 - \alpha - \beta = 0.1$, the F1-measure in Sect. 5 has the best result, which means the best effectiveness. To calculate simply and accurately, we do normalization on the three kinds of scores.

5 Evaluation

We perform a set of experiments on the real-world data to validate our proposed algorithm. The dataset is sampled from Foursquare, a popular LBSN across the world. We select the dataset in the New York by matching the timezone, there are 413,989 check-ins, 4,741 users, 56,868 POIs and 128892 social links. The data sparsity is 1.07%.

Evaluating the search results is a great challenge since relevance judgments can only be assessed by the searchers themselves, especially in the social search context. We divide all the users into the training set and the test set by 10-fold cross-validation, then we selected a group of volunteers from the tested users randomly to help us initiate 500 queries and manually judge the Top-10, 20, 30 searching results obtained from four methods: traditional KNN search, which does not contain clustering friends, Searching

Score and Spatial Score (Method1); traditional KNN search based on clustering friends, which does not contain Searching Score and Spatial Score (Method2); KNN search based on clustering friends, which does not contain Spatial Score (Method3); the ranking algorithm in this paper (Method4). We use Method1 as the baseline.

For the purpose of evaluation, we use precision, recall and F1-Measure [20] as the effectiveness metric (Top-N ranking POIs, N = 10, 20, 30). The comparison results are presented in Figs. 2, 3 and 4. We can see that, our ranking algorithm (Method4) is better than other three methods. Method2 is better than Method1, which means that improving data density by clustering users' friends is effective, and friends are the right people in the research of social search. Method3 is better than Method2, which means that professional relevance (Searching Score) is also an important factor in enhancing the accuracy of searching results. And our ranking algorithm is the best, which means taking Social Score, Searching Score and Spatial Score into account comprehensively is suitable to be applied in the social search in LBSNs.

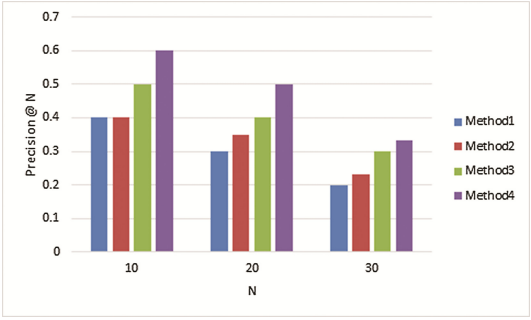

Fig. 2. This is the figure displaying the Precision of Top-N (N = 10, 20, 30)

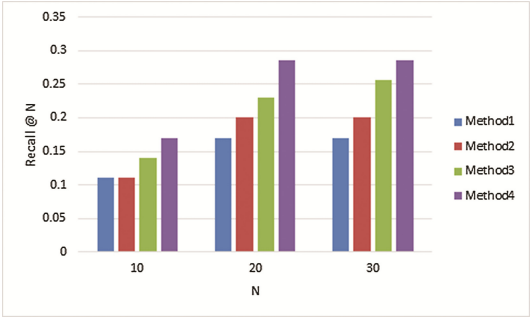

Fig. 3. This is the figure displaying the Recall of Top-N (N = 10, 20, 30)

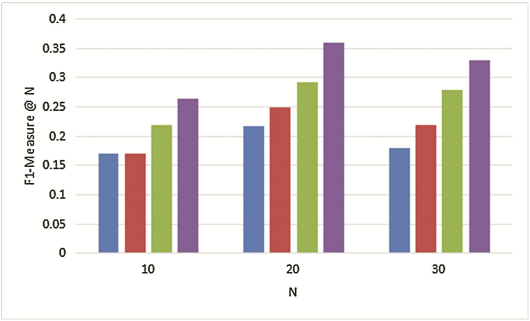

Fig. 4. This is the figure displaying the F1-Measure of Top-N (N = 10, 20, 30)

6 Conclusion and Future Work

In this paper, we propose a social search model for LBSNs. To produce high-quality searching results quickly, a novel ranking algorithm that considers Social Score, Searching Score and Spatial Score comprehensively is proposed. Meanwhile, to solve the problem of sparse data from heterogeneous networks, considering the offline and online mechanism of traditional search engines, we creatively enhance the density by clustering users' friends off the line, which reduces the influence on the accuracy while online searching. We have performed experiment on the real-world data collected from a popular LBSN across the world. The experiment results demonstrate the effectiveness of our ranking algorithm.

In fact, there are still a lot of optimizable opportunities to explore in our work. Firstly, we can get other data types, in addition to the "check-ins", users would write comments on the POIs that they have visited, although compared with the "check-ins", the data of comments is much sparser, which still produces some influence on social part. Secondly, semantic analysis could be used in calculating the similarity between the query and the "document" in the Inverted Index to improve the accuracy. Thirdly, how to run the ranking algorithm at a low time and memory cost are worth further analyzing for future research.

Acknowledgments. This work is supported by National Natural Science Foundation of China (61272531, 61202449, 61272054, 61370207, 61370208, 61300024, 61320106007 and 61472081), China high technology 863 program (2013AA013503), Jiangsu Technology Planning Program (SBY2014021039-10), Jiangsu Provincial Key Laboratory of Network and Information Security under Grant No. BM2003201 and Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grant No. 93k-9.

References

1. Andris, C.: LBSN data and the social butterfly effect (vision paper). In: Proceedings of the 8th ACM SIGSPATIAL International Workshop on Location-Based Social Networks. ACM (2015)
2. Irfan, R., et al.: Survey on social networking services. IET Netw. **2**(4), 224–234 (2013)
3. Freyne, J., Smyth, B.: An experiment in social search. In: Bra, P.M.E., Nejdl, W. (eds.) AH 2004. LNCS, vol. 3137, pp. 95–103. Springer, Heidelberg (2004). doi: 10.1007/978-3-540-27780-4_13
4. Khan, Z.C., Mashiane, T.: An analysis of Facebook's graph search. In: Information Security for South Africa (ISSA). IEEE (2014)
5. Evans, B.M., Chi, E.H.: Towards a model of understanding social search. In: Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work. ACM (2008)
6. Smyth, B., Briggs, P., Coyle, M., O'Mahony, M.: Google shared. A case-study in social search. In: Houben, G.-J., McCalla, G., Pianesi, F., Zancanaro, M. (eds.) UMAP 2009. LNCS, vol. 5535, pp. 283–294. Springer, Heidelberg (2009). doi:10.1007/978-3-642-02247-0_27
7. McNally, K., O'Mahony, M.P., Coyle, M., Briggs, P., Smyth, B.: A case study of collaboration and reputation in social web search. ACM Trans. Intell. Syst. Technol. (TIST) **3**(1), 4 (2011)
8. Bouadjenek, M.R., Hacid, H., Bouzeghoub, M.: LAICOS: an open source platform for personalized social web search. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1446–1449. ACM (2013)
9. Horowitz, D., Kamvar, S.D.: The anatomy of a large-scale social search engine. In: Proceedings of the 19th International Conference on World Wide Web, pp. 431–440. ACM (2010)
10. Sharma, D., Alam, A.K.Z., Dasgupta, P., Saha, D.: A ranking algorithm for online social network search. In: Proceedings of the 6th ACM India Computing Convention, p. 17. ACM (2013)
11. Bao, S., Xue, G., Wu, X., Yu, Y., Fei, B., Su, Z.: Optimizing web search using social annotations. In: Proceedings of the 16th International Conference on World Wide Web, pp. 501–510. ACM (2007)
12. Guo, L., Que, X., Cui, Y., Wang, W., Cheng, S.: A hybrid social search model based on the user's online social networks. In: 2012 IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS), vol. 2, pp. 553–558. IEEE (2012)
13. Hu, H., Feng, J., Liu, S., Zhu, X.: Social-Aware KNN search in location-based social networks. In: Li, F., Li, G., Hwang, S., Yao, B., Zhang, Z. (eds.) WAIM 2014. LNCS, vol. 8485, pp. 242–254. Springer, Cham (2014). doi:10.1007/978-3-319-08010-9_27
14. Yuan, Y., Lian, X., Chen, L., Sun, Y., Wang, G.: RS k NN: k NN search on road networks by incorporating social influence. IEEE Trans. Knowl. Data Eng. **28**(6), 1575–1588 (2016)
15. Hatcher, E., Gospodnetic, O.: Lucene in action (2004)
16. Inverted Index. https://en.wikipedia.org/wiki/Inverted_index
17. Liu, F., Lee, H.J.: Use of social network information to enhance collaborative filtering performance. Expert Syst. Appl. **37**(7), 4772–4778 (2010)
18. Chi, E.H.: Information seeking can be social. Computer **3**, 42–46 (2009)
19. Ye, M., et al.: Exploiting geographical influence for collaborative point-of-interest recommendation. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM (2011)
20. Precision, Recall, F1-measure. https://en.wikipedia.org/wiki/Precision_and_recall