Cluster Analysis of Stations Based on Weight SimRank in Sharing Bicycle

Bo Guan, Wei Jiang*, Jie Yao, Shijie Tang, Jinhao Wang
School of Electric and Information Engineer, Ningbo University of Technology, China
**Corresponding Author.*

Abstract

With the increasing popularity of shared bikes, the indiscriminate parking of bicycles in cities has increasingly become a difficulty in urban management. The tidal phenomenon of large numbers of urban residents during rush hour is the root cause of the indiscriminate parking of bicycles in many subway stations and commercial areas. Optimizing the scheduling strategy of shared bikes is one of the effective solutions to solve the problem of random parking and reduce the scheduling cost. The cycle is a short - distance vehicle, and its circulation law is in line with the characteristics of the small world of urban traffic. That is, most of the bikes flow within the small world region, while only a small part of the bikes flow between the small world regions. With the massive accumulation of bike-sharing borrow and return data, the method of clustering the borrow and return stations and dividing regions according to the clustering results has attracted the attention of industry experts and researchers. it is effectively to apply in intelligent scheduling related industries. Although there have been some studies on the station clustering in the current literature, because these studies are basically based on the fixed features of the site (site location, pile number, etc.), the results cannot find an effective small world region of bicycles. In order to find out the effective small world region of bicycles, we introduced the idea of SimRank (that is, the similarity of a station is due to the similarity of its bicycle source station and destination station), and assigned weights to association relationships (the number of times of borrowing and returning) to define the similarity algorithm w-SimRank of stations. Then, the station clustering was done in line with skyline thinking. Finally, in order to verify the effectiveness of the algorithm, we implemented the station clustering based on SimRank algorithm, and compared the clustering effect with the W-SimRank algorithm proposed in this paper to verify the effectiveness of the W-SimRank algorithm, and analyzed the influence of the key parameters of the algorithm on the algorithm. And then

***Keywords**: Cluster Analysis, Sharing Bicycle, SimRank, Intelligent Scheduling.*

I. Introduction

Bike-sharing companies provide services at campuses, subway stations, bus stops, residential areas, business districts and public service areas to complete the final "jigsaw" of the transportation industry and drive residents' enthusiasm for using other sharing transportation tools. Synergies with other forms of sharing transport. Bike-sharing is not only a time-sharing leasing model, but also a new kind of green and environment-friendly sharing economy. The bike-sharing industry is entering a new stage of development after years of brutal growth. But bike-sharing is also facing some new challenges: uneven distribution of stations, high operating costs and a serious imbalance between the supply and demand of bicycles during tidal periods are among the more prominent problems. With the new round of financing of Harlow Travel and Qingju Bike successively, bike-sharing gradually starts to enter the stage of steady development. However, how to use technical means to improve the efficiency of supply and demand matching and enhance the refined operation ability has become the focus of the industry competition in the future. The biggest operating costs of bike-sharing include the bike-sharing system and off-line bike dispatching. Due to the commuting tide problem of bicycle use, bicycle scheduling has not only become one of the important costs of bicycle operation enterprises, but also become the main source of disorderly parking. Therefore, good scheduling is not only the key to reduce operating costs for bicycle operators, but also the key to solve the problem of disorderly parking of shared bikes in cities.

With the increasing popularity of shared bikes, the problem of bicycle random parking in the city has become increasingly difficult to urban management. The tidal phenomenon of large numbers of urban residents during rush hour is the root cause of the indiscriminate parking of bicycles in many subway stations and commercial areas.

ISSN: 0010-8189

Optimizing the scheduling strategy of shared bikes is one of the effective solutions to solve the problem of random parking and reduce the scheduling cost. The cycle is a short - distance vehicle, and its circulation law is in line with the characteristics of the small city world of traffic. That is, most of the bikes flow within the small world region, while only a small number of bikes flow between the small world regions. This is also the reason why the current urban cycle scheduling basically adopts the scheduling strategy combining local and global, that is, most of the scheduled vehicles flow only within a local region, while a small number of scheduled vehicles flow between regions. Most of the current regional divisions are based on simple administrative regions, but due to the recent rapid development of the city, the way of division of urban administrative regions is difficult to represent the actual function of the city.

Meanwhile, sharing bikes and shared bikes in cities across the country have been in operation for many years and have accumulated a huge amount of data on bike borrowing and returning. Many industry experts regard the adoption of big data analysis technology as the best solution or practical basis to solve this problem. Many researchers have studied the station clustering and proposed the corresponding algorithm. At present, most of the station clustering is based on static features of the station (such as station location, number of piles, etc.) or dynamic features (the number of bikes lent, returned, and the number of empty piles, etc.). However, this clustering algorithm is difficult to be applied in the actual bicycle scheduling policy, the reason is that this clustering algorithm is only to cluster the stations with the same characteristics into a class, and the stations in these clustering lack correlation.

Because cities have the characteristics of a small world, that is, most of the people in a certain area move in a certain area. As a means of transportation in the last kilometer, the data of bike borrowing and returning also have regional characteristics, that is, most bikes will be transferred within a certain area, while the number of bikes transferred between areas will not be too large. Therefore, by finding out these areas and dividing the whole city into multiple areas, a combination of global scheduling and local scheduling can be adopted in the urban bicycle scheduling algorithm to replace the disadvantages of the current scheduling which is simply divided by administrative areas.

Each bike-sharing system and city sharing bicycle system records the data of bicycle borrowing and returning, which not only records the time of bicycle borrowing and returning, but also records the location of the bicycle borrowing and returning. Based on the urban traffic has the characteristics of a small world, this paper calculates the similarity between stations on the basis of the correlation between stations to borrow and return bicycles, and then clusters the stations on this basis. It aims to provide data basis for bicycle scheduling strategy, urban bicycle station layout, intelligent traffic and other aspects.

II. Related Research

As an important part of urban transportation, urban sharing bicycle system focuses on solving the problem of "the last kilometer of bus", which is of great significance to residents' travel and the development of urban sharing transportation. However, since the development of the sharing bicycle system, many scholars at home and abroad have done relevant studies on the problems it faces from the aspects of station clustering and regional division. Literature [2] uses the K-means algorithm to cluster the stations of Paris sharing bicycle system (Velib '), dividing the stations into three types: balanced, loaded and non-loaded. The dynamic time leveling (DTW) algorithm is used to calculate the degree of similarity among the clusters. At last, the DTW Barycenter (DBA) algorithm is used to compute the values of the cluster center. Related to this, literature [3] conducted functional clustering of rental points according to the accumulated historical use data of the sharing bicycle system to distinguish the use patterns among various rental points. LDA model was used, and k-means algorithm was also used, but the classification was based more on the functionality of the station. Literature [4] defines three ratios: available bicycle rate (NAB), travel accumulation rate (CumT) and station turnover rate (TS). Each ratio is divided into three categories: high, medium and low. All stations are classified to obtain three groups, and the final classification is obtained by combining these three clustering groups.Literature [5] clear using the Euclidean community clustering analysis method, using the flow clustering algorithm and greedy fast detection algorithm and so on, calculating and analyzing the operation of Chicago's urban bicycle system, and dividing the time period, analysing how to classify the Chicago's shared bicycle stations according to the pressure of the vehicles entering and leaving the station and other parameters, finally put forward combining with the research results to create a seamless multimodal transport system (more seamless multi - modal transportation system).

In the study most similar to this paper, literature [1], starting from the need of real-time scheduling of sharing bicycle system, proposed the method of combining association rules and spatial clustering to cluster urban sharing bicycle rental points twice successively (the first time using K-means algorithm to cluster the stations; the second time is based on the clustering results of the first stage, integrating the geographic location of the region, demand type, tidal direction and the number of rental points in the region, and assigning different weight coefficients to cluster), so as to achieve the purpose of real-time scheduling and regional division. But the quadratic clustering algorithm increases the complexity of calculation, and the weight coefficient is too much and there is no uniform standard, which reduces the accuracy of the algorithm to some extent.

Another research area related to this paper is SimRank algorithm.There are still many problems in the similarity calculation of SimRank, such as low efficiency of large graph, too much overhead, unable to judge the similarity accurately, or not can give a worst-case error information and so on, many effective technologies have been proposed for all pair search, single source search, single pair search and partial pair search to optimize the calculation of SimRank. In order to solve the problem of high time complexity of iterative computation, lizorkin et al. Proposed an iterative method to calculate SimRank accuracy for all pairing searches, the required number of iterations is provided, the computational complexity of the iterative algorithm is optimized from O (n ^ 4) to o (n ^ 3), and threshold filtering is introduced in the worst case to improve the efficiency of the method.this greatly optimizes the computing speed of SimRank[6]. Weiren Yu et al. Proposed a strategy to eliminate partial double computation, which accelerated SimRank computation to o (KD'n ^ 2), and expressed SimRank matrix as transition matrix.The convergence rate of SimRank iteration is further accelerated by the exponential sum equation and the general geometric sum of relative objects[7]. At the same time, a new fine-grained locality is proposed.The maximum clustering method improves the computing speed of SimRank. However, the methods proposed by lizorkin and Weiren Yu all need o (n^2) memory to output every time iteration, which is not realistic for computing large-scale graphs. Cuiping Li et al. Proposed the use of noniterative method to compensate for the iterative method in dynamic network gap. They use Kronecker and vectorization operators, and propose a new SimRank algorithm for static and dynamic information networks[8]. This new computing model approximates the SimRank result and obtains o (R ^ $4N^2$) time,Where R (≤ n) is the target rank of SVD. When R is large, in order to achieve high accuracy, it will not blindly speed up the calculation. For single source search, Boyu Tian and Xiaokui Xiao proposed an effective SimRank computing index structure sling, which guarantees the maximum number of errors returned in each SimRank score. It answers that the time complexity of O (1 / ε) and O (n / ε) is close to optimal for any single pair and cell SimRank queries[10]. Yingxia Shao et al. Focused on the problem that graphics become bigger and bigger in reality, a new two-stage random walk sampling framework (TFs) is proposed for SimRank based similarity search, can process dynamic graph with high performance[9]. Minhao Jiang et al. Also proposed to design TSF separate reading index scheme to deal with top SimRank search on dynamic graph,Their random walk based index based SimRank algorithm can deal with large dynamic graphs in a funny way[11]. In order to solve this problem, Yu Liu et al. Proposed probesim, an index free unit algorithm and Top-k SimRank query to solve the problem of too long preprocessing time or too much space cost when processing large dynamic graphs. Because there is no index structure, it can support real-time SimRank queries on dynamic graphs[12]. There are some other optimizations of SimRank. Zhenguo Li et al. Designed the cloudwalker algorithm, which is divided into offline phase and online phase[13]. It is highly parallelizable only in linear time and space. For single pair and fixed time unit queries, couldwalker has a higher efficiency and scalability than the existing one level. Wenbo Tao et al. Designed a two-step effective framework for computing Top-k similar pairs based on Top-k SimRank, In addition, an approximation algorithm is designed to let users be Top-k similar pairs under the specified precision requirements, which makes it has high performance and good expansibility[14]. Zhang et al. Have implemented 10 published algorithms for iterative method, noniterative method and random walk method, and obtained that none of the known algorithms can control other algorithms, for different applications, the best choice of algorithm is not used, so the existing algorithm still has a lot of room for improvement[15].

III. Basic Concept

Urban bicycle-sharing stations and bicycle-sharing stations can be viewed as a graph composed of stations as nodes, borrowing and returning records as edges. It is obvious that SimRank can be used to calculate the similarity of nodes in the graph. However, SimRank and its improved algorithm are based on the correlation between two stations as the evaluation standard, while there is a quantity of correlation strength between sharing bicycle stations. Therefore, this paper will make some improvements to SimRank. In order to describe the algorithm more clearly and accurately, this section introduces some basic concepts and definitions.

Definition 1 (Topological network of sharing bicycle stations) Given the topological network of sharing bicycle stations $G = (\mathrm{V}, \mathrm{E})$, the triad $(a, b, value)$ of the relationship between any two stations a and b, where value represents the weight of directed line segment a->b, namely the number of records from point a to point b in a unit time. $I(a)$ represents the collection of stations pointing to station a, and $|I(a)|$ represents the number of records going to station a per unit time.

Obviously, according to Definition 1, the similarity between two stations can be calculated using the SimRank idea. However, since SimRank has no weight value, the number of bicycles transferred will be ignored if directly calculated. In order to better express the factor of the number of bicycles flowing between stations, we define the similarity between stations as follows:

Definition 2 (similarity) Given two stations a and b, based on Definition 1, the similarity between stations a and b is defined as follows:

$$K(a, b) = \min\left(1, \frac{|I(a)|}{|I(b)|}\right) * \frac{1}{|I(a)| * |I(b)|}$$

$$Sw(a, b) = K(a, b) * \sum_{i \in I(a)} \sum_{j \in I(b)} Sw(i, j) \tag{1}$$

The iterative form of similarity algorithm is as follows:

$$Iw(a, b)^0 = \begin{cases} 1\,; (a = b) \\ 0\,; (a! = b) \end{cases}$$

$$Iw(a, b)^{k+1} = K(a, b) * \sum_{i \in I(a)} \sum_{j \in I(b)} Iw(i, j)^k \tag{2}$$

Where K (a, b) is an intermediate variable.

Definition 3 (T-Similarity) Given a threshold value t, given two stations a and b, based on Definition 2, if the values of Sw(a,b) and Sw(b,a) are both greater than the threshold value t, then a and b are said to meet T-Similarity. It can also be called a (or b) as T-Similarity station of b (or a).

IV. Station Clustering Algorithm Based on Simrank

According to the definitions given in the previous section, the realization process of the clustering algorithm for sharing bicycle stations in this paper is given below. The whole algorithm is divided into two stages. In the first stage, station similarity matrix L is calculated according to SimRank idea and Definition 2, and based on this, matrix elements are normalized (L is a sparse matrix determined by bicycle flow characteristics). The normalization process is realized by function Stretch, and the whole process is realized by function GetSimRankMatrix. In the second stage, based on the given matrix L and the given threshold t, the algorithm classifies the stations into different clusters according to Definition 3. The whole process is implemented by the function GetCluster.

Algorithm 1: Play (G, t)
Input: network topology G, threshold t
Output: clustering results

```
L = GetSimRankMatrix (G, k);
return GetCluster (L, t);
```

The first stage is to calculate the similarity matrix. Similarity matrix L is calculated based on the idea of SimRank, that is, iterative calculation is carried out through the similarity between related stations. However, due to the fact that bicycles are short-distance transportation vehicles, the stations associated with the stations are usually limited to the surrounding stations (the correlation matrix is sparse matrix). Although W-SimRank will enhance the correlation factor (weight) between each station, it will also make the element value of similarity matrix small. Therefore, in addition to introducing weights into SimRank, w-SimRank algorithm also normalizes the similarity matrix obtained by the final iteration. First, the algorithm will iteratively calculate the similarity matrix L according to SimRank algorithm, and the similarity calculation is based on the equation in Definition 2. Secondly, the similarity of all stations is sorted according to size and divided into different values according to proportion. For example, if the similarity of all stations is sorted in ascending order, the similarity of the 90th percentile data in the sorted set is 0.9.

Algorithm 2: GetSimRankMatrix (G, k);
Input: network topology G, the number of iterations k
Output: similarity matrix L

```
L* = Init (L);
i=0;
while(i<k){
    For each station i, j in the station collection
        L[i][j]=Iw(i,j);
    L* = L;
}
Stretch (L);
return L;
```

The second stage is station clustering. The algorithm uses the idea of SkyLine[16] to cluster the stations. The specific idea is as follows: First, the concepts of seed station and super station are introduced, each station is traversed, the other T-Similarity stations that are most similar to this station are defined as the seed station for this station. Define the operation of judging whether a station has a super station is called HavaSuperStation, and the operation of binding a super station and a seed station is called Bind. After the traversal, cluster merging is carried out, and the merging method is as follows: traverse each station and find out its super station; if the super station does not exist, skip this cycle and conduct the next cycle; otherwise, make the following judgement:

<1>If neither the seed station nor the super station belongs to any cluster, the two stations are treated as a new cluster. The super station is the cluster control point, and this operation is defined as CreateCluster.

<2> If only one of the two stations of the seed station and the super station belongs to a certain cluster, add the seed station (super station) that does not belong to any cluster to the cluster of the super station (seed station), and define this operation as ClusterAddStation;

<3> If the seed station and the super station belong to two different clusters, merge the two clusters and define this operation as MergeClusterOfStation. Define the operation of judge whether a station belongs to a certain cluster is called HaveCluster and the operation of determine whether two stations belong to the same cluster is called BelongSameCluster.

Algorithm 3: GetCluster (L, t)
Input: similarity matrix L, threshold t
Output: clustering results

```
for(a:stations)
    for(b:stations;begin=a.next)
        if(a and b meet T-Similarity){
            if(HaveSuperStation(a)) Bind (a, a);
            if(HaveSuperStation(b)) Bind (a, b); }
        else
          if L[a][b]>L[SuperStation[b]][b]
                Bind (a, b);



for(a:stations and HaveSuperStation(a))
        b=SuperStation[a];
        if(!HaveCluster(a) and !HaveCluster(b))
            CreateCluster (b, a);
        if(HaveCluster(a) != HaveCluster(b))
            ClusterAddStation (a, b);
        if(HaveCluster(a) and HaveCluster(b))
            if(!BelongSameCluster (a, b))
                MergeClusterOfStation (a, b);
```

V. Experimental Results

The data used in the experiment was the borrow and return data from sharing bicycle stations in a city in China, and the starting station, borrowing time and returning station were extracted as the test data. The experimental environment was CPU: Intel(R) Core(TM) I5-8300h @2.30GHz and memory: 8GB. In order to verify the effectiveness of the algorithm, we implemented the station clustering of SimRank algorithm (hereinafter referred to as SimRank clustering algorithm), which was compared with our algorithm (hereinafter referred to as w-SimRank clustering algorithm).

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

5.1 Experimental comparison

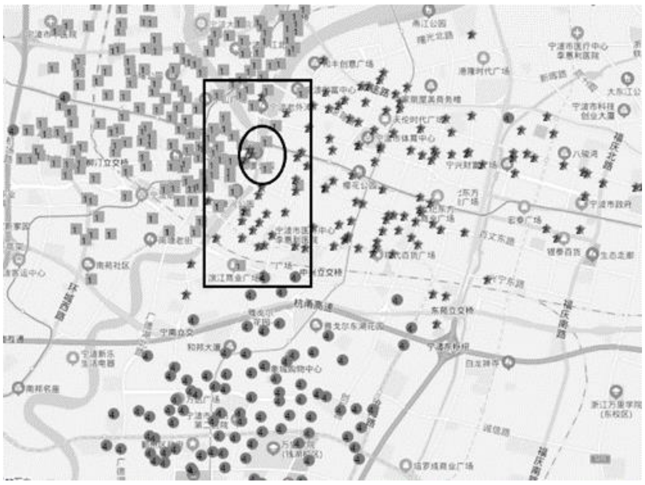

Fig 1: SimRank clustering result (t=0.75)

Fig.1 shows the station clustering analysis results of SimRank algorithm (t=0.75). It can be seen from the figure that although the SimRank cluster realizes the boundary division of cluster 1 (rectangle) and cluster 2 (pentagram) in the large rectangular area. However, because the SimRank algorithm only considers whether there are borrowing and returning records between stations, and does not consider the amount of borrowing and returning, it is difficult to effectively distinguish between areas with high frequency of bicycle use. For example, the cluster division of stations in elliptic region B has some randomness.

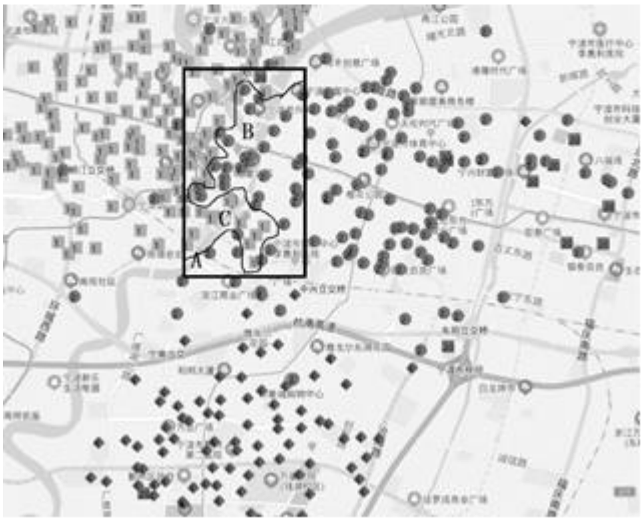

Fig 2: w-SimRank clustering result (t=0.72)

Fig.2 shows the station clustering analysis result of W-SimRank algorithm (t=0.72). It can be seen from the figure that, in the same rectangular region A, w-SimRanK clustering algorithm divides cluster 1 and cluster 2 into jagged shapes (the station of cluster 2 in region B extends to the other side of the river, because the left side of rectangular region A is the commercial square, most residents who come from the right side of A by bicycle will return their bicycles at the station of region B, while the business area of region C is continuous, and many residents will ride bicycles to the station of this region from the left side).

By analyzing the bicycle borrowing and returning records of the stations in rectangular region A, it is found that bicycles in this region are frequently used, which makes the stations in this region have bicycles from multiple stations on both sides or going to multiple stations on both sides. Since SimRank only takes the relationship between borrowing and returning as the feature of the station and ignores the amount of borrowing and returning, the stations with correlation on both sides are easily affected by the number of related stations, rather than the number of borrowing and returning records. After the introduction of weight, w-SimRank can judge the similarity between stations based on the number of borrowing and returning records, thus classifying the cluster more accurately. Therefore, w-SimRank has better clustering effect of shared bicycles.

5.2 Parameter influence

The user-defined parameter in the w-SimRank clustering algorithm is only the parameter t. The influence of this parameter on the algorithm results is analyzed below. Fig.3, Fig.4, Fig.5 and Fig.6 show the clustering results of w-SimRank under different t values.

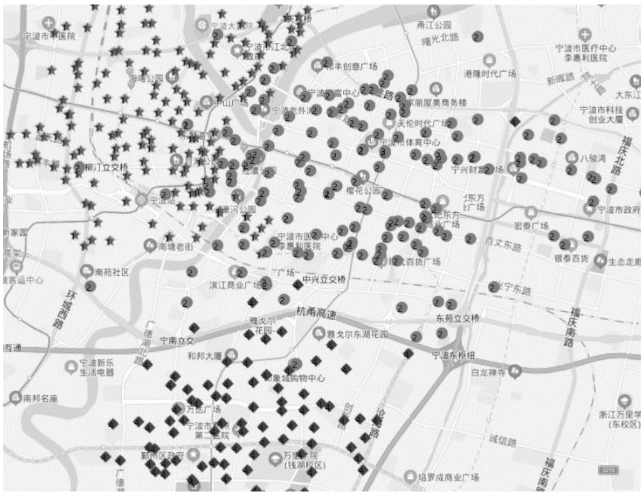

Fig 3: w-SimRank clustering result(t=0.62)

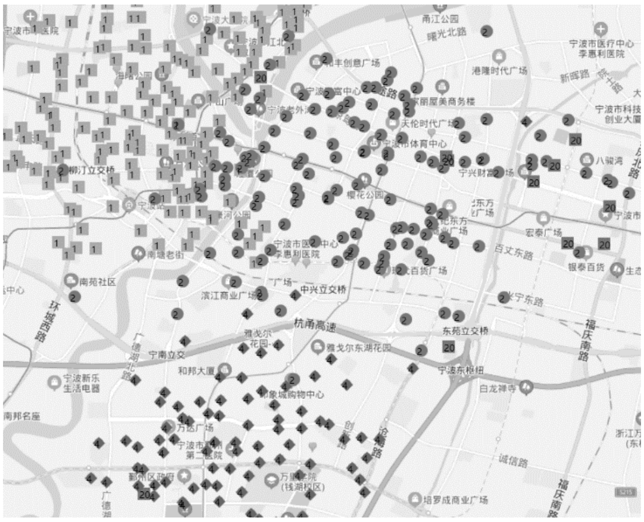

Fig 4: w-SimRank clustering result (t=0.72)

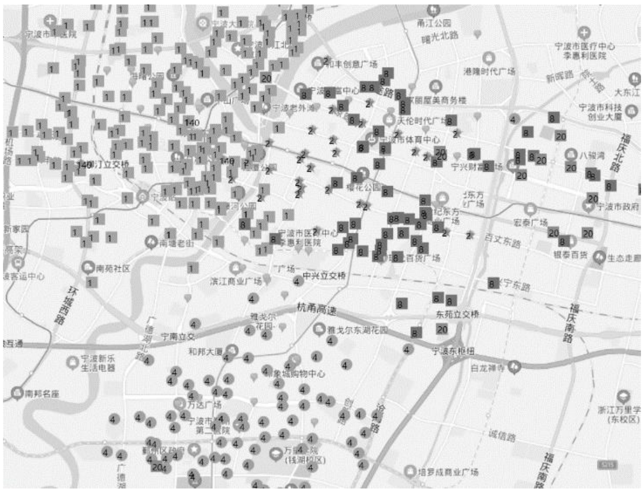

Fig 5: w-SimRank clustering result(t=0.82)

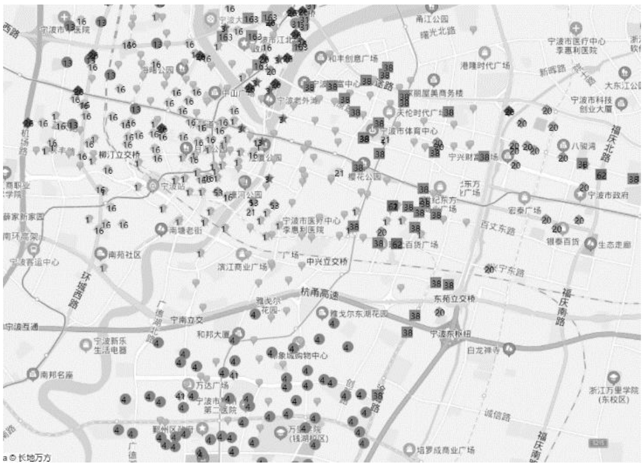

Fig 6: w-SimRank clustering result(t=0.92)

From the four figures, it can be found that with the increase of t, the number of clusters gradually increases, especially after 0.82, a large number of abnormal stations begin to appear. This is related to the definition of parameter t in this algorithm, combined with the normalized processing algorithm of similarity matrix and the clustering idea of w-SimRank algorithm. The parameter t can directly affect the number and quality of the clusters mined. When calculating the similarity matrix, 10% of the stations are defined as having a similarity of more than 0.9. Thus, when the stations are clustered, only 10% of the stations have a chance to form seed stations. Therefore, in this algorithm, analyze the proportion of normalized treatment of similarity matrix and the parameter t is the key. The experiment also proves that the parameter t is about 0.7 has the best clustering result, that is, take 30% of the stations have a high similarity.

VI. Conclusions

With the rapid development of the society, the social traffic and personnel circulation has become a key issue. In order to strengthen the traffic system and facilitate people's travel, in the management of sharing bicycles, it is necessary to know where sharing bicycles should be put and how much they should be put. In this paper, SimRank thought was used to define the similarity of bicycle-sharing stations, and on this basis, a station clustering algorithm based on SkyLine was defined. Experimental results show that this algorithm has high research value.

References

[1] Dong Hongzhao, Shi Caixia, Chen Ning, Liu Dongxu. Clustering of public bicycle scheduling regions based on association rules [J]. Science and Technology Bulletin, 2013, (09): 209-212 + 216.

[2] Y. Chabchoub, C. Fricker, "Classification of the vélib stations using Kmeans, Dynamic Time Wraping and DBA averaging method," 2014 International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM), Paris, pp. 1-5, 2014.

[3] Shen Xingfa, Wang landi. Clustering and function identification of public bicycle system [J]. Computer Engineering, 1-7

[4] P. Jiménez, M. Nogal, B. Caulfield, et al., "Perceptually important points of mobility patterns to characterise bike sharing systems: The Dublin case," Journal of Transport Geography, vol. 54, pp. 228-239, 2016.

[5] X.L. Zhou, "Understanding spatiotemporal patterns of biking behavior by analyzing massive bike sharing data in Chicago," Plos One, vol. 10, no. 10,e0137922, 2015.

[6] D. Lizorkin, P. Velikhov, M.N. Grinev, D. Turdakov," Accuracy estimate and optimization techniques for SimRank computation," PVLDB, vol. 1, no. 1, pp. 408–421, 2008.

[7] W. Yu, X. Lin, W. Zhang, J.A. McCann, "Fast all-pairs SimRank assessment on large graphs and bipartite domains," IEEE Trans. Knowl. Data Eng, vol. 27, no. 7, pp. 1810–1823, 2015.

[8] C. Li, J. Han, G. He, X. Jin, Y. Sun, Y. Yu, T. Wu, "Fast computation of SimRank for static and dynamic information networks," EDBT, pp. 465–476, 2010.

[9] Y. Shao, B. Cui, L. Chen, M. Liu, X. Xie, "An efficient similarity search framework for SimRank over large dynamic graphs," PVLDB, vol. 8, no. 8, pp. 838–849, 2015.

[10] B. Tian, X. Xiao, "SLING: a near-optimal index structure for SimRank," In: SIGMOD Conference, pp. 1859–1874, 2016.

[11] M. Jiang, A.W. Fu, R.C. Wong, K. Wang, "READS: a random walk approach for efficient and accurate dynamic SimRank," PVLDB, vol. 10, no. 9, pp. 937–948, 2017.

[12] Y. Liu, B. Zheng, X. He, Z. Wei, X. Xiao, K. Zheng, J. Lu, "ProbeSim: scalable single-source and top-k SimRank computations on dynamic graphs," PVLDB, vol. 11, no. 1, pp. 14–26, 2017.

[13] Z. Li, Y.Fang, Q. Liu, J. Cheng, R. Cheng, J.C.S. Lui, "Walking in the cloud: parallel SimRank at scale," PVLDB, vol. 9, no. 1, pp. 24–35, 2015.

[14] W. Tao, M. Yu, G. Li, "Efficient top-k SimRank-based similarity join," PVLDB, vol. 8, no. 3, pp. 317–328, 2014.

[15] Z. Zhang, Y. Shao, B. Cui, C. Zhang, "An experimental evaluation of SimRank-based similarity search algorithms," PVLDB, vol. 10, no. 5, pp. 601–612, 2017.

[16] R. Hu, Y. Lu, L. Zou, C. Zhou, "Progressive Subspace Skyline Clusters Mining on High Dimensional Data. In: Washio T. et al. (eds) Emerging Technologies in Knowledge Discovery and Data Mining," PAKDD, Lecture Notes in Computer Science, vol 4819. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-77018-3_28, 2007.