Journal of Network Intelligence ©2022 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **7**, Number **4**, November **2022**

A Novel Feature Generalization Approach for Privacy Protection in Location Based Service

Hui Wang

School of Software
Henan Polytechnic University
Jiaozuo 454000, China
wanghui_jsj@foxmail.com

Hai-Yang Zhang

School of Software
Henan Polytechnic University
Jiaozuo 454000, China
z_h_y133@163.com

Zi-Hao Shen*

School of Computer Science and Technology
Henan Polytechnic University
Jiaozuo 454000, China
hpuxxfzyjs@qq.com

Pei-Qian Liu

School of Software
Henan Polytechnic University
Jiaozuo 454000, China
2362695089@qq.com

Kun Liu

School of Software
Henan Polytechnic University
Jiaozuo 454000, China

Wei Zhen

School of Computer Science and Technology
Henan Polytechnic University
Jiaozuo 454000, China

*Corresponding author: Zi-Hao Shen

Received June 7, 2022, revised September 11, 2022, accepted October 18, 2022.

Abstract. *With the rapid development of mobile devices, rapid positioning has become one of the most popular services in mobile online social networks. In the fast location service, users can enjoy a better social experience by updating their real-time location information. However, the accuracy of the query results and the leakage of personal information may hinder the further development of the service. Hence, a location privacy protection method based on feature generalization is proposed for the existing problems. In this method, an algorithm called dynamic clustering is constructed to improve the construction efficiency of clusters in the anonymous region and simultaneously enable each cluster to meet the needs of K anonymity. Then, using the captain election algorithm and query algorithm, while meeting the user's mobility and flexible privacy protection requirements, accurate query results can be quickly returned to the user. Through theoretical analysis and experiments, it is verified that this method is a safe and effective privacy protection solution.*

Keywords: Privacy protection, Feature generalization, Clustering algorithm, Mobile online social network

1. **Introduction.** With the rapid development of mobile online social networks (mOSNs), location-based service (LBS) has become essential to mobile users' daily lives. According to a recent survey by Statista [1], as of October 2020, there were an estimated 4.08 billion active mOSNs users worldwide, and the number of users using mOSNs is expected to reach approximately 4.4 billion in 2025.

Mobile users upload personal information such as their ID, location, and interests to LBS applications to access appropriate services (e.g., find nearby restaurants, hospitals, etc.) and can also share location-tagged content to find friends with similar interests. LBS in mobile online social networks enrich how people socialize and bring great convenience to their lives. However, while people enjoy the convenience of LBS, they also face the threat of privacy leakage. For example, location service providers may use techniques such as data mining to illegally obtain sensitive information about users, such as home address, religious beliefs, health status, etc., from the location information submitted by users [2]. Therefore, how to provide accurate services to users while protecting their privacy is a central issue at hand.

To address this issue, researchers have proposed several privacy-preserving mechanisms. Wu et al. [3] propose a novel provably secure authentication protocol for protecting sensitive information transmitted in public channels from interception or tampering. Chen et al. [4] devised a verifiable dynamic encryption with ranked search scheme that satisfies forward privacy, which not only helps users to accurately update outsourced data but also protects their privacy. To prevent the leakage of users' privacy in mOSNs and to improve the flexibility of privacy protection, Sun et al. [5] designed a user-defined rapid positioning system to overcome the drawbacks of leakage of users' social network privacy and to prevent location servers from revealing users' complete social network relationships, but the communication overhead of the system is significant. Xiao et al. [6] designed a centralized architecture to solve the privacy problem in rapid positioning systems, which are highly prone to performance bottlenecks due to the heavy computational tasks it carries. Li et al. [7] proposed a scheme that uses multiple location servers (MLS) to protect users' social network privacy, but due to the complexity of the real world, the appropriate placement of multiple location servers is often challenging to achieve. In addition, traditional location privacy protection schemes have mainly focused on basic security requirements in rapid positioning systems, such as protecting the user's identity information and location, with little regard to the efficiency of algorithm execution and the accuracy of queries for each cluster in the anonymized region. It is a pressing issue to design a privacy protection mechanism that provides personalization for users while providing an efficient query service. To solve the above problems, this paper introduces a novel privacy-preserving algorithm that anonymizes LBS requests in a mobile environment. The main contributions of this paper are as follows:

1. This paper uses the dynamic clustering algorithm to divide the anonymous region into multiple clusters. Each cluster uses the captain election algorithm to select the appropriate captain and sends a query to the LBS through the captain. When the users in the anonymity region change, the clusters can be quickly re-clustered by the dynamic clustering algorithm, and each cluster still satisfies the requirement of K-anonymity.
2. The LBS query algorithm by feature generalization reduces the query burden of the location server and also takes the user's mobility into account. For constantly moving users, the clusters are updated when the change of users in the clusters exceeds a certain threshold. Based on the guarantee of execution time and query accuracy, it can effectively prevent attacks based on background knowledge and enhance the security protection of users' location privacy.
3. Through experiments, it is proved that the execution time of the method in this paper is shorter and the query accuracy is higher, while the privacy protection success rate of this paper is higher compared with the location privacy protection schemes in the latest mobile social networks.

The rest of this paper is organized as follows. Section 2 provides existing methods proposed to protect users' privacy of mOSNs. Section 3 introduces the basic definitions and the required related knowledge used in this paper. The details of our proposed scheme are presented in Section 4, and the analysis of the proposed method is discussed in Section 5. In Section 6, we evaluate the algorithms by comparing them with existing techniques. Finally, conclusions and future work are discussed in Section 7.

2. Related Work.

With the popularity of mobile devices, especially smartphones, mOSNs have experienced rapid growth [8]. As an increasingly important service in mOSNs, rapid positioning brings excellent convenience to people. However, privacy issues caused by rapid positioning have also become a pressing issue [9]. Many privacy-preserving methods have been proposed, such as k-anonymity [10, 11, 12], spatial hidingy [13, 14], and encryption-based schemes [15, 16, 17, 18, 19]. Security and efficiency are two essential aspects of location privacy security systems. Therefore, this section presents some of the current location privacy security solutions based on these two aspects.

Currently, many scholars have proposed solutions for the protection of users' sensitive information. Son et al. [20] used a broadcast form to query information about nearby friends and introduced a new cryptographic primitive called functional pseudonym to protect the user's identity. Wu et al. [21] proposed a lightweight and authenticated key agreement protocol to guarantee the security of information transmission over a public channel. To overcome the privacy threat caused by the same location information of the user, Olteanu et al. [22] proposed a game theory framework to explore the relationship between user behavior and location. Lin et al. [23] pointed out that users may provide inaccurate location data in rapid positioning systems, so they designed an attack method to reveal the shortcomings of protecting location privacy in existing rapid positioning mechanisms.

However, the above solutions mainly focus on the basic security requirements for privacy and security protection, such as protecting users' identity privacy and location privacy, but cannot meet users' customized privacy protection needs K. To be able to protect the privacy of users better, Zhang et al. [24] proposed an SCPPS scheme based on the concept of information segmentation and user incentives. In this scheme, it is difficult for collaborating users to obtain information about the requesting user, and it is well protected against collusion attacks. At the same time, the scheme also proposes an incentive mechanism where only the earliest collaborative user who submits partition information and receives corresponding feedback is rewarded, which ensures the activity of anonymous zone construction users and can provide good protection for the information of requesting users. Jin et al. [25] proposed a user-centric location privacy trading framework ULPT and designed heuristic algorithms with limited optimal gaps to reduce the privacy budget while protecting the security of user location privacy.

While protecting user privacy, scholars have proposed some solutions to improve query efficiency. Peng et al. [26] proposed a user-defined rapid positioning scheme that does not require the introduction of a third party and can provide more flexible privacy protection for users in different environments. To improve the transmission efficiency, Shen et al. [27] proposed a scheme called B-Mobishare and used Bloom filters to filter sensitive data in transmission between social network servers and location servers. However, B-Mobishare suffers from high time costs and computational overhead. To prevent location servers from accessing users' complete social network relationships, Xu et al. [28] proposed a PPLS algorithm to secure users' location privacy in mOSNs, which sets different thresholds between users' different friends and does not use a broadcast encryption scheme to avoid location servers from leaking users' sensitive information. Li et al. [29] used the APS algorithm to divide the user's anonymous region into several polygonal regions while ensuring that only one point in each polygon can send a query request to the LBS, and then selected the fake location by the four color theorem and mixed the fake location with the actual location of the requesting user to construct the anonymous region. Yang et al. [30] proposed a SELS algorithm for association grids, which can filter out the locations of users' friends not in the association grid, thus reducing the burden of distance calculation and comparison on the location server. Meanwhile, for some specific geofences or sensitive areas that users do not want to share with their friends, users can use access control policies to prevent privacy leakage in social networks.

Based on the above discussion, the previous approach did not consider the user's characteristic information as a reference attribute of the obfuscation scheme. Moreover, traditional approaches to location privacy protection focus primarily on the user's location rather than the user's anonymity. Therefore, this paper proposes a feature generalization-based approach to location privacy protection. Considering that users in the cluster area are constantly moving, this paper also adds mobility to the model and proposes a dynamic clustering algorithm to provide privacy and security protection for users in mOSNs.

3. **Related Knowledge.** This section provides an introduction to some relevant definitions.

3.1. **Generalization of user features.** User features are unique identifiers that indicate basic information about the user. The primary purpose of this paper is to generalize user features, not to generalize user location information. Therefore, it is necessary to know how much information in the features needs to be generalized. The traditional generalization of user features means that if all the user's data are grouped into a feature set, the user's information is considered fully generalized. However, this will reduce the accuracy of the LBS query results. If an attacker finds no changes in user features, that is, the feature generalization level is 0%, the attacker has a high probability of correctly identifying the user.

Therefore, to address the above problems, this paper proposes a generalization calculation method for user feature information to quantify the generalization of user features and how to protect the location privacy security of users through generalization. The user's feature generalization calculation formula is as follows:

$$uf\left(g_{u_i}\right) = \frac{dist\left(u_i, area_{u_i}\right)}{dist\left(u_i, gf\right)} * 100 \tag{1}$$

where uf is the user's feature, $area_{u_i}$ is the anonymous region where the user is located, and gf is the general feature.

3.2. **Accuracy of LBS queries.** There are two methods for users to send query requests to the LBS: pre-query clustering and post-query clustering. Pre-query clustering is where we perform the clustering algorithm and maintain each user's features, which are used at query time to reduce the latency when the user sends a query request to the LBS. Post-query clustering, on the other hand, is performed after the user sends a query to the LBS.

For LBS queries, the accuracy of the query will be improved if the relevant location information of the user can be obtained. Therefore, this paper uses the priority search algorithm on top of the pre-query algorithm. When a user sends a query request to the LBS, the query results will be sorted according to user characteristics and user interests, and the query results will be further filtered to ensure that the query results can be relevant to the user. In this paper, we use the following formula to ensure the accuracy of the priority search algorithm.

$$Query\ Accuracy = \frac{Number\ of\ relevant\ results\ of\ the\ query}{Total\ number\ of\ search\ results} \times 100 \tag{2}$$

3.3. **Information loss.** The information loss of users is measured by generalizing features to that user. The more features are generalized to the user, the more information is lost, which reduces the accuracy of the query location results. To calculate the information loss, this paper uses $loss_i$ to denote the information loss of a user u_i, and $dist$ to denote the difference between the calculated original features and the generalized features.

$$loss_i = dist\left(u_i, g_{u_i}\right) \tag{3}$$

For all users in the anonymous region, their total information loss is:

$$loss_{total} = \sum_{i=1}^{m} loss_i \tag{4}$$

3.4. **Symbols and definitions.** The key symbols used in this paper and their meanings are shown in Table 1.

Table 1. Symbols and definitions

Symbols	definitions
u_i	User i
c_i	Captain i
r	Radius of the anonymous region
$area_{u_i}$	The anonymous area where user i is located
g_{u_i}	User features after generalization
$loss_i$	Loss of information of user i
K	Users' privacy protection needs

4. Proposed Method.

In practical applications, user locations are constantly changing. Therefore, if global clustering is performed to construct anonymous regions and generate generic features, one or several users near the attacker may still have the feature, which makes the user vulnerable to background knowledge-based attacks. Moreover, the user's anonymity region does not satisfy K-anonymity. Therefore, this paper proposes a method of user feature generalization, which avoids the problems existing in location anonymity, ensures the accuracy and efficiency of continuous queries while satisfying K-anonymity, and can effectively protect the location privacy security of users.

4.1. Overall architecture.

The approach in this paper requires the execution of a local clustering algorithm that combines users based on their personal information but requires them to provide parameters K for constructing anonymous regions. To execute such a clustering algorithm, we need to select an anonymous region and divide the whole region into multiple cells, and each cell also satisfies the condition of K-anonymity. Otherwise, we will continue to expand the region until K users are included. Once a region is identified, we elect a captain to perform the clustering. The election of the captain is based on four factors: credit value, computing power, moving speed, and distance from the region boundary. Once a captain is selected, he will be responsible for a secure equicardinal k-means clustering algorithm and generate generic features. Users then use these generic characteristics to send query requests to the LBS. Meanwhile, considering the mobility of users, this paper proposes an algorithm for dynamic clustering of anonymous regions to consider users' departure, arrival, and return. The system structure is shown in the figure below.

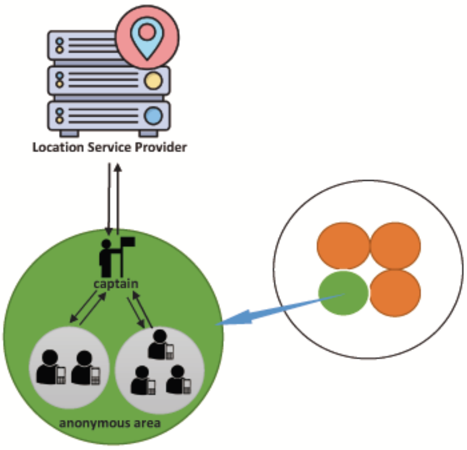

Figure 1. System structure

4.2. Captain election algorithm.

This paper uses the smart contract technology in the blockchain [31, 32] to select the captain c_i of each area. Four parameters are defined as indicators for evaluating c_i, and the trust score formula is as follows:

$$CTS = n\% * cp + (1 - n\%) * cv + \frac{dis}{sp} \tag{5}$$

Where CTS is the trust score of c_i, cp is the computing power of the user, cv is the credit value of the user, dis is the distance from the user to the center of the anonymous region, and sp is the moving speed of the user.

For computing power cp, the implication is to predict whether the mobile user's device can perform clustering and maintain generic features. Although the anonymous region size and number of users may be small, it is still necessary to ensure that the selected mobile user's device is capable of performing the calculation.

Secondly, the user's credit value cv, means the user's credit value under the national credit system, through which a person's creditworthiness can be visualized. The higher the credit value, the lower the probability that the user will disclose the location information of other users. Also, using credit values can reduce the probability of electing attackers as captains, thus reducing the probability of captains leaking sensitive information about collaborating users. Therefore, this paper uses the credit value of mobile users as a metric to evaluate user trust scores. The higher the cv, the lower the probability that the user is an attacker.

Finally, the meaning of sp and dis is to ensure that c_i can stay in the same anonymous region for longer. If c_i is about to leave the region, then the captain selection and clustering algorithm must be executed again. To securely calculate the captain of all users in the network, we use the smart contract technology in blockchain to make the selection. The decentralization and tamper-evident nature of blockchain technology can guarantee users' privacy and security. The specific implementation process is shown in Algorithm 1:

Algorithm 1 Captain Election Algorithm

```
Input: User's credit value $cv$; Computing power of user devices $cp$; Set of all users in the region $U$; The user's encryption key in the blockchain $Enc_{pk}$
Output: Captain $c_i$
1:  $PB \leftarrow Enc_{pk}(0)$;
2:  $c_i \leftarrow Enc_{pk}(0)$;
3:  for $u_i \in U$ do
4:      Calculate $CTS$ based on the user's $cp$ and $cv$;
5:      $DB \leftarrow Enc_{pk}(CTS)$;
6:      $PB \leftarrow$ Smart Contract($PB$,$DB$);
7:      if $PB == DB$ then
8:          $c_i \leftarrow Enc_{pk}(i)$;
9:      end if
10: end for
11: Selecting out trusted captains in the blockchain;
12: Broadcast the information of captain $c_i$ to all users in the region;
13: return;
```

In Algorithm 1, lines 1-2 perform the initialization operation, and lines 3-13 are the main body of the algorithm, where lines 3-4 calculate the trust score CTS for each user u_i in the clustered region, and lines 5-11 select the trusted captain c_i based on the calculated trust score CTS using the blockchain. Lines 12-13 broadcast the information of the captain to all users in the region.

4.3. **Dynamic clustering algorithm for anonymous regions.** In this paper, we divide the whole anonymous region into several sub-regions while making each sub-region satisfy K-anonymity. However, mobile users may not stay in the same region for long. Therefore, we need to set a threshold value to judge the change in the neighborhood, and if the captain c_i is still in the neighborhood, we only need to re-clustering. Otherwise, the captain c_i needs to be reselected, and then the clustering algorithm is executed.

In this paper, the mobile user's speed and direction of movement are considered to calculate the time for the user to leave or arrive at the specified location. Each location must select a c_i access user's features to execute the algorithm. However, it is also possible for c_i to leave this location. Therefore, for each t moment, the algorithm must be executed. However, if the user is moving slowly and the network has not changed, there is no need to repeat the process. Therefore, the process is re-executed only when the network changes and exceeds a certain percentage. In this paper, the speed of the mobile user is considered to calculate the moment t. The average speed $\overline{sp}$ is expressed as follows:

$$\overline{sp} = \frac{1}{n}\sum_{i=1}^{n} sp_i \tag{6}$$

The maximum distance a user can move within a region is $dis = 2r$, and $t = \frac{dis}{\overline{sp}}$, so every time $t = \frac{2r}{\overline{sp}}$ seconds pass, all users in the network need to be scanned once, and if any user's position changes more than n%, then the clustering algorithm will be re-executed. The specific implementation is shown in Algorithm 2.

Algorithm 2 Dynamic Clustering Algorithm

```
Input: User $u_i$; User credit value $cv$; Anonymous area radius $r$; Anonymous regional center $cntr$; User Features $uf$; User moving speed $sp$; Captain $c_i$
Output:The captain of this anonymous region
1:  while $t\%\,(r * n\%/50 * sp)$ == 0 do
2:      if $c_i$==$NULL||c_i$ Leaving the anonymous area==TRUE then
3:          $c_i \leftarrow$ Select captain($cv, r, cntr$);
4:          if $u_i$ == $c_i$ then;
5:          The captain of this anonymous area $\leftarrow$ Clustering($uf$);
6:          end if
7:      end if
8:      if The network has changed $n\%$ then
9:          if $u_i$== $c_i$ then
10:             Clustering($uf$);
11:         end if
12:     end if
13: end while
14: return;
```

In Algorithm 2, line 1 determines the algorithm's execution period and scans all network users, and lines 2-7 determine whether the captain needs to be re-elected and clustered. Only re-clustering is required if the captain is still in the anonymity zone. Otherwise, the captain needs to be re-elected and re-clustered. Lines 8-12 indicate that if the network changes by more than n%, re-clustering is performed.

4.4. **Feature generalization based LBS query algorithm.** The final step of the algorithm is that the user receives the generalized features and sends a query to the LBS. When user u_i is ready to send a query to the LBS, the request is first sent to c_i, which identifies the anonymous region $area_{u_i}$ to which u_i belongs. c_i responds by sending the features of $area_{u_i}$. When user u_i receives the features of $area_{u_i}$, the query is sent to the LBS by the features of $area_{u_i}$ and the real location of u_i. The specific implementation is shown in Algorithm 3.

Algorithm 3 Feature Generalization Based LBS Query Algorithm

```
1. User sends a query request $req_{u_i}$
[$Message1$: $req_{u_i} \rightarrow c_i$]
$< t_p, u_i >$
For the captain of the region $c_i$
Input: Users' privacy protection needs $K$; User Set $U$
while TRUE do
    if $c_i$ receives a query request from $u_i$ then
        $c_i \leftarrow c.find(u_i)$;
        [Message 2: $c_i \rightarrow req_{u_i}$];
        $< t_{p+1}, U, K >$;
    end if
end while
return;
```

Since this paper performs a user-anonymous query algorithm and does not need to hide the actual location of u_i, the query results are very accurate. In addition, if an attacker tries to steal the location privacy of u_i, it will get K user locations with the same features, so all users in that anonymized region satisfy the need for K anonymity, which not only protects the location privacy of users but also improves the accuracy of LBS queries.

5. **Analysis.**

Theorem 5.1. *Within a certain moment, with a certain number of users in the network, when the user's privacy protection needs K satisfies $K = logm$, the user's location privacy security can be protected to the maximum extent.*

Proof: In this paper, we assume that all users in a given anonymity region are equidistant from the center of the anonymity region in which they are located. To make the user's privacy more secure, we need to maximize the privacy requirement parameter K and minimize the user's information loss $loss_i$. Specifically, we need to minimize $loss_i + \frac{1}{K}$.The formula is shown below:

$$\frac{\sum_{i=1}^{K}\sum_{j=1}^{m_i}\left(n_{ij}-\overline{n}_{loss}\right)'\left(n_{ij}-\overline{n}_{loss}\right)}{\sum_{i=1}^{K}\sum_{j=1}^{m_i}\left(n_{ij}-\overline{n}\right)'\left(n_{ij}-\overline{n}\right)}+\frac{1}{K} \tag{7}$$

In this paper, we consider that each anonymous region has the same number of users. Therefore, the number of users in each anonymous region is K and the number of anonymous regions is $\frac{m}{K}$.

$$\frac{\sum_{i=1}^{\frac{m}{K}}\sum_{j=1}^{m_i}\left(n_{ij}-\overline{n}_{loss}\right)'\left(n_{ij}-\overline{n}_{loss}\right)}{\sum_{i=1}^{\frac{m}{K}}\sum_{j=1}^{m_i}\left(n_{ij}-\overline{n}\right)'\left(n_{ij}-\overline{n}\right)}+\frac{1}{K} \tag{8}$$

Since the distances are equal for each user, replacing the distances and summing them yields the following formula.

$$K * \frac{m}{K} * m + \frac{1}{K} \Rightarrow m^2 + \frac{1}{K} \tag{9}$$

Taking the derivative with respect to formula (9) and making it equal to 0 can obtain the following formula:

$$K = logm \tag{10}$$

Therefore, at a certain moment, when the number of users in the network is constant, the closer the user's privacy protection requirement K is to $logm$, the higher the privacy protection security.

Theorem 5.2. *The time when users leave the anonymous area is related to the network change n%.*

$$t = \frac{r * n}{100} * \left(\frac{1}{sp_{max}} + \frac{1}{sp_{min}} \right) \tag{11}$$

where r is the radius of the anonymous region, and sp_{max} and sp_{min} are the maximum and minimum speeds of user u_i, respectively.

Proof: If the user's speed is sp_i, then the time t required for the user u_i to move d is:

$$t = \frac{d}{sp_i} \tag{12}$$

If the requesting user intends to leave the anonymous region, the farthest he can move is the diameter of the anonymous region. Therefore, the time t taken to leave this anonymous region can be simplified to $t = \frac{2r}{sp_i}$. The user who takes the shortest time to cross the same anonymous region is the fastest moving user, and the user who takes the longest time is the slowest user. As shown in Eq. (13)(14):

$$t_{max} = \frac{2r}{sp_{\ min}} \tag{13}$$

$$t_{min} = \frac{2r}{sp_{\ max}} \tag{14}$$

Therefore, the average time for all users in the anonymous region to leave is:

$$t_{avrg} = \frac{t_{max} + t_{min}}{2} = r * (\frac{1}{sp_{max}} + \frac{1}{sp_{min}}) \tag{15}$$

In summary, when a user leaves the anonymous region he is in, the time required is $t = \frac{r*n}{100} * (\frac{1}{sp_{max}} + \frac{1}{sp_{min}})$.

6. **Experimental Results and Discussion.** This experiment was carried out on a laptop computer with Intel(R) Core(TM) i7-9750H CPU, 16G memory, and Windows 10 64-bit operating system. This experiment uses the Geolife Trajectories 1.3 dataset, which contains the user's ID, longitude, latitude, altitude, road number, date, time, and point of interest. The attribute information in this dataset is also used to calculate the speed and direction of the mobile user's travel.

The experiments compare the PPLS algorithm, the APS algorithm, and the SELS algorithm. The efficiency of the algorithm in this paper is proved by comparing the accuracy of the LBS query and the execution time of the algorithm, and the security of the algorithm in this paper is proved by comparing the success rate of user privacy protection.

6.1. **Impact of the number of clusters in anonymous regions on query accuracy and execution time.** More clusters in an anonymous region mean fewer users in each cluster while reducing the number of clusters will increase the number of users in each cluster. Therefore, by increasing the number of clusters, the relevance of the users in the clusters can be improved. It can make LBS query results more accurate while improving generalizability. The results of the effect of the number of clusters on the query accuracy are shown in Figure 2.

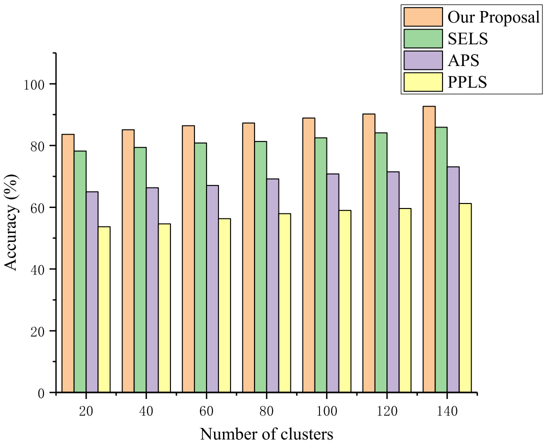

Figure 2. The influence of clustering number on query accuracy

For the PPLS algorithm used in scheme [28], when the number of anonymous regions increases, the communication between users and friends will be redundant, reducing the query accuracy of LBS. At the same time, the requesting user may send query requests to the same friend multiple times, which increases the execution time of the algorithm.

The scheme [29] uses the APS algorithm to divide the user's anonymous region into several polygonal regions, where only one fixed location in each region can issue a query request. However, when the number of clusters continues to increase, to improve the user's privacy and security, the APS algorithm makes the fixed point coordinates and the user's actual location far away, causing information loss to the user. At the same time, each cluster's size and fixed point need to be recalculated, which significantly increases the execution time.

Scheme [30] proposes a SELS algorithm for the association grid, which filters out the locations of user friends that are not in the association grid. When the number of clusters in the anonymous region increases, the number of friends in each cluster will decrease, and the relevance of the results returned by the LBS to the user will decrease when the requesting user sends a query request. The results of the execution time of each algorithm are shown in Figure 3:

When the number of clusters in the anonymous region increases, the scheme [30] cannot divide the clusters efficiently and increases the algorithm's execution time. When increasing the number of clusters in anonymous regions, this paper shortens the execution time of the algorithm by using a dynamic clustering algorithm to efficiently divide the sub-regions, using a captain election algorithm to elect a suitable captain quickly and simultaneously send a query request to the LBS, based on ensuring the accuracy of the

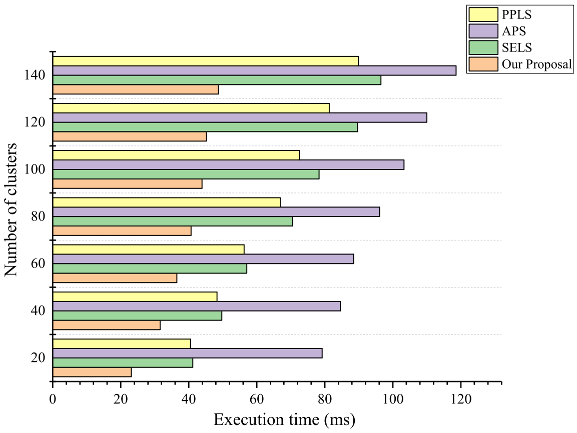

Figure 3. The influence of cluster number on execution time

LBS query, when the number of clusters k=60, the LBS query accuracy of the method in this paper can reach 86.4%, while the average execution time of the algorithm can be guaranteed to be within 70ms.

6.2. **Impact of the number of users in the network on query accuracy and execution time.** For the variation of the number of users in the network, this paper aims to compare with the crowd sparse environment, so the number of anonymous regions, the number of clusters, and the average movement speed of users are kept constant. It is specified that all experiments use an anonymous region with a number of clusters equal to 60, and the average movement speed of users is 10 km/h. It is demonstrated through a large number of experiments that users have better performance in terms of LBS query accuracy and anonymity when the number of clusters is equal to 60. Therefore, the query accuracy of each algorithm is compared in this setting by comparing the number of users in the network, and the results are shown in Figure 4.

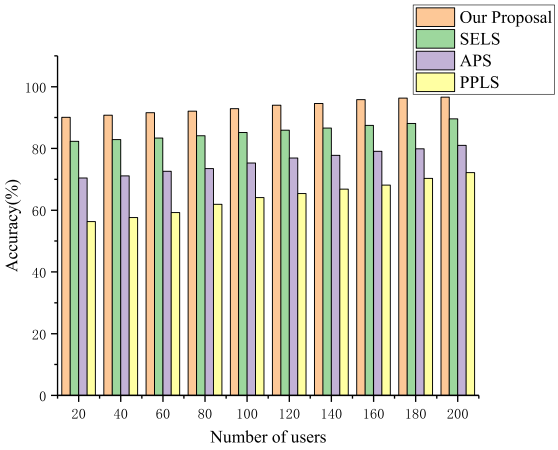

Figure 4. The influence of users quantity on query accuracy

For the PPLS algorithm of the scheme [28], when the number of users in the network increases, the number of friends of the requesting users will decrease, the relevance of the users will decrease, and the query accuracy will decrease for the returned query results.

At the same time, each time a new user is added, the threshold between the requesting user and each user needs to be recalculated, dramatically increasing the overhead.

The scheme [29] uses the APS algorithm to divide the anonymous region of users into several polygonal regions. When the number of users in the network increases, the clusters need to be continuously redivided, and the fixed points need to be recalculated for the divided clusters, so the algorithm's execution time keeps increasing with the number of users.

The scheme [30] proposes a SELS algorithm for associative grids, which increases the overhead when the number of users in the network increases, and more users need to be added to the access control policy to avoid disclosing their sensitive information. In this paper, the smart contract technology and feature generalization-based LBS query algorithm shorten the algorithm's execution time while ensuring the LBS query's accuracy. The execution time results are shown in Figure 5.

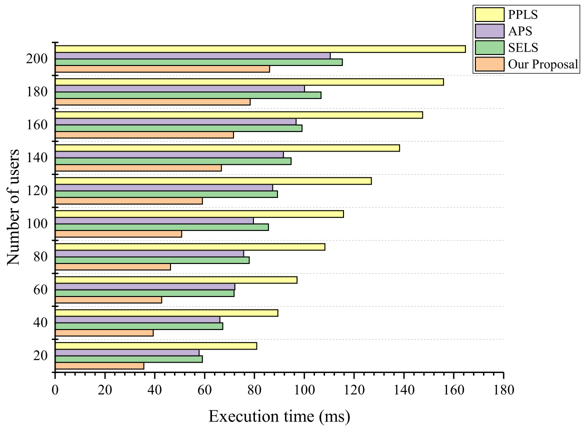

Figure 5. The influence of users quantity on execution time

6.3. **Privacy Protection Success Rate.** To verify the success rate of privacy protection, this paper assumes that there is an anonymous region in which 10 people send query requests to the LBS at the same time, and the number of collaborating users is half of the number of people in the anonymous region. The change in privacy protection success rate as the user privacy protection demand K increases is shown in Figure 6.

As can be seen from Figure 6, the scheme [28] needs to communicate with friends. As the K increases, the number of friends in the cluster gradually decreases, and the risk of information leakage from the requesting user gradually increases. When the number of users is too large, the success rate of privacy protection will drop significantly. Scheme [29] uses the APS algorithm to construct a polygon and selects a fixed point to send a query request to the LBS. Under the assumption of this paper, the number of polygonal areas of this algorithm is fixed, so when K continues to increase, the number of dummy positions generated by using the four color theorem decreases, and the position of each query request to LBS will not change. Therefore, the success rate of privacy protection will continue to decrease with the increase of K. The SELS algorithm used in Scheme [30] uses an access control policy to reduce user privacy leakage while filtering out the locations of users' friends who are not in the associated grid. Therefore, when the user's privacy protection demand K increases, the execution time of the SELS algorithm will increase significantly, but it still has a good performance for the user's privacy protection. Through the generalization of user features, the selection of appropriate captains, and the use of the

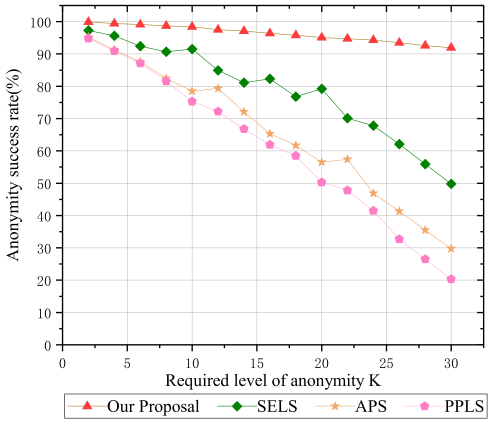

Figure 6. Privacy protection success rate

LBS query algorithm based on feature generalization, this paper can ensure the accuracy and efficiency of the query, and at the same time, with the continuous increase of the K, it can still ensure the user's The success rate of privacy protection is over 90%.

7. **Conclusion.** Distributed location privacy protection schemes tend to cluster users based on their locations. If one of the cluster's users contains a feature entirely different from the others, an attacker can identify the exact user from the cluster using some simple background knowledge attacks. Again, since the locations are generic, the LBS query results do not provide accurate results. Therefore, this paper proposes a feature generalization-based approach to location privacy protection. This method anonymizes the user based on their features and sends the actual location for LBS queries. Considering the mobility of users, users will be regrouped when the initial clustering is no longer valid. The experimental results show that compared with other location privacy protection techniques, the algorithm provided in this paper can stabilize the success rate of privacy protection for users at more than 90%, while the accuracy rate is about two times higher. However, the algorithm in this paper still has some improvement space, and further research is needed on how to exclude anomalies and improve the relevance of clusters in anonymous regions in mobile social networks.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (61300216), Doctoral Scientific Fund of Henan Polytechnic University(B2022-16) and Youth Fund of Henan Polytechnic University(Q2014-05).

REFERENCES

[1] Mobile social media worldwide - Statistics & Facts. Feb 8, 2022
Available: https://www.statista.com/topics/2478/mobile-social-networks/.

[2] X. Shi, Z. Yu, Q. Fang, and Q. Zhou, "A visual analysis approach for inferring personal job and housing locations based on public bicycle data," *ISPRS International Journal of Geo-Information*, vol. 6, no. 7, pp. 205–220, 2017.

[3] T.-Y. Wu, Z. Lee, L. Yang, and C.-M. Chen, "A provably secure authentication and key exchange protocol in vehicular ad hoc networks," *Security and Communication Networks*, vol. 2021, 9944460, 2021.

[4] C.-M. Chen, Z. Tie, E. K. Wang, M. K. Khan, S. Kumar and S. Kumari, "Verifiable dynamic ranked search with forward privacy over encrypted cloud data," *Peer-to-Peer Networking and Applications*, vol. 14, no. 5, pp. 2977–2991, 2021

[5] G. Sun, Y. Xie, D. Liao, H. Yu, and V. Chang, "User-defined privacy location-sharing system in mobile online social networks," *Journal of Network and Computer Applications*, vol. 86, pp. 34–45, 2017.

[6] X. Xiao, C. Chen, A. K. Sangaiah, G. Hu, R. Ye, and Y. Jiang, "Cenlocshare: A centralized privacy-preserving location-sharing system for mobile online social networks," *Future Generation Computer Systems*, vol. 86, pp. 863–872, 2018.

[7] J. Li, H. Yan, Z. Liu, X. Chen, X. Huang, and D. S. Wong, "Location-sharing systems with enhanced privacy in mobile online social networks," *IEEE Systems Journal*, vol. 11, no. 2, pp. 439–448, 2015.

[8] J. Liu, L. Fu, X. Wang, F. Tang, and G. Chen, "Joint recommendations in multilayer mobile social networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 10, pp. 2358–2373, 2019.

[9] H. Jiang, J. Li, P. Zhao, F. Zeng, Z. Xiao, and A. Iyengar, "Location privacy-preserving mechanisms in location-based services: A comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 1, pp. 1–36, 2021.

[10] Y. Yan, E. A. Herman, A. Mahmood, T. Feng, and P. Xie, "A weighted k-member clustering algorithm for k-anonymization," *Computing*, vol. 103, no. 10, pp. 2251–2273, 2021.

[11] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, "A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services," *Future Generation Computer Systems*, vol. 94, pp. 40–50, 2019.

[12] P. Zhang, J. Li, F. Zeng, F. Xiao,C. Wang and H. Jiang, "ILLIA: Enabling k-anonymity-based privacy preserving against location injection attacks in continuous LBS queries," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1033–1042, 2018.

[13] J. W. Kim, K. Edemacu, and B. Jang, "Privacy-preserving mechanisms for location privacy in mobile crowdsensing: A survey," *Journal of Network and Computer Applications*, pp. 1–19, 2022.

[14] T. Zhou, Z. Cai, B. Xiao, L. Wang, M. Xu and Y. Chen, "Location privacy-preserving data recovery for mobile crowdsensing," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 3, pp. 1–23, 2018.

[15] T.-Y. Wu, X. Guo, Y.-C. Chen, S. Kumari and C.-M. Chen, "Amassing the Security: An Enhanced Authentication Protocol for Drone Communications over 5G Networks," *Drones*, vol. 6, no. 1, pp. 10, 2022.

[16] J. Chen, K. He, Q. Yuan, M. Chen, R. Du and Y. Xiang, "Blind filtering at third parties: An efficient privacy-preserving framework for location-based services," *IEEE Transactions on Mobile Computing*, vol. 17, no. 11, pp. 2524–2535, 2018.

[17] T.-Y. Wu, Z. Lee, L. Yang, J.-N. Lou and R. Tso, "Provably Secure Authentication Key Exchange Scheme Using Fog Nodes in Vehicular Ad-Hoc Networks," *The Journal of Supercomputing*, vol. 77, pp. 6992-7020, 2021.

[18] J. Zhang, F. Yang, Z. Ma, Z. Wang, X. Liu and J. Ma, "A decentralized location privacy-preserving spatial crowdsourcing for internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2299–2313, 2020.

[19] T.-Y. Wu, X. Guo, Y.-C. Chen, S. Kumari and C.-M. Chen, " SGXAP: SGX-Based Authentication Protocol in IoV-Enabled Fog Computing," *Symmetry*, vol. 14, no. 7, pp. 1393, 2022.

[20] J. Son, D. Kim, R. Tashakkori, A.-O. Tokuta, and H. Oh, "A new mobile online social network based location sharing with enhanced privacy protection," in*2016 25th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2016, pp. 1–9.

[21] T.-Y. Wu, X. Guo,L. Yang, Q. Meng and C.-M. Che, "A Lightweight Authenticated Key Agreement Protocol Using Fog Nodes in Social Internet of Vehicles," *Mobile Information Systems*, vol. 2021, pp. 3277113, 2021.

[22] A.-M. Olteanu, M. Humbert, K. Huguenin, and J.-P. Hubaux, "The (Co-)Location Sharing Game," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 2, pp. 5–25, 2019.

[23] T.-L. Lin, H.-Y. Chang, and S.-L. Li, "A location privacy attack based on the location sharing mechanism with erroneous distance in geosocial networks," *Sensors*, vol. 20, no. 3, pp. 918–934, 2020.

[24] L. Zhang, D. Liu, M. Chen, H. Li, C. Wang, Y. Zhang, and Y. Du, "A user collaboration privacy protection scheme with threshold scheme and smart contract," *Information Sciences*, vol. 560, pp. 183–201, 2021.

[25] W. Jin, M. Xiao, L. Guo, L. Yang, and M. Li, "Ulpt: A user-centric location privacy trading framework for mobile crowd sensing," *IEEE Transactions on Mobile Computing*, pp. 1–18, 2021.

[26] T. Peng, J. Liu, G. Wang, Q. Liu, J. Chen, and J. Zhu, "A user-defined location-sharing scheme with efficiency and privacy in mobile social networks," *Scientific Programming*, vol. 2020, pp. 1–13, 2020.

[27] N. Shen, K. Yuan, J. Yang, and C. Jia, "B-mobishare: Privacy-preserving location sharing mechanism in mobile online social networks," in *2014 Ninth International Conference on Broadband and Wireless Computing, Communication and Applications*. IEEE, 2014, pp. 312–316.

[28] C. Xu, X. Xie, L. Zhu, K. Sharif, C. Zhang, X. Du, and M. Guizani, "Ppls: a privacy-preserving location-sharing scheme in mobile online social networks," *Science China Information Sciences*, vol. 63, no. 3, pp. 1–11, 2020.

[29] W. Li, C. Li, and Y. Geng, "Aps: Attribute-aware privacy-preserving scheme in location-based services," *Information Sciences*, vol. 527, pp. 460–476, 2020.

[30] G. Yang, S. Luo, Y. Xin, H. Zhu, J. Wang, M. Li, and Y. Wang, "A search efficient privacy-preserving location-sharing scheme in mobile online social networks," *Applied Sciences*, vol. 10, no. 23, pp. 8402–8421, 2020.

[31] Q. Mei, H. Xiong, Y.-C. Chen and C.-M. Chen, "Blockchain-Enabled Privacy-Preserving Authentication Mechanism for Transportation CPS With Cloud-Edge Computing," *IEEE Transactions on Engineering Management*, pp. 1–12, 2022.

[32] Z. Bao, W. Shi, S. Kumari, Z.-Y Kong and C.-M. Chen, "Lockmix: a secure and privacy-preserving mix service for Bitcoin anonymity," *International Journal of Information Security*, vol. 19, no. 3, pp. 311-321, 2020.