Research on the Personalized Trajectory Privacy Protection Scheme for Mobile Social Network Based on Sensitivity Degree

Chun-yuan Li
Department of Mathematics, Baicheng Normal University, Baicheng, China
lichunyuan521@126.com

Keywords: Sensitivity degree, mobile social network, trajectory privacy protection

Abstract. In order to deal with the problem of trajectory privacy disclosure in mobile social network, this paper proposes a personalized mobile social network trajectory privacy protection scheme based on sensitivity degree. First of all, according to the various types users of different privacy needs, the appropriate protection principles are selected, so as to make up for the traditional privacy protection in the existence of excessive protection or low trajectory defects. Then, the sensitive trajectory anonymous and other related concepts are defined. This scheme was based on some technologies on Tire tree, including construction, pruning and reconstruction. Finally, it evaluated this scheme on real-world data set, the experiment results show that our scheme is better in trajectory location loss ratio and more efficient than state-of-the-art for privacy-preserving.

Introduction

With the widespread use of smart phones and social networks, the current trend of mobile Internet development is "SoLoMo" model, that is, social, location and mobile integration [1]. This trend promotes WeChat, facebook, Sina, QQ space and other social applications rapidly developed, so that people can share their information anytime and anywhere. Among them, the location of the track is a very important information. People share information at the same time, also faced with privacy information security issues [2]. These locations and their connected trajectory usually contain user privacy information, which provides an opportunity for some malicious acts to use the hidden value and interests. The position and the trajectory have an intrinsic link, and the path where the individual moves in the spatial position is called the trajectory [3]. Therefore, the trajectory privacy protection can be understood as both to ensure that the trajectory itself does not reveal the sensitive position, but also to prevent through its trajectory derived other sensitive information. This issue has been the concern of domestic and foreign experts and scholars, this paper will also be on this issue to study.

System Structure

First of all, this paper introduces and analyzes the system structure of personalized trajectory privacy protection process, as shown in Figure 1. Based on the three principles of first collection, privacy protection and finally release, the basic structure of the system including track data provider, trusted third party and data user is constructed. In particular, it is necessary to explain these two concepts: the ability of attackers and the protection. Attacker ability, that is assumed that a malicious person has the ability to infer the true identity of the user's individual or some sensitive privacy information and knowledge through location or trajectory. Protect the objectives and ideas. Personalized needs for individual users, personalized privacy for mobile social network location or trajectory.

Copyright © 2018, the Authors. Published by Atlantis Press.
This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

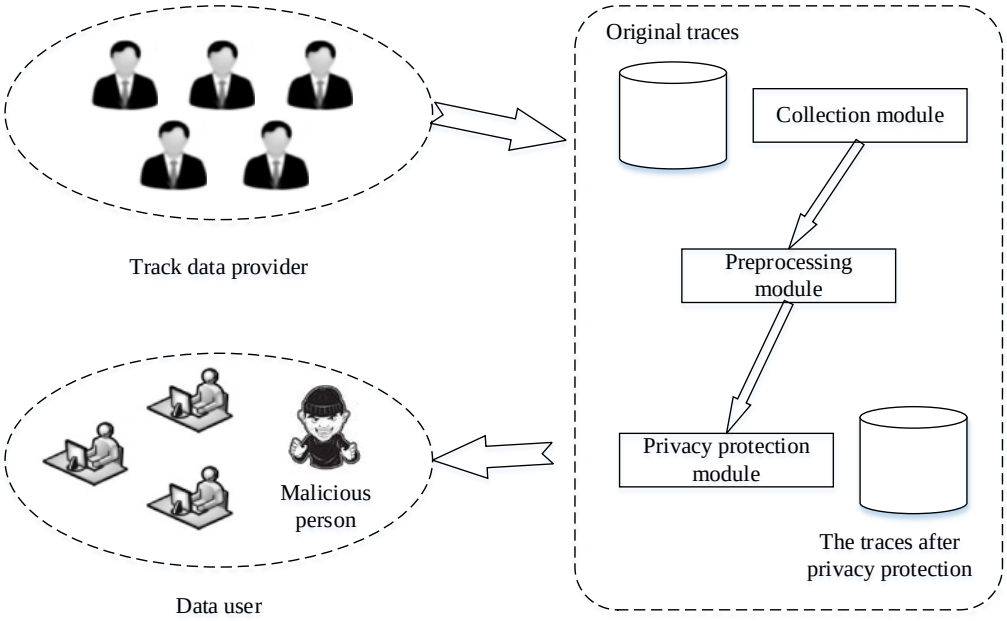

Figure 1. System structure

The Design of Personalized Trajectory Privacy Protection Scheme

In the standard trajectory privacy scheme, the trajectory privacy protection strategy adopted is unified, which leads to the lack of flexibility and other shortcomings [4]. Then, in real life, different individuals exist the different privacy protection needs, that is, some users are very focused on the individual trajectory privacy, and some individual users are not very concerned on the track privacy protection. In addition, the uniform standard of the track privacy protection cannot be achieved with a high degree of privacy protection after the track data is highly available [5]. Based on this, this paper presents a sensitivity-based personalized trajectory privacy protection scheme to address the differences in individual privacy requirements.

According to the degree of sensitivity, this paper divides the privacy protection needs of social network users into four categories, namely, the no need for protection, the need for primary protection, the need for intermediate protection and the need for advance protection. The details are shown as follows:

(1) No protection need (P0). Individual positions and trajectories are fully public and do not require privacy protection.

(2) The need for primary protection (P1). Individuals specify that certain sensitive locations can not be known to the outside world, but they do not have privacy protection requirements for trajectories.

(3) The need for intermediate protection (P2). Individuals specify that certain sensitive locations can not be known to the outside world, and that all their tracks are subject to privacy guarantees (see definition 3 for formal definitions).

(4) The need for advance protection (P3). Individuals specify certain sensitive locations where all traces are guaranteed by privacy and that the probability of a particular sensitive trajectory is within the specified likelihood.

Then, some concepts related to the design of the track privacy protection scheme are explained.

(1) Actual position sequence. The user's record at a certain location can be expressed as l_i, and the corresponding specific location is AL. The actual position sequence is a chronological record of the location, which can be expressed as $LC_i = \{l_1, l_2, l_3, \cdots, l_n\}$.

The actual trajectory of an individual a_i is a path in which the location is connected by time, expressed as $aP_i(X) = l_1 \to l_2 \to \cdots \to l_n$, where X represents the privacy protection requirement flag (P0, P1, P2, or P3) of the individual. The sum of the actual trajectories of n individuals is

$aD = \{aP_1(X), aP_2(X), aP_3(X), \cdots, aP_n(X)\}$.

(2) Trie tree. The Trie tree PTr is expressed as a triple set $PTr = (M, S, R(PTr))$, where M represents the set of Trie tree nodes, S represents the set of Trie trees line, and $R(PTr) \in M$ is the virtual root node of the Trie tree.

In Trie tree, all nodes except the virtual root node have a unique parent node and have a unique path from the root node to the node. All nodes except the root node can be saved in this form $\langle AL, it, su, ch \rangle$, where AL is the identifier of node *m*, *it* is the location record of node, *su* is the support of node, and *ch* is the child node of node *m*.

(3) *k* sensitive trajectory anonymous. The sum of the user trajectories aD is given, and if only the trajectories of any user are present at least the $k-1$ tracks are the same, so that the probability of their path privacy disclosure is less than or equal to $1/k$.

(4) Privacy protection Trie tree. Privacy protects Trie tree ensures that all trajectories in the prefix tree satisfy the *k*-sensitive trajectory. The support for all nodes except the root node is greater than or equal to *k*.

(5) (k, p) sensitive trajectory anonymous. The user trajectory satisfies the *k*-sensitive trajectory anonymously, and the probability that the user's individual pre-specified special sensitive traces exist is less than or equal to *p*.

The following will detail the design of the trajectory privacy protection, including two algorithms, namely PriPTCon and TraPri.

(1) PriPTCon algorithm

Input: *aD* //The sum of the individual trajectories
k //Sensitive trajectory anonymous parameters
Output: PTr^k, L_{cut} //Privacy protection Trie tree and pruning branches

```
1   PTr = {}; //Trie tree structure
2   L_cut = {}; //Initialization is empty
3   for each aP_i(P2orP3) in aD do
4       LPTr = LongestTrie (R (PTr), aP_i (P2orP3));
        // The search has formed the longest prefix in the Trie tree that conforms to the aP_i (P2orP3) trajectory
5       append aP_i ( P2orP3) to PTr; // Add the full path aP_i (P2orP3) to the Trie tree
6   for each node m_i in LPT do
7       m_i.su = m_i.su + 1;
8   for each node m_i in aP_i( P2orP3) but not in LPT do
9       m_i.su = 1;
10      for each node m_i in R(PTr).ch do
11          if m_i.su < k then
12              L_cut = the set of all path in path(PTr, m_i) ;
                // path (PTr, m_i) represents all trajectories from the root node through m_i
13              for each m_j in each path(PTr, m_i) do
14                  m_j.su = m_j.su - m_i.su;
15                  PT r= PTr \ the subtree induced by m_i; // Delete the subtree with the m_i node
16          else
17              for each m_j ∈ m_i.ch do
18                  L_cut = L_cut ⋃ execute 11 ~ 18; //Follow steps 11~18
19  return (PTr^k and L_cut)
```

PriPTCon algorithm realizes the actual construction of the privacy protection Trie tree (Definition 4) by entering the sum of the individual trajectories and the anonymous parameter *k* of the sensitive trajectory. It outputs the privacy protection Trie tree and trimmed branches collection. The algorithm firstly selects the individual trajectory of the privacy protection requirement I or A in

the sum of the individual trajectories, and constructs the Trie tree (line 3 ~ 9). Line 4 and 5 search for the longest prefix that matches the trajectory in the constructed Trie tree and add the full path of the trace to Trie. Line 6 ~ 9 update the support of the node. Line 10 ~ 18 achieve the pruning of the Trie tree that has been built on the Trie tree privacy protection Trie tree conversion. From each sub-node of the virtual root node to determine whether its support meets the requirements of the sensitive trajectory anonymous parameters. If the requirements are met, recursively find their child nodes until the nodes that are not satisfied are found to perform the pruning operations of lines 11 ~ 18.

(2) TraPri algorithm

Input: PTr^k, L_{cut} , p

Output: aD^P //The Trajectory after personalized privacy protection

1 for each $S_i \in L_{cut}$ do
2 LCS = longest common subsequence (S_i, PTr^k) ;
3 L = shortest prefix (PTr^k, LCS) ;
4 if L is not empty then
5 for each node $m_i \in L$ do
6 m_i.su = m_i.su + 1;
7 update aD;
8 for each aP_i(P3) in aD do
9 if (| aP_i(P3) contain PL_i | / | aP_i(P3) | > p) then
10 random transformation, until| aP_i(P3) contain PL_i | / | aP_i(P3) | < p;
11 return aD^P

TraPri algorithm realizes the reconstruction of the pruning trajectory in the privacy protection module and the privacy protection of the privacy protection requirement flag. It inputs privacy protection Trie tree, the set of branches cut off and the probability threshold pre-specified by individuals. And it outputs the trajectory after personalized trajectory privacy protection. Line 1 ~ 6 achieve the reconstruction of the Trie tree. Next, the protection of the (k, p) sensitive trajectory anonymity is made for each individual with a privacy requirement of P3, which is implemented by line 8 ~ 10. The algorithm randomly removes the position of the special sensitive trajectory or changes the position of the aD to the position of the aD_i until the aP_i (P3) contains the number of PL_i tracks divided by the number of aP_i (P3) tracks less than p. This ensures that the probability of a particular sensitive trajectory is within the specified likelihood.

Example Analysis

Here we make an example analysis for the proposed personalized trajectory privacy protection scheme on the performance and efficiency. Experiments are performed on the Gowalla dataset, and the contrast algorithm is Trajectory k-anonymity [6]. The Gowalla dataset is a social networking site where users sign a shared location, collecting 5946280 check-in records between April 2015 and December 2016. It exists 203215 nodes and 963524 side. In this paper, the trajectory position loss ratio (TPLR) is used as a measure of the availability of trajectory data generated by the personalized trajectory privacy protection scheme. The trajectory position loss rate measures the ratio of the position change before and after the track privacy protection, which can be expressed as:

$$\mathrm{TPLR} = \frac{\left|\wedge\left(aP_i^{\,p}(\mathrm{P1orP2orP3}), aP_i(\mathrm{P1orP2orP3})\right)\right|}{aP_i(\mathrm{P1orP2orP3})} \tag{1}$$

Where $\left|\wedge(X,Y)\right|$ represents the number of position differences in the contrast between the X and Y trajectories, $aP_i(X)$ and $aP_i^{\,p}$ respectively represents the trajectory path before and after the privacy protection of individual a_i. The calculation of the trajectory position loss rate is the number of differences in the position of the track positions before and after the privacy protection is divided by the number of all positions before privacy protection. The position loss rate of the track is in the range of 0 to 1. If the track position before and after the privacy protection does not change,

then the track position loss rate is 0.

In this paper, we choose *p* with 0.3 and 0.5 two parameter values for the program simulation, and the track loss rate changes is shown in Figure 2. As can be seen from the figure, the individualized scheme of different parameters have obvious advantages in the trajectory position loss rate, and the variation trend of the trajectory position loss rates are different in different *p* values. The reason for this is that personalized protection program does reduce the unified standard under the protection of the problem. The Trajectory *k*-anonymity scheme implements the trajectory *k* by an iterations, and reconstructs the original data set at random. Therefore, the performance of the program trajectory position loss rate is not ideal. It can be seen from Figure 2 that when the parameter *k* value is above 15, only the trajectory location loss rate of this scheme is controlled below 50%. It can be seen that the scheme has obvious advantages in the track position loss rate.

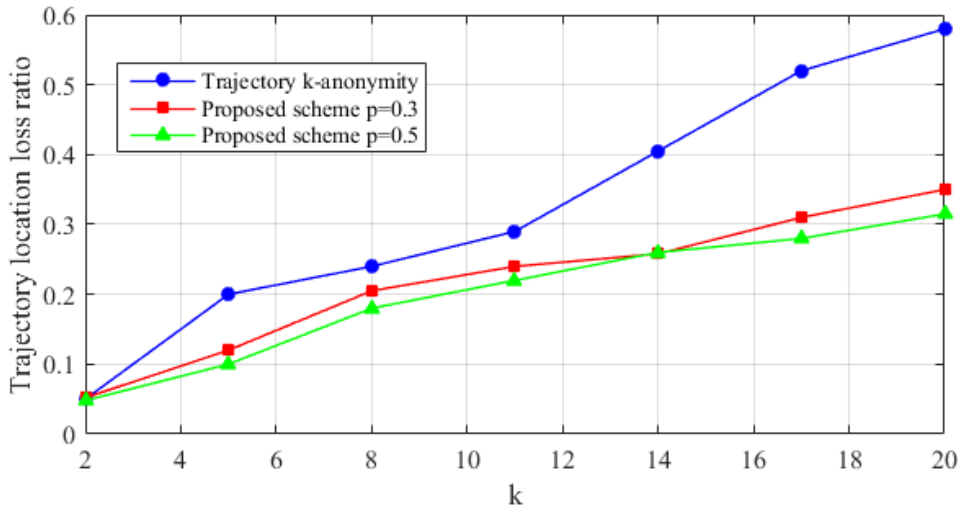

Figure 2. Evaluation of track location loss rate

The calculation delay is analyzed, shown in Figure 3. As can be seen, the computational delay of all algorithms increases with the increase of privacy protection parameter *k*. The time complexity of the complex clustering generation is $O(n^3)$ based on the generalized Trajectory *k*-anonymity scheme, and the time complexity of $O(l^2)$ is still required for each ERP dynamic programming algorithm. Therefore, the calculation delay is relatively high. In this paper, the personalized privacy protection method is better than the contrast scheme in calculating the delay.

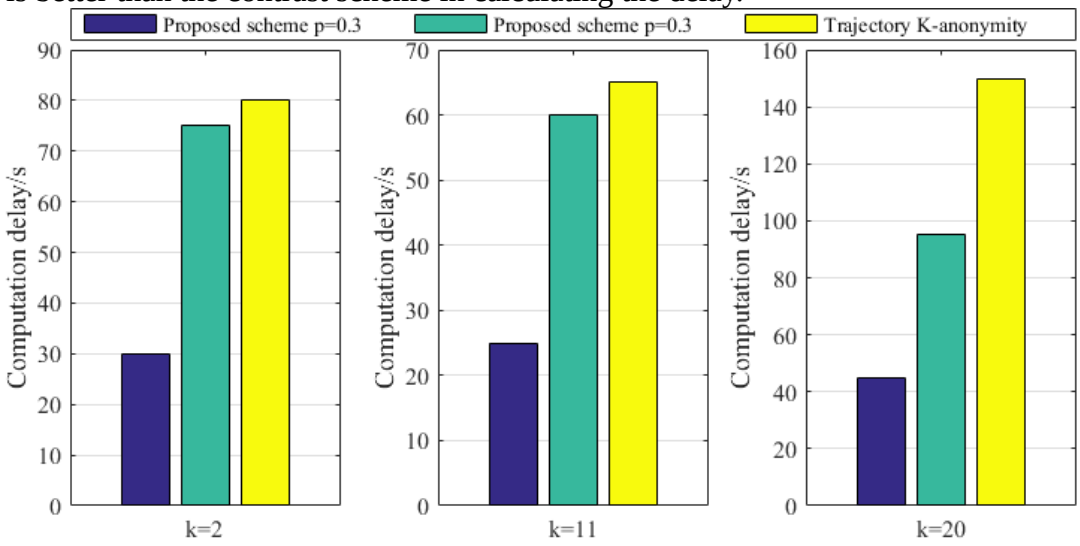

Figure 3. Calculation delay evaluation

Conclusion

This paper puts forward the personalized privacy protection scheme for the mobile social network.

Based on the comprehensive analysis, the proposed scheme is superior to the two performance schemes in terms of performance and efficiency, and can be customized to protect the demand and flexibility. However, this scheme also has some limitations, the scheme on the sensitive position of the replacement is only random, and did not take into account the random replacement of its quality of service impact. The next step should be to consider a more appropriate location replacement algorithm to achieve trajectory privacy protection.

References

[1] Bonchi F, Lakshmanan L, Wang H. Trajectory anonymity in publishing personal mobility data[J]. ACM SIGKDD Explorations Newsletter, 2011, 13(1): 30-42.

[2] Chen Rui, Fung B C M, Mohammed N, et al. Privacy-preserving trajectory data publishing by local suppression[J]. Information Sciences, 2013, 231: 83-97.

[3] Nergiz M, Atzori M, Saygin Y, et al. Towards trajectory anonymization: a generalization-based approach [J]. Trans on Data Privacy, 2009, 2(1): 47-75.

[4] Li Hongjuan, Cheng Xiuzhen, Li Keqiu, et al. Efficient customized privacy preserving friend discovery in mobile social networks [C] // Proc of the 35th International Conference on Distributed Computing Systems. IEEE Press, 2015: 225-234.

[5] Shen Haiying, Lin Yuhua, Chandler H. An interest-based per-community P2P hierarchical structure for short video sharing in the YouTube social network[C] //Proc of the 34th International Conference on Distributed Computing Systems. IEEE Press, 2014: 298-307.

[6] Poulis G, Skiadopoulos S, Loukides G, et al. Distance-based kmanonymization trajectory data [C] //Proc of the 14th International Conference on Mobile Data Management. IEEE Press, 2013: 57-62.