International Journal of u- and e- Service, Science and Technology
Vol. 7, No. 6 (2014), pp. 61-68
http://dx.doi.org/10.14257/ijunesst.2014.7.6.06

Privacy-Preserving for Check-In Services in MSNS Based on Association Graph

Chen Wen

School of Mathematics and Computer Science, Tongling College, Tongling, P.R.China
tlxychenwen@163.com

Abstract

Check-in service as one of the most popular services in mobile social network services (MSNS), has serious personal privacy leakage threats. In this paper, check-in sequences of pseudonym users were buffered , and association graph for buffered check-in sequences was built, which can achieve a privacy guarantee of k-anonymity. The method guarantee the number of lost check-in locations is minimized while satisfying users' privacy requirements, it is also reduces the cost of finding trajectory k-anonymity set. At last, the results of a set of comparative experiments with (k,δ) - anonymity *on real world datasets show the method's accuracy and effectiveness.*

Keywords: *privacy preservation, location privacy, trajectory privacy, mobile social networks*

1. Introduction

Check-in service is the mainstream application of the mobile social network. Users use a mobile device with positioning function to send their geographical position to the mobile social network server, and select the semantic location corresponding to geographical position for check-in. However, privacy disclosure has become the primary obstacle to check-in service [1].Trajectories formed by the users' check-in records over a period of time can cause the exposure of the users' tracks [2]. The attackers can analyze the users' behavior patterns of that day, then speculate on the users' identity, and can even predict the users' future trajectory. Trajectory anonymization [3], by way of dummy location, reconstructs true tracks for privacy preserving. Data suppression [4-5] restricts the release of certain sensitive position on the trajectory or it restricts the trajectory segments that may lead to the disclosure of sensitive information. Generalization is the mainstream privacy preserving technology of trajectory with trajectory *k*-anonymity [6-8] as its representation. The privacy preserving technology of location based on location service and the privacy preserving technology of trajectory data have made some research results, but they can't be directly applied to mobile social networks. In recent years, researchers have made studies on privacy in mobile social networks, such as, location inference attack [9] in mobile social networks, location privacy preserving and absence privacy preserving [10] in mobile social networks, location privacy preserving of proximity based services [11-12] and so on. Making use of prefix trees, the literature [13] has achieved privacy protection of trajectory *k*-anonymity. This paper use the means of buffer check-in sequences proposed by literature [13], construct association graph to achieve *k*-anonymity privacy preserving. The experiment has proved that the method is more efficient.

ISSN: 2005-4246 IJUNESST

2. System Structure

This paper uses the central server system structure proposed by literature [13]. The central server system structure is composed of client, privacy preserving server and service provider. Presuming privacy preserving server and users are reliable, only pseudonym users in the system use privacy preserving service. Before using check-in service, users are required to send a registration request to privacy protection server. User registration module is in charge of user registration and storage of personalized privacy preserving parameters set by the users. These parameters mainly include: privacy preserving parameters k, the longest tolerance time Δt, a sensitive location privacy region. When ready to sign, the user first sends the "pre-sign" command to privacy preserving server. Then the pre-processing module judges the location submitted by the user is the exact geography location or semantic location. If it is the semantic location, the module will directly buffer the user's check-in location; if it is the geographical location, the module will make an anonymous inquiry to the service provider, gain access to point of interest nearby and then return it to the user, who can select the proper point of interest for another pre-sign. Preprocessing module is also in charge of deleting sensitive location privacy region of check-in sequences. According to the user check-in sequences, privacy preserving module constructs association graph, gains access to k-anonymity sequences and sign to the service provider.

3. The Generation Algorithm of k-anonymity Check-in Sequence

Definition 1, Check-in Sequence. Check-in records in a certain position of user u_i may be expressed as (L_i, t_i), where L_i indicates ID of registration location and t_i represents registration time. Check-in records constitute a check-in sequence of user u_i in chronological order.

Definition 2, k-anonymity check-in sequences. Given user u_i, predefined privacy parameters k and the longest tolerance time Δt, if and only if has other k-1 users which has the same check-in sequence with u_i within Δt time , the check-in sequence is named as k-anonymity check-in sequence of u_i.

The generation algorithm of k- anonymity check-in sequence comprises the following steps:

Step 1, Preprocessing of check-in sequence. Delete the location belonging to a predefined sensitive location privacy region in the users' check-in sequence;

Step 2, Sort the check-in sequence according to the check-in location ID , built association graph and finally get k- anonymity which could be signed .

Step 1 is implemented by preprocessing module, here not narrate. In step 2, the paper transforms k- anonymity problem into frequent itemsets mining problem based on association graph, delete the check-in position which is less than k and get k-anonymity check-in sequence. The following check-in sequence examples linked with Figure 1 (a) are used to introduce algorithm steps.

Step 1, Establish bit vector. Establish bit vector BV_{L_i} for each check-in record L_i. If check-in records L_i appears in the check-in sequence of the user u_j , the value of the bit vector of the j-th position is set to 1, otherwise is 0 . The bit vector of the check-in sequence shows in Figure 1 (a) is BV_{L_1} = {11111100000}, BV_{L_2} = {11100011110}, etc.

Step 2, Establish association graph. If the number "1" of $BV_{L_i} \wedge BV_{L_j}$ (i<j) is not less than the number of privacy protection parameters k, then record the values of i and j and then form a sequence which is denoted as S_2.Each item in the sequence S_2 constitute association graph nodes, and establish an edge between the two nodes. The association graph constructed according to the above method is expressed in Figure 1(b).

For example, from figure 1(b) we can obtain:

S_2={{1,2},{1,3},{1,4},{1,5},{2,7},{2,8},{3,4},{3,5},{4,5},{7,8}} .

Step 3, Expand the sequence S_n to generate a sequence S_{n+1}. Calculate the degree d(i) of association graph nodes. Investigate of the node's degree is not less than n, if every two node in n+1 nodes have connected edge and the number "1" of $BV_{L_{i1}} \wedge BV_{L_{i2}} \wedge \ldots \wedge BV_{L_{in}} \wedge BV_{L_{iu}}$ is not less than the privacy parameters K, then , {i_1, i_2, ..., i_n, i_u } is recorded into S_{n+1} sequence. Extend sequence successively until a new sequence could not be expanded.

For example, in connection with the association graph in Figure 1 (b) , analyze the specific process of sequence extension . Assume that privacy parameter k is 3, when n=2,d(1)=4, d(2)=3, d(3)=2, the degree of nodes are not less than n . $BV_{L_1} \wedge BV_{L_2} \wedge BV_{L_3}$ = {11100000000}, the number of "1" is not less than the number of privacy parameters k, thus, $\{1,2,3\} \subset S_3$. Extend each subsequences, the final sequence set are obtained as follows .

S_3={{1,2,3},{1,2,4},{1,3,4},{1,3,5},{1,4,5},{3,4,5},{2,7,8}}

S_4={{1,2,3,4},{1,3,4,5}}

S_5= Φ

Step 4, Restore the sequence in the S_n (n> = k) into check-in sequence S'_n. Compares check-in sequence L_i and S'_n, get the k- anonymity sequence which could be checked-in by the longest common subsequence.

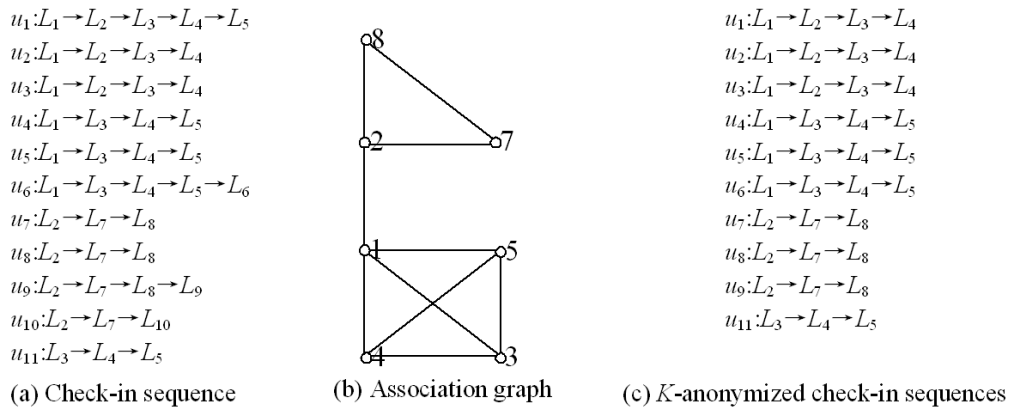
Definition 3, The longest common subsequence. Given two sequences S_1 and S_2, if there are sequence S_{sub} which meets $S_{sub} \subset S_1$ and $S_{sub} \subset S_2$, and there are not subsequence $S'_{sub} \subset S_{sub}$ meeting the above conditions, therefore, S_{sub} is called the longest common subsequence of S_1 and S_2 .

For example, k = 3, S_3,S_4 are reverted to check-in sequences :

S'_3={{ L_1,L_2,L_3},{L_1,L_2,L_4},{L_1,L_3,L_4},{L_1,L_3,L_5},{L_1,L_4,L_5},{L_3,L_4,L_5},{L_2,L_7,L_8}}

S'_4 = {{L_1, L_2, L_3, L_4}, { L_1, L_3, L_4, L_5}}

Traverse the check-in sequence in Figure 1, select S'_n having the same length subsequence according to the check-in sequence length of u_i to make a compares and get the longest common subsequence. For instance, to get the longest common subsequence between the check-in sequence of u_6 and S'_4 , the result is u_6: $L_1 \rightarrow L_3 \rightarrow L_4 \rightarrow L_5$. u_7's check-in sequence length is 3 , to get the longest common subsequence with S'_3 , the result is u_7: $L_2 \rightarrow L_7 \rightarrow L_8$. Ultimately, the sign obtained on the k- anonymous sequence is shown in Figure 1 (c).



(a) Check-in sequence (b) Association graph (c) K-anonymized check-in sequences

Figure 1 Check-in sequences, association graph and k- anonymized check-in sequences

The following description is given based on association graph for k- anonymity check-in sequence algorithm.

Algorithm： *k*- anonymized check-in sequences generation algorithm
Input: check-in sequences L_i, privacy preserving parameters *k*
Output: *k*-anonymized check-in sequences
for(j=1;j<=*M*;j++)
/* *M* is equal to the number of users */
for all item *i* in L_j do
set the *j*th bit of BV_{L_j} to 1;
for all *i* in L_i do
for every two items *i*,*j*(*i*<*j*) do
if (the number of 1 in $BV_{L_i} \wedge BV_{L_j}$)>=*k*
i,*j* constitute association graph nodes and establish an edge between the two nodes;
i.deg++;*j*.deg++;
/* the degree of node *i*,*j* increased 1*/
$S_2 = S_2 \cup \{i,j\};$}
/* generate S_{n+1} sequence */
n=2;
for all sequences $\{i_1, i_2, ..., i_n\} \subset S_n$ do
{
n=*n*+1;
if((the number of 1 in $BV_{L_{i1}} \wedge BV_{L_{i2}} \wedge \ldots \wedge BV_{L_{in}} \wedge BV_{L_{iu}}$)>=*k*)&&(every two items in *n*+1 items have connected edge))
$S_{n+1} = S_{n+1} \cup \{i_1, i_2, ..., i_n, i_u\}$
}
if(*n*>=*k*)
transform S_n to S'_n
for all L_i do
L'_i=LCS(L_i,$S'_{|Li|}$)
/* LCS is the longest common subsequence function, $|L_i|$is the length of check-in sequences, L'_i is the *k*-anonymity check-in sequence*/

4. Experimental Results and Analysis

4.1 Experiment Data

The algorithm of this article uses Java, running on E5800 3.2GHz processor and Windows XP platform with 2G memory which also use true check-in data on Brightkite of [14]. The datasets collected data of 24 months and the data set properties were shown in Table 1. We Use *k*- anonymity algorithm [6] as a comparison algorithm.

Table 1. Dataset Properties

check-in number of locations	*total number of users*	*regional area /km²*	*user density*
541169	9435	443556	0.02
the average number of check-in	POI average number of check-in	average check-in interval /h	average check-in distance interval /km

57.36	6.10	56.81	15.39

The parameters of experimental include privacy protection parameters *k*, the check-in sequence length *len* and the maximum tolerable time Δt. Check-in success rate C_S reflects the proportion of the users' original check-in location contained in the *k*- anonymity check-in sequence and use formula (1) to measure.

$$c_s = \frac{|kChS_iL| - \|kChS_iL| - |ChS_iL\|}{|kChS_iL|} \qquad (1)$$

Wherein $|kChS_iL|$ signify the number of check-in locations of *k*- anonymity check-in sequence; $|ChS_iL|$ indicates number of check-in locations of the original check-in sequence.

Experiment randomly selected the check-in sequence in the data set from 4,000 users and don't consider the changes in the maximum tolerance time while testing the influence on check-in success rate from privacy protection parameters *k* and the sequence length *len* . This algorithm is denoted by AGC (Association Graph Check-in privacy preserving algorithm). Figure 2 shows the check-in success rate of the algorithm.

4.2. Result Analysis

4.2.1. Privacy Protection Parameters and Check-in Success Rate: Figure 2 shows an impact on check-in success rate from privacy parameters *k*. The values of privacy protection parameters *k* increases from 5 to 12. As can be seen from Figure 2(a), the check-in success rate decreases with the increase of the privacy parameters *k*. The increases in *k* cause more losses of check-in location, which results in the decrease of check-in success rate. Meanwhile, increases in check-in sequence due to failure of anonymity of (k,δ)-anonymity algorithm also leads to a decline in check-in success rate. From the comparison experiment, we can get the idea that the success rate of the algorithm in this article is much higher.

4.2.2. Length of Check-in Sequence and Check-in Success Rate: The value of sequence length of location fluctuates is between 5and 30; it increases by 5 each time. The value of location sequence length relates to the dataset properties. Selecting check-in sequence randomly from the 4,000 users, the experiment intercepts and complements the value at random that satisfies the length requirement. From figure 2(b), when *k* equals 10, the increase of check-in sequence length leads to the decrease of check-in success rate of algorithm. The reason is that the longer the location sequence is, the less the identical location sequence will be, which may result in more loss of location. Due to the limited impact of increase of sequence length on trajectory distance and clustering results, the growth of sequence length barely affects the check-in success rate of (k,δ)-anonymity.

4.2.3. The Longest Tolerance Time and Check-in Success Rate: Figure 2(c) indicates the impact of the longest tolerance time of check-in success rate. The longest tolerance time is the time of cache check-in sequence. In the analysis of data set attributes, the paper gets the time interval of check-in data set through the experiment as table 1 shows. The value of longest tolerance Δt is the 1-5 times of average check-in time. In the experiment, the value of k is 10, doesn't have specific restrictions on check-in sequence length. In figure 2(c), the increase of check-in success rate of algorithm is accompanied by the growth of longest tolerance time. For the more the cache of check-in sequence is, the easier the *k*-anonymous check-in sequence generates; while the less the check-in sequence is, the more loss of check-in location the *k*-anonymous check-in sequence

generates. So does the method of (k,δ)-anonymity: the more the cache of location sequence is, the more possibility of search of *k*- anonymous set through cluster.

According to the experiments, the method introduced in the paper significantly outperforms the method of (k,δ)-anonymity in success rate of check-in.

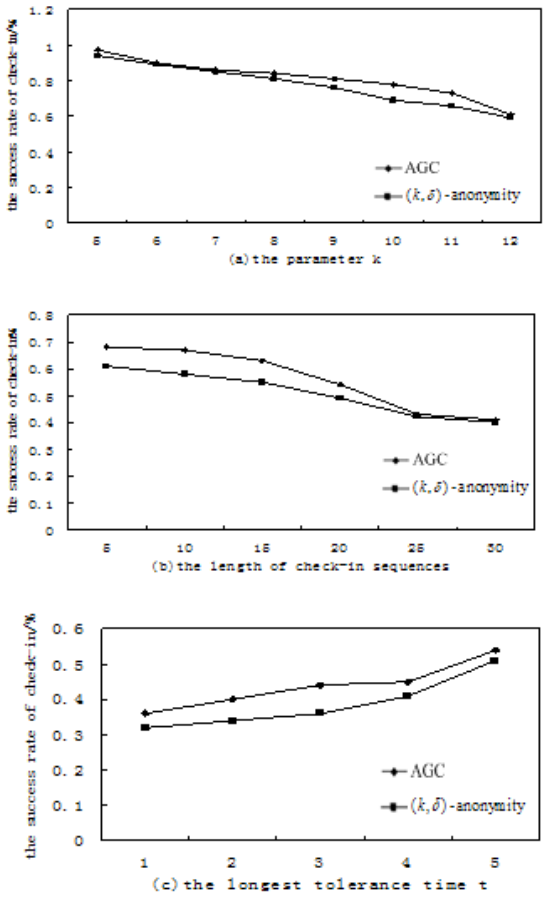


Figure 2. Experimental Results

5. Conclusions

As to the problem of privacy disclosure of the location and the trajectory of pseudonym users in mobile social network, the paper presents a means of privacy protection based on the association graph which ensures both the trajectory privacy of the users and a higher success probability of check-in. Through transformation of the trajectory *k*- anonymity into the *k*-frequent item sets mining based on the association graph, the method deletes the check-in location whose support degree is less than *k* and gets the check-in sequence of *k*- anonymity. The method not only simplifies the searching process of trajectory *k*- anonymity set and saves storage by association graph, but also reduces the calculation and improves the privacy protection.

Acknowledgements

This work was supported by funds from Universities Key Fund of Anhui Province for Young Talents of China under Grant 2013SQRL082ZD and Natural Science Research Universities Key Project of Anhui Province of China under Grant KJ2014A256.

References

[1] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking", In *Proceedings of the 1st international conference on Mobile systems, applications and services*, ACM, (**2003**) May, pp. 31-42).
[2] H. Z. M. Xiao-Feng, "A survey of trajectory privacy-preserving techniques", *Chinese Journal of Computers*, vol. 10, (**2011**), p. 008.
[3] Z. Huo, X. Meng, H. Hu and Y. Huang, "You can walk alone: trajectory privacy-preserving through significant stays protection", In *Database Systems for Advanced Applications*, (**2012**) January, pp. 351-366. Springer Berlin Heidelberg.
[4] T. H. You, W. C. Peng and W. C. Lee, "Protecting moving trajectories with dummies. In *Mobile Data Management, 2007 International Conference on*, IEEE, (**2007**) May, pp. 278-282.
[5] M. Terrovitis and N. Mamoulis, "Privacy preservation in the publication of trajectories", In *Mobile Data Management, 2008. MDM'08. 9th International Conference on*, IEEE, (**2008**) April, pp. 65-72.
[6] O. Abul, F. Bonchi and M. Nanni, "Never walk alone: Uncertainty for anonymity in moving objects databases", In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, IEEE, (**2008**) April. pp. 376-385.
[7] M. E. Nergiz, M. Atzori and Y. Saygin, "Towards trajectory anonymization: a generalization-based approach", In *Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*, ACM, (**2008)** November, pp. 52-61.
[8] R. Yarovoy, F. Bonchi, L. V. Lakshmanan and W. H. Wang, "Anonymizing moving objects: how to hide a MOB in a crowd?", In*Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, ACM, (**2009**) March, pp. 72-83.
[9] A. Sadilek, H. Kautz and J. P. Bigham, "Finding your friends and following them to where you are", In *Proceedings of the fifth ACM international conference on Web search and data mining*, ACM, (**2012**) February, pp. 723-732.
[10] D. Freni, C. Ruiz Vicente, S. Mascetti, C. Bettini and C. S. Jensen, "Preserving location and absence privacy in geo-social networks", In*Proceedings of the 19th ACM international conference on Information and knowledge management*, ACM, (**2010**) October, pp. 309-318.
[11] S. Mascetti, D. Freni, C. Bettini, X. S. Wang and S. Jajodia, "Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies", *The VLDB Journal—The International Journal on Very Large Data Bases*, vol. *20*, *no.* 4, (**2011**), pp. 541-566.
[12] S. Mascetti, C. Bettini, D. Freni, X. S. Wang and S. Jajodia, "Privacy-aware proximity based services", In *Mobile Data Management: Systems, Services and Middleware, 2009. MDM'09. Tenth International Conference on*, IEEE, (**2009**) May, pp. 31-40.
[13] Z. Huo, X. F. Meng and Y. Huang, "PrivateCheckIn: Trajectory privacy-preserving for check-in services in MSNS", *Jisuanji Xuebao(Chinese Journal of Computers)*, vol. 36, no. 4, (**2013**), pp.716-726.
[14] E. Cho, S. A. Myers and J. Leskovec, "Friendship and mobility: user movement in location-based social networks", In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, (**2011**), August, pp. 1082-1090.

Author

Chen Wen, he is an Associate Professor in the School of Mathematics and Computer Science, Tongling College, Tongling, P. R. China. He holds a master degree in Computer Science and Technology from the Anhui University, Anhui, P. R. China. His previous research areas include privacy preserving.