ISBN 978-981-11-3671-9
Proceedings of 2017 the 7th International Workshop on Computer Science and Engineering (WCSE 2017)
Beijing, 25-27 June, 2017, pp. 460-464
doi: 10.18178/wcse.2017.06.079

A Fuzzy Query Based Location Privacy Protection

Yifu Li [1], Shan Yao [1], Xiangzhan Yu [2] and Bo Sun [1+]

[1] National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing ,China

[2] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

Abstract. In this paper, we put forward a fuzzy query method—FCM, to provide relatively reliable inquiry service based on fuzzy set. Using this method, we can realize the balance between location privacy protection and the quality of service.

Keywords: fuzzy set; location privacy protection; quality of service

1. Introduction

In recent years, location-based service which is short for LBS[1] is developing rapidly, making the object location search of the internet of things fast and extensible. When using the service of LBS, users need to provide their own location information to the LBS servers. But these servers are usually not fully reliable, so as to leak the users’ location information collected to the attackers. Therefore, location-based service makes a great challenge for users’ location privacy protection.

LBS has three main goals: "where are you", "who are you with" and "who and what is nearby". This paper mainly solves the problem of "who and what is nearby". First, we should know “who” and “what”. “Who” stands for the location of object, and “what” stands for the request of services.

In this paper, we propose a fuzzy query based location privacy protection. This method protects the users’ location privacy in a fuzzing way and by realizing balance between location privacy protection and the quality of location service in some degree, this method also provides relatively reliable QoS.

2. The Structure of Location Privacy Protection

A traditional LBS server provides the users with "who and what is nearby" and makes up of three parts: Client Manager, System Service and Positioning Function Service, as shown in Fig. 1.

In order to protect the location privacy of the user, some sort of firewall is usually placed between the user end and the LBS server[2]. In this paper we add a trusted fuzzy server, as displayed in Fig. 2.

The Structure of Location Privacy Protection with the fuzzy server of Fig. 2 works as follows: Firstly it collects the location request information sent by the users and stores the information; Secondly it fuzzes the real location information of the request information, and then uses the fuzzed set instead so as to protect theusers’ real location information. In the end it receives the results set from the LBS server, and then filters the results set to find the relatively reasonable results according to the need of the corresponding users.

3. The Construction of Fuzzy Sets Based on K-anonymity

According to the structure of location privacy protection, the real location information of users has to be fuzzed before it is sent to the LBS server. In this paper we construct the fuzzy set based on k-anonymity

[+] Corresponding author. Tel.: +86-13811810387
E-mail address: liyifu_1981@163.com

method[3]. By summarizing and hiding[4], k-anonymity makes every record has the same identification with the other k-1 records in the data table, so as to reduce privacy disclosure. K-anonymity ensures the sensitive attributes of every unit to hide in the group whose scale is k. In this way, the chance of identifying each single unit remains no more than 1/k. Now K-anonymity technology mainly has two methods: generalization [5] and hiding technology. From all k-anonymity realities, this paper chooses Datafly algorithm [6] to calculate the appearing probability of every attribute, and to generalize the attributes which do not meet the requirements until this attribute meets the requirement of k-anonymity afterwards.

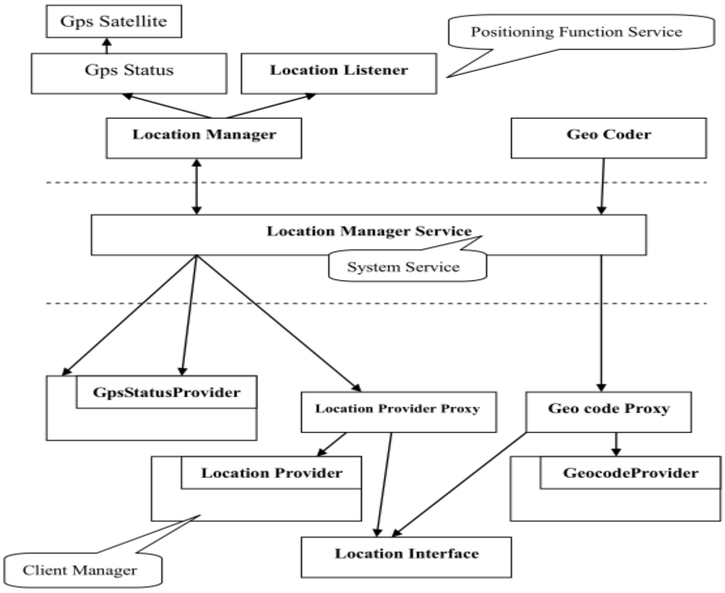

Fig. 1: The structure of the traditional location privacy protection

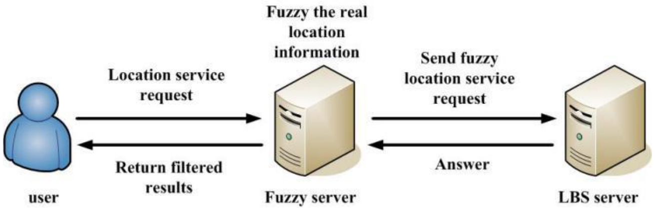

Fig. 2: The structure of the fuzzing based location privacy protection

4. A Fuzzy Query Based Location Privacy Protection

After constructing the fuzzy set by using k-anonymity algorithm, LBS server needs to provide the query service of "what is nearby " for the whole fuzzy set. However, the key problem is how to provide the best service based on the fuzzy set. According to this problem, this paper offers a fuzzy query method which is called FCM method. This method can be added into the LBS server and by calculating the shortest paths between the fuzzy set and the near request points, it provides relatively reliable query service. Therefore, we can protect the real location of object, and provide the relatively good service at the same time.

In order to calculate the length of the shortest path between the fuzzy set and the near request points, we realize an improved Dijkstra algorithm[7] in FCM method. This is because Dijkstra algorithm expands outward one layer after another from the starting point and needs to make all the points ergonomic, so that its complexity is not acceptable when processing large number of points

4.1. FCM Computing Method.

4.1.1 The Partition of the fuzzy set

The goal of the FCM method is to choose the appropriate points to response the users, but the shortest path to the nearby service points differs for the points of the fuzzy set. To solve this problem, we divide the fuzzy set into different parts and process them separately. This method also guarantee the quality of service

4.1.2 FCM Computing method

To facilitate the research, we set a virtual point s, and connect s to the points of , the value of the weights of their edges are 0.0. As shown in Figure 3.

In the actual calculation, we can first calculate the length of the shortest path between the virtual point s and the points of the fuzzy set , and then make the shortest path negative, so after the reverse order, the point before s must be in the request set .

The description of FCM method is shown as follows:

Algorithm input: picture G = (V, E) which has edge weights, fuzzy set, physical location ,request set .

Algorithm output:

(1),,when min(o1) = min(o2)；

(2) stands for the possibility of q to be the appropriate request point.

1. Construct picture， , 。
2. Assume there is a road between any two nodes, even if not direct, another way can arrive.
3. Start from the s point, use greedy strategy to choose a road to, record the length as boundary called bound.
4. Return to the root node s, construct the solution space tree, define table PT to put the live nodes,the solution vector is X=(x1,…,xn), the range of xi is Si,|Si|=ri, ri stands for child nodes.
5. For every child note x, calculate the length of path between itself and the root node.
6. If one childe node's weight of the length of path is more than bound, give it up, else put this child node into the table PT.
7. Continue to expand, repeat step 7 untile is the leaf node. Check its length, if less than bound,update the value of bound.
8. Take the first node of the table PT as the expanding root node, repeat step 5, until getting a optimal solution X=(x1,…,xn),record the length of shortest path as the final bound.
9. Repeat the above steps to get the shortest path between point s and the other points of .
10. Define second: ， second(o)→ v_q， v_q is the second node in the path between s and o, this point must be in the request set .
11. ,when second(o_1) = second(o_2), now the partition is constructed completely.
12. For every child note x, calculate the length of path between itself and the root node.
13. If one childe node's weight of the length of path is more than bound, give it up, else put this child node into the table PT.
14. Continue to expand, repeat step 7 untile is the leaf node. Check its length, if less than bound,update the value of bound.
15. Take the first node of the table PT as the expanding root node, repeat step 5, until getting a optimal solution X=(x1,…,xn),record the length of shortest path as the final bound.
16. Repeat the above steps to get the shortest path between point s and the other points of .
17. Define second: ,， second(o)→ vq， vq is the second node in the path between s and o, this point must be in the request set .
18. ,when second(o1) = second(o2),now the partition is constructed completely.

In FCM method, step 3-10 calculates the shortest path between s and points of , step 11-12 divides the fuzzy set based on, step 13-19 chooses the appropriate points of to return. If the partition has only one, that must be (line 13), showing,, any point of can take the place of physical location, return (line 14). When there are many partitions,if the user agrees to identify the class of his real location, return (line 17). Else return any point in the class of the biggest area to take the place of physical location,as (line 19). In line 19, stands for the possibility of q to be the appropriate request point.As is constant for a fuzzy set, we just need to compare to determine the size of. stands for the number of points corresponding to q in the partition.

5. Simulation Experiment and Result Analysis

FCM method provides service as reliable as possible under the condition of uncertainty, and uses the shortest path as the standard measurement.

We assume the picture G = (V, E) has 68 points and 580 edges, we set the fuzzy testing set as follows:

O_1 = {14, 21, 22, 23, 24, 27, 29, 30, 32, 35}

O_2 = {50, 52, 54, 56 ,57, 58, 59, 63}

O_3 = {8, 10, 11 ,15 ,18 ,19 ,24 ,30 ,32 ,33 ,34 ,36 ,37 ,43 ,47 ,53 ,55 ,62 ,65, 68}

The request are set as follows:

Q = {67, 38, 24, 65, 35, 62, 31, 27, 14, 11, 5, 59},,

Points of O_1 and O_2 are relatively concentrated, while points of O_3 are random discrete points. After adding these virtual points into the picture G, there are 69 points and 592 edges, since there are 12 points in Q. On the basis of the matrix of the edges' weights, we calculate the shortest length between s and the points of O_1 ,O_2 and O_3 respectively. The results are shown in table 1, table 2 and table 3.

Table 1: Test results of O_1

Starting point	Ending point	Shortest length	Negative sequence	The second inverse point
69	14	1705	14 -> ...-> 27 -> 69	27
69	21	719	21 -> ...->27 -> 69	27
69	22	536	22 -> ...->27 -> 69	27
69	23	1436	23 -> ...->14-> 69	14
69	24	0	24 -> 69	24
69	27	0	27 -> 69	27
69	29	572	29 -> ...->27 -> 69	27
69	30	305	30 -> ...->27 -> 69	27
69	32	501	32 -> ...->27 -> 69	27
69	35	909	35 -> ...->27 -> 69	27

Table 2: Test results of O_2

Starting point	Ending point	Shortest length	Negative sequence	The second inverse point
69	50	1506	50 ->...-> 59 -> 69	59
69	52	1635	52-> ...->59 -> 69	59
69	54	1765	54-> ...->59 -> 69	59
69	56	1579	56 ->...-> 65-> 69	65
69	57	1661	57 -> ...->65 -> 69	65
69	58	1581	58 -> ...->67-> 69	67
69	59	372	59 ->...-> 65 -> 69	65
69	63	1243	63 -> ...->65 -> 69	65

Table 3: Test results of O_3

Starting point	Ending point	Shortest length	Negative sequence	The second inverse point
69	8	2569	8 -> ...->5 -> 69	5
69	10	1556	10 -> ...->11 -> 69	11
69	11	0	11...-> -> 69	11
69	15	515	15 ->...-> 14 -> 69	14
69	18	0	14...->-> 69	14
69	19	1435	19 ->...-> 25 ->69	25
69	24	1436	24 ->...-> 14-> 69	14
69	30	305	30 -> ...->27 -> 69	27
69	32	501	32 ->...-> 27 -> 69	27

69	33	1107	33 ->...-> 27 -> 69	27
69	34	576	34 ->...-> 27 ->69	27
69	36	495	36 -> ...->31 -> 69	31
69	37	909	37 ->...-> 27 ->69	27
69	43	1453	43 -> 35 -> 69	35
69	47	2072	47 -> 57 -> 59 -> 69	57
69	53	1319	53 -> 59 -> 69	59
69	55	1114	55 -> 59 -> 69	59
69	62	672	67 -> 62 -> 69	62
69	65	1243	69 -> 62 -> 69	62
69	68	814	73 -> 72 -> 69	67

Table 1 shows that since 80% evaluation results points to point number 27, the fuzzy server shall return point number 27 to the user. Table 2 shows that there are 50% and 37.5% evaluation results point to point number 65 and 59 respectively, and the fuzzy server shall returns these two points to the user. Table 3 shows that although the number of points is larger, and the points are more random, point number 27 can also be selected and returned to the user. This result indicates that even though the number of points is larger, and so is the ratio of the random points, the FCM method can still keep a certain effectiveness. In summary, the proposed method can provide relatively reliable query service under condition of uncertainty.

6. Conclusion

This paper has realized to provide relatively reliable service to users under the condition of uncertainty based on the structure of location privacy protection.On the basis of fuzzy set ,we use the shortest path as a standard measurement to solve the query problem "who and what is nearby". Who stands for the fuzzy set. This paper puts forward a fuzzy query method based on location privacy protection—FCM, after protecting the users' location information, we still can get relatively reliable service.This method realizes the balance between location privacy protection and the quality of service.In the end ,after testing,we prove that FCM method has practical value in some degree.

7. References

[1] Kevin Ashton. C. 2009. That ‘Internet of Things’ ThingRFID Journal.

[2] Xiao Z, Meng X, Xu J. C. 2007. Quality-aware privacy protection for location-based services. *Proceedings of the International Conference on Database Systems for Advanced Applications.*Bangkok.Thailand.113-120.

[3] Samarati P, Sweeney L. J.2002. Protecting privacy when disclosing information：k-anonymity and its enforcement through generalization and suppression.*International Journal on Uncertainty*. Fuzziness and Knowledge-based Systems. 10(5),557-570.

[4] Machanavajjhala A,Gehrke J, Kifer D. C.2006. l-diversity: Privacy beyond K-Anonymity. *Proceedings of the 22nd International Conference on Data Engineering Atlanta.* GA.USA:ACM Press. 3-8.

[5] HundepoolAnco, Willenborg Leon. C.2003. u-argus and t-argus：software for statistical disclosure control. *Proceedings of the Third International Seminar*. Statistical Confidentiality. 156-163.

[6] Sweeney L. D. 2001. Computational disclosure control: a primer on data privacy protection. *PhD dissertation: Computer Science Dept.* Massachusetts Institute of Technology.

[7] ZhangYongLong.J.2005.Dijkstra shortest path algorithm optimization. *Journal of nanchang institute*. 1(2):23-25.