Towards Preserving Privacy in Participatory Sensing

Leyla Kazemi
Information Laboratory, Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
lkazemi@usc.edu

Cyrus Shahabi
Information Laboratory, Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
shahabi@usc.edu

***Abstract*—With the abundance and ubiquity of mobile devices, a new class of applications is emerging, called participatory sensing (PS), where people can contribute data (e.g., images, video) collected by their mobile devices to central data servers. However, privacy concerns are becoming a major impediment in the success of many participatory sensing systems. While several privacy preserving techniques exist in the context of conventional location-based services, they are not directly applicable to the PS systems because of the extra information that the PS systems can collect from their participants. In this paper, we formally define the problem of privacy in PS systems and identify its unique challenges assuming an un-trusted central data server model. We propose PiRi, a privacy-aware framework for PS systems, which enables participation of the users without compromising their privacy.**

***Keywords*-participatory sensing; privacy;**

I. Introduction

With the advent of mobile technology, the area of participatory sensing (PS) has attracted many researchers in different domains such as public health, urban planning, and traffic. The goal is to leverage sensor equipped mobile devices to collect and share data, which can later be utilized for analysis, mining, prediction or any other type of data processing. While many *unsolicited* PS systems exist (e.g., Flickr, Youtube), in which users participate by arbitrarily collecting data, other PS systems are *campaign*-based, which require a *coordinated* effort of the participants to collect a particular set of data that the server requires for any purpose. Some real-world examples of PS campaigns include [1], [2], where users leverage their mobile devices to collect traffic information.

However, privacy concerns are the significant barriers to the success of any participatory sensing campaign, which delay the progress of massive deployment of such systems. Consider a scenario where the goal of the PS campaign is to collect pictures/videos from the anti-government riots at different locations of a city with the coordinated effort of the participants. Accordingly, each participant u should query the server for the set of closeby locations from which data needs to be collected (termed data collection points or *DC-points*). These are the DC-points that are closer to u than to any other participant. However, u may not be willing to disclose his identity due to safety reasons. An alternative is that u sends his query to a trusted server, known as *anonymizer*. The anonymizer removes the user's ID from the query and forwards the query to the server. However, the server requires u's location information to answer the query. Due to the strong correlation between people and their movements (see [3]), a malicious server can identify u by associating his location information to u. Thus, the server can identify a query issuer by associating the query to the location from which the query is issued. We refer to this process as a *location-based* attack. Our goal in this paper is to protect the campaign participants from location-based attacks by disassociating a query from the query location.

Existing privacy preserving techniques have been proposed to address these concerns in the context of location-based services (LBS) [4], [5]. Unfortunately, certain characteristics of a PS campaign distinguish it from conventional LBS, and therefore, prevent a direct adaption of LBS approaches to such systems. One characteristic of a PS campaign is that in order to collect data through a coordinated effort, *all* the participants query the PS server for the closeby DC-points. This is in contrast to LBS which serves millions of users from which any arbitrary subset of them might ask query at a given time and location. We refer to this as the *all-inclusivity* property. Another characteristic of a PS campaign, is that each participant queries for all the DC-points, which are closer to him than to any other participant. Thus, the second property of the PS campaign is that each participant asks a range query from the server which is dependent on the location of other users. We refer to this property as *range dependency*. These two properties, which reveal extra information to the server as compared to the conventional LBS, introduce major privacy leaks to the system. Thus, the system becomes unresilient to location-based attacks.

In this paper, we devise a privacy-aware framework for PS campaigns, which addresses these two major privacy leaks. Our approach, termed **PiRi** has the two following properties: **P**artial-**i**nclusivity and **R**ange **i**ndependence. PiRi is based on the observation that the range queries sent by participants have significant overlaps. Therefore, instead of each participant asking a separate query, only a group of the representative participants ask queries from the server, and share their results with those who have not posed any query. Moreover, instead of each participant submitting a range query, which is dependent on other participants' locations, we propose an adjustment technique that adjusts the range query such that the query becomes independent of the others. To the best of our knowledge, this paper is the first attempt in introducing a privacy framework for PS campaigns during the coordination phase.

The remainder of this paper is organized as follows. Section II reviews the related work. In Section III, we formally define our problem, and discuss our system model. Thereafter, in Section IV we explain our PiRi approach. Finally, in Section V we conclude and discuss the future directions of this study.

II. Related Work

Privacy preserving techniques have been studied in the context of location-based services. One category of well-known techniques is the spatial *K*-anonymity (SKA) [6], [5], [4], where the user's location is blurred in a cloaked area that contains at least K-1 other users.

Most of the SKA techniques assume a *centralized* architecture [4], which utilizes a trusted third party known as *location anonymizer*. The anonymizer is responsible for first cloaking user's location in an area before contacting the location-based server. The centralized approach have two drawbacks. First, it does not scale because the users should repeatedly report their location to the anonymizer. Second, by storing all the users' locations, the anonymizer becomes a single point for attacks. To address these shortcomings, recent techniques [6] focus on distributed environments, where the users employ some complex data structures to anonymize their location among themselves via fixed infrastructures (e.g., base stations). However, because of high update cost, these approaches are not designed for the cases where users frequently move or join/leave the system. Therefore, alternative approaches have been proposed [5] for unstructured peer-to-peer networks where users cloak their location in a region by communicating with their neighboring peers without requiring a shared data structure. In this paper, we employ the P2P spatial cloaking techniques to hide the user's location when querying the PS server.

Despite all the studies about privacy in the context of LBS, only a few work [7], [8] have studied privacy in participatory sensing. However, their focus is on the data contribution, rather than the coordination phase. That is, these approaches deal with how participants upload the collected data to the server without revealing their identity, whereas our focus is on how to privately assign a set of data collection points to each participant.

III. Preliminaries

A. Formal Problem Definition

A major focus in the PS campaign is to design a framework in which each participant is assigned to a set of data collection points (DC-points), where data should be collected. In this section, we formally define this problem.

Definition 1 (Participatory Assignment): Given a campaign $C(P, U) \in R^2$, with P as the set of DC-points, and U as the set of participants, the *Participatory Assignment (PA)* problem is to assign to each participant $u \in U$ any DC-point $p \in P$, such that p is closer to u than to any other participant in U.

Note that for simplification, we do not assume the participants move during the assignment. Moreover, participants are the current active users of the system willing to participate in the process.

In order to solve the PA problem, a straightforward solution is that each participant sends his location to the server. The server then assigns to each participant the set of DC-points close to him by computing the *Voronoi diagram* of the participants, which is a partitioning of environment into a set of cells, where each cell V_u belongs to a participant u, and any point in the cell V_u is closer to u than to any other participants in the environment. Figure 1 depicts such scenario.

Once the server computes the Voronoi diagram of the participants, it forwards to each participant u, all the DC-points lying inside the corresponding cell V_u. However, in many scenarios the server is not trusted, and therefore, a participant may not be willing to reveal his identity to the server. Even if the participant hides his identity from the server (i.e., only reveals his location), due to the strong correlation between people and their movements ([3]), a participant can still be identified by his location. In the following, we first formally define our privacy attack. Thereafter, we define the privacy problem.

Definition 2 (Location-based attack): A *location-based attack* is to identify a query issuer by associating the query to the query location (i.e., location from which the query is issued).

Definition 3 (Problem Definition): The *Privacy-Aware Participatory Assignment (PAPA)* problem is a variation of the PA problem (Definition 1), in which the goal is to protect participants' identity from location-based attacks.

B. System Model

In this section, we first describe our privacy threat model, and then discuss our system architecture which consists of two entities, participants and the PS server.

Our assumption is that participants trust each other, and do not reveal any sensitive information about their peers. However, they trust neither non-participant nor the PS server. We refer to any such entity as *adversary*. Moreover, the adversary, if needed, can obtain the locations of all participants [9]. The reason is that participants often issue their queries from the same locations (office, home), which can be identified through physical observation, triangulation, etc. Moreover, each user must register with the server, receive the campaign password, and become the campaign participant before communicating with other campaign participants. Finally, in order to guarantee the pseudonymity of participants' location information, each query is assigned with a unique pseudonymous identity, which is totally unrelated to the participants's personal identity.

Our PS server which contains the list of DC-points, is equipped with a privacy-aware query processor, which processes the queries issued by the participants. Each participant can determine his privacy level, by specifying two parameters: *K*, and *A*. *K* determines the *K*-anonymity, and *A* specifies the minimum resolution of the cloaked region. Each participant is equipped with two wireless network interface cards. One is dedicated to the communication with the PS server through a base station or wireless modem. The other one is dedicated to the P2P communication among the peers through a wireless LAN, e.g., Bluetooth or IEEE 802.11. Also, each participant is equipped with a positioning device, e.g., GPS, which can determine its current location.

IV. PiRi Approach

To solve the PAPA problem, participants cannot share their locations with the untrustworthy server for the assignment of DC-points. Therefore, the centralized solution to the PA problem is no longer a viable solution. Thus, one baseline solution is that participants communicate among their peers to compute their Voronoi cell. Thereafter, each participant performs a privacy-aware range query [5] to retrieve all the DC-points inside his Voronoi cell.

However, this baseline approach has major privacy leaks, which originates from the two characteristics of a PS campaign: all-inclusivity and range dependency. These properties leak enough information to the server with which the server can easily identify each participant by linking his query to the query location. This gets even easier, if the server knows the exact locations of all the participants. The reason is that on one hand the server receives a set of query regions, and on the other hand, the server has the query locations. Each query region overlaps with a set of participants, one of which have issued the query. Therefore, the server can associate the query to its location by solving a matching problem between these two sets of data. As a result, the more information the server has, the more correct matches it can find between the queries and query locations. Hence, the baseline approach is not appropriate for our PAPA problem.

Our goal is to overcome the drawbacks of the baseline approach by preventing these privacy leaks. Our PiRi approach has two major steps, *Query Formation* and *Query Selection*, which are discussed in the following.

A. Query Formation

To solve the PAPA problem, a set of DC-points those inside his Voronoi cell, should be assigned to each participant. This indicates that each participant should first compute his Voronoi cell in a distributed way [10] to form the spatial range query. Thereafter, by employing the P2P SKA technique [5], the participant forms a privacy-aware range query. However, the problem is that the range query is dependent on the size of the participant's Voronoi cell (range dependency), which is a potential for information leak. The reason is that the participant U_i must send its cloaked region along with the radius r_i (i.e., the radius of the smallest enclosing circle of his Voronoi cell) to the server to retrieve those DC-points that are inside his Voronoi cell. However, each of the *K* participants in the cloaked region, termed *local peer*, has a different Voronoi size, and therefore, a different r. Consider an extreme case where the server knows the locations of the participants and hence it can compute their Voronoi cells, and the radius r for each of them. Consequently, the server can easily identify the query issuer (i.e., the set of all participants in the cloaked region with radius r). Figure 2 depicts such scenario, where U_1 (black dot) cloaks himself with U_2, and sends the cloaked region along with radius r_1 to the server (see the size of r_1 as compared to r_2). The server, knowing the location of the participants, and hence their Voronoi cells (i.e., r_1, and r_2), matches the query with radius r_1 to its query location (i.e., the location of a participant with the Voronoi cell of the same radius).

One approach to avoid the range dependency leak is that each participant U_i not only cloaks his location among *K*-1 other peers but also cloaks his range query among those of the other *K*-1 peers. For example, instead of forming his range query with radius r_i, the participant can form his query with radius r_{max}, where r_{max} is the maximum radius among all the *K* peers inside the cloaked region. This guarantees the *K*-anonymity at all times.

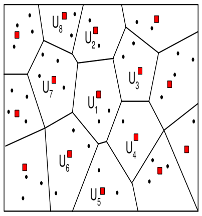

Figure 1. Illustrating the assignment of DC-points to the participants

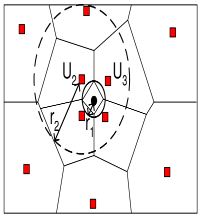

Figure 2. Illustrating an example of Range dependency

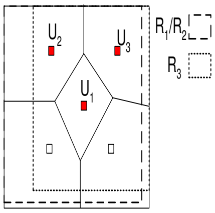

Figure 3. Illustrating an example of all-inclusivity leak

B. Query Selection

Once all participants formed their query regions, they can send them out to the server. Since the server is receiving queries from all participants, it can utilize the gathered information (i.e, query regions) from all participants to form an attack (all-inclusivity leak). Figure 3 illustrates such scenario. For simplicity, we assume that only users $U_{1..3}$ participate in the campaign. The figure shows that U_1 cloaks himself with U_2. Similarly, U_2 forms a cloaked region with

U_1. Consequently, both U_1 and U_2 form identical query regions. The figure also depicts that U_3 cloaks himself with U_1. Accordingly, the server can easily identify U_3 by relating it to the query region R_3, since U_3 appears only once (i.e., R_3) in all the three submitted query regions to the server. This indicates the more participants submit queries to the server, the more information server has to infer the participants' identities. Our algorithm attempts to prevent this leak by minimizing the number of queries submitted to the server, while assigning the nearby DC-points to *every* participant.

In order to address this issue, we observe that there is a large overlap among the query regions of the participants. Therefore, by receiving the result from the server, one can share his result with all his peers whose Voronoi cells lay completely inside his query region. The question is how to select the group of representative participants. To answer this question, we should solve the following optimization problem.

Definition 4 (V-Cover): Given a campaign $C(P, U) \in R^2$, with P as the set of DC-points, and U as the set of participants, let R and V be the set of query regions and Voronoi cells for the set U, respectively, where R_i corresponds to the query region for user U_i, and V_i is the Voronoi cell for U_i. The *V-Cover* problem is to cover the entire set V with minimum subset of query regions.

It can be proved that the *V*–cover problem is NP-hard by reduction from the minimum set cover problem. Thus, we can employ one of the well-known heuristics for solving the set cover problem, a greedy algorithm, which at each iteration picks the set with the largest number of uncovered elements. Similarly, in order to solve the *V*-cover problem, at each step of iteration, we should pick a representative participant whose query region covers the largest number of uncovered Voronoi cells from V. However, this approach is applicable only in a centralized architecture, where a global knowledge of the environment is available. Toward this end, we need to extend the greedy heuristic to support a distributed architecture. One approach is to design a voting mechanism such that the participants agree locally among their neighbors on selecting a set of representatives. That is, each participant picks a peer from the set of his local peers, based on a score value. Intuitively, the score value captures how significant a participant is in representing other peers, which can be defined based on 1) the number of local peers covered by his query region (*K*), and 2) the number of query regions covering each of his local peers. According to (1), a participant with large query region (i.e., large *K*) is assigned a high score value. However, as (2) suggests, the number of query regions that cover each of those local peers also affects the score value. After being selected, each representative issues a query, filters the result on behalf of every local peer, and sends them the corresponding result.

V. Conclusion and Future Work

In this paper, for the first time we introduced the problem of privacy-aware participatory assignment in PS systems. We proposed the PiRi approach, a solution to the PAPA problem, which addresses the major privacy leaks in PS system.

As a future work, we aim to extend the problem where participants have different constraints (e.g., time, source and destination). Our goal is to incorporate these constraints in the framework yet preserving the privacy of the participants.

Acknowledgment

This research has been funded in part by NSF grant CNS-0831505 (CyberTrust), the NSF Integrated Media Systems Center (IMSC), unrestricted cash and equipment gift from Google, and in part from the METRANS Transportation Center, under grants from USDOT and Caltrans. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "Cartel: a distributed mobile sensor computing system," in *SenSys'06*, pp. 125–138.

[2] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *SenSys'08*, pp. 323–336.

[3] M. C. Gonzalez, C. A. H. R., and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature'08*, vol. 453, pp. 779–782.

[4] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *VLDB'06*, pp. 763–774.

[5] C.-Y. Chow, M. F. Mokbel, and X. Liu, "Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments," in *GeoInformatica'09*.

[6] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Mobihide: A mobilea peer-to-peer system for anonymous location-based queries," in *SSTD'07*, pp. 221–238.

[7] K. L. Huang, S. S. Kanhere, and W. Hu, "Towards privacy-sensitive participatory sensing," in *IEEE PerCom'09*.

[8] L. Hu and C. Shahabi, "Privacy assurance in mobile sensing networks:go beyond trusted servers," in *PerCom 2010 Workshops*.

[9] G. Ghinita, K. Zhao, D. Papadias, and P. Kalnis, "A reciprocal framework for spatial k-anonymity," *Inf. Syst.'10*, vol. 35, no. 3, pp. 299–314.

[10] W. Alsalih, K. Islam, Y. Nú nez-Rodríguez, and H. Xiao, "Distributed voronoi diagram computation in wireless sensor networks," in *SPAA'08*, pp. 364–364.